BOARD OF STUDIES MEETING

B.Tech Degree (Information Technology) Program

THIAGARAJARCOLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to AnnaUniversity)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Approved in BOS Meeting on 13.10.2012 Approved in 45th Academic Council Meeting on 24.11.2012

REVISED CURRICULUM AND DETAILED SYLLABI

FOR

B.Tech DEGREE (Information Technology) PROGRAM

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2012-2013

THIAGARAJARCOLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to AnnaUniversity)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Approved in BOS Meeting on 13.10.2012 Approved in 45th Academic Council Meeting on 24.11.2012

Department of Information Technology

Graduating Students of B.Tech program of IT will be able to

1. Explain and apply appropriate information technologies and employ appropriate methodologies to help an individual or organization achieve its goals and objectives.
2. Manage the information technology resources of an individual or organization.
3. Anticipate the changing direction of information technology and evaluate and communicate the likely utility of new technologies to an individual or organization.
4. Develop IT systems that would perform tasks related to E-governance and/or Health Care Management.
5. Work in a team using common tools and environments to achieve project objectives.

Scheduling of Courses for students admitted in the Academic Year 2012-13

Semester							Laboratory/Project Work		
8(21)	Elective 6 3:0	Elective 7 3:0	Elective 8 3:0				T84 Project 0:12		
7(22)	T71 Management Theory and Practice 3:0	T72 Wireless and Mobile Communication 3:1	Elective 3 3:0	Elective 4 3:0	Elective 5 3:0		T78 Project 0:6		
6(22)	T61 Operations Research 3:1	T62 Accounting and Finance 3:0	T63 Data Warehousing and Mining 3:1	T64 Web Programming 3:0	Elective 1 3:0	Elective 2 3:0	T67 Web Programming Lab 0:1	T68 Software tools Lab 0:1	
5 (25)	T51 Discrete Mathematics 4:0	T52 Software Engineering Methodologies 3:1	T53 Information Storage and Management 3:0	T54 Information Security and Assurance 4:0	T55 Graphics and Visualization 3:1	T56 Platform Technologies 4:0	T57 Platform Technologies Lab 0:1	T58 Network Security Lab 0:1	
4(25)	T41 Engineering Mathematics - IV 4:0	T42 Algorithms: Design Principles 3.0	T43 Operating Systems: Principles 3:1	T44 Computer Networks 3:0	T45 Data bases: Principles and Design 4:0	T46 System and Data Centre Administration 3:0	T47 Network Programming Lab 0:1	T48 System Admin Lab 0:1	T49 Professional Communication 1:1
3(22)	T31 Engineering Mathematics – III 4:0	T32 Object Oriented Programming 3:0	T33 Computer Programming -II 3:0	T34 Data Structures 3:0	T35 Data Bases: Practice 3:0	T36 Computer Organization 3:0	T37 Object Oriented Programming Lab 0:1	T38 Data Bases Lab 0:1	T39 Data Structures Lab 0:1
2 (22)	T21 Engineering Mathematics – II 4:0	T22 Information System Design 3:0	T23 Digital System Design 3:1	T24 Computer Programming -I 3:0	T25 Operating Systems : Configuration and use 3:0	T26 Environment and Ecology 2:0	T27 Operating System Lab 0:1	T28 Computer Programming Lab 0:1	T29 Workshop 0:1
1 (25)	H11 Engineering Mathematics – I 4:0	H12 Physics 3:0	H13 Chemistry 3:0	H14 English 3:0	H15 Basics of EEE 4:0	H16 Basics of ME and CE 4:0	H17 Physics Laboratory 0:1	H18 Chemistry Laboratory 0:1	H19 Engineering Graphics 0:2

REVISED CURRICULUM AND DETAILED SYLLABI

FOR

B.Tech DEGREE (Information Technology) PROGRAM

SECOND SEMESTER

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2012-2013

THIAGARAJARCOLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to AnnaUniversity)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Approved in BOS Meeting on 13.10.2012 Approved in 45th Academic Council Meeting on 24.11.2012

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015

B.Tech. Degree (Information Technology) Program

SUBJECTS OF STUDY

(For the candidates admitted from 2012-2013 onwards)

SECOND SEMESTER

Subject code	Name of the subject	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
T21	Engineering Mathematics II	BS	4	-	-	4
T22	Information System Design	DC	3	-	-	3
T23	Digital System Design	DC	3	1	-	4
T24	Computer Programming -I	DC	3	-	-	3
T25	Operating Systems : Configuration and use	ES	3	-	-	3
T26	Environment and Ecology	HSS	2	-	-	2
PRACTICAL						
T27	Operating System Lab	DC	-	-	2	1
T28	Computer Programming Lab	ES	-	-	2	1
T29	Workshop	ES	-	-	2	1
Total			**18**	**1**	**6**	**22**

BS : Basic Science
HSS : Humanities and Social Science
ES : Engineering Science
DC : Department Core

L : Lecture
T : Tutorial
P : Practical

Note:

1 Hour Lecture/Tutorial is equivalent to 1 credit
2/3 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015

B.Tech. Degree (Information Technology) Program

SCHEME OF EXAMINATIONS

(For the candidates admitted from 2012-2013 onwards)

SECOND SEMESTER

S.No.	Sub. code	Name of the subject	Duration of Terminal Exam. in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	T21	Engineering Mathematics II	3	50	50	100	25	50
2	T22	Information System Design	3	50	50	100	25	50
3	T23	Digital System Design	3	50	50	100	25	50
4	T24	Computer Programming -I	3	50	50	100	25	50
5	T25	Operating Systems : Configuration and use	3	50	50	100	25	50
6	T26	Environment and Ecology	3	50	50	100	25	50
PRACTICAL								
7	T27	Operating System Lab	3	50	50	100	25	50
8	T28	Computer Programming Lab	3	50	50	100	25	50
9	T29	Workshop	3	50	50	100	25	50

* Continuous Assessment evaluation pattern will differ from subject to subject and for different tests. This will have to be declared in advance to students. The department will put a process in place to ensure that the actual test paper follow the declared pattern.

** Terminal Examination will be conducted for maximum marks of 100 and subsequently be reduced to 50 marks for the award of terminal examination marks.

Sub Code	Lectures	Tutorial	Practical	Credit
T 21	4	0	--	4

T21Engineering Mathematics II 4:0

(Common to all branches of Engineering B21, C21, D21, E21, G21, T21)

Program Outcomes addressed

a. An ability to apply knowledge of engineering, information technology, mathematics, and science
b. An ability to identify, formulate and solve engineering problems
c. An ability to engage in life-long learning

Competencies: At the end of the course the students should be able to

1. Formulate and solve problems of engineering dynamics using different differential operators.
2. Formulate the problem of computing areas and volumes through vector integration, and determine them by applying Green, Stokes and Divergence theorems
3. Determine maxim and minima of functions of several variables using analytical and Lagrangian multipliers methods
4. Determine the values of multiple integrals directly or by changing the order of integration or by making transformation with Jacobians.
5. Determine areas and volumes of geometrical figures using multiple integrals, beta and gamma functions.
6. Analyze functions of complex variable in terms of continuity, differentiability and analyticity.
7. Apply Cauchy-Riemann equations and harmonic functions to problems of fluid mechanics, thermodynamics and electro-magnetic fields.
8. Find singularities of complex functions and determine the values of integrals using residues.
9. Geometrically interpret conformal and bilinear transformations

Course Level Learning objectives

Understand

1. Distinguish between solenoidal and irrotational vectors?
2. How do you understand an analytic function? Give examples of an analytic function and a non-analytic function?
3. Distinguish among the three types of singularities of a complex valued function f(z)?
4. Examine the following function for extreme values

$$u = x^4 + y^4 - 2x^2 + 4xy - 2y^2$$

5. Explain the difference between a stationary value and an extreme value of a real valued function f(x, y) illustrating with examples?
6. If the three thermodynamic variables P, V and T are connected by a relation f(P,V,T) = 0, show that

$$\left(\frac{\partial P}{\partial T}\right)_V \left(\frac{\partial T}{\partial V}\right)_P \left(\frac{\partial V}{\partial P}\right)_T = -1$$

7. Verify Stoke's theorem for $\bar{F} = (2x - y)\bar{i} - yz^2\bar{j} - y^2z\bar{k}$ where S is the upper half of the sphere $z^2 + y^2 + z^2 = 1$ and C is the boundary in the xy plane?

Apply

1. Find the unit vector normal to the surface $x^2 + 2y^2 + z^2 = 7$ at (1, -1, 2)?
2. Establish the result

$$Div(\bar{u} \times \bar{v}) = \bar{v}.curl\,\bar{u} - u.curl\,\bar{v})$$

Deduce that if $\bar{u}, \bar{v}$ are irrotational, then $\bar{u} \times \bar{v}$ is solenoidal?

3. Transform the equation $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0$ into polar coordinates?
4. A fluid motion is given by $\bar{V} = (y + z)\bar{i} + (z + x)\bar{j} + (x + y)\bar{k}$. Is this motion irrotational? If so, find the velocity potential? Is this motion possiblefor an incompressible fluid?
5. Determine the value of $\int_0^5 \int_0^{x^2} x(x^2 + y^2)dxdy$?
6. Find the volume bounded by the cylinder $x^2 + y^2 = 4$ and the planes y + z = 4 and z = 0 using multiple integrals?
7. The plane $\frac{x}{a} + \frac{y}{b} + \frac{z}{c} = 1$ meets the axes at A, B and C. Apply Dirichlet's integrals to find the volume of the tetrahedron OABC?
8. Transform the integral $\int_0^{\pi} \int_0^{a} r^3 \sin\theta\cos\theta\, dr\, d\theta$ into Cartesian form and determine the value of the integral?
9. If $\omega = \phi + i\psi$ represents the complex potential for an electric field and $\psi = x^2 - y^2 + \frac{x}{(x^2 + y^2)}$, determine the function ϕ?

10. Find the bilinear transformation mapping the points z = 1, i, -1 into the points ω = 0, 1, ∞ ?

11. Use Cauchy's integral formula to determine $\int_C \frac{e^{2z}}{(z-1)(z-2)}$ where 'C' is the circle |z| = 3?

12. Obtain the Laurent series expansion of f(z) = $\frac{z}{(z^2-1)(z^2+4)}$ for regions (a) |z|<1 (b) 1<|z|<2 (c)|z|>2?

13. Determine the value of the integral $\int_{-\infty}^{\infty} \frac{x^2\,dx}{(x^2+1)(x^2+4)}$ using the theorem of residues?

14. Apply the method of contour integration to prove that $\int_0^{2\Pi} \frac{\cos 2\theta}{5+4\cos\theta}\,d\theta = \frac{\Pi}{6}$ clearly indicating the contour selected?

15. A tent on a square base of side x, has vertical sides of height y, and the top is a regular pyramid of height h. Find x and y in terms of h, if the canvas required for its construction is to be minimum for the tent to have a given capacity?

Assessment Pattern

	Bloom's Category	Test 1	Test 2	End-semester examination
1	Remember	10	10	0
2	Understand	30	30	30
3	Apply	60	60	70
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	0	0	0

Course Contents:

1. Functions of Several Variables:

1.1 Partial derivatives and Jacobians

1.2 Total differentiation and applications

1.3 Lagrangian Multiplier method

1.4 Applications to Maxima and Minima

2. Multiple Integrals:

2.1 Double integrals and areas

2.2 Triple integrals and volumes

2.3 Change of order of integration

2.4 Beta and Gamma functions with applications

2.5 Change of variables between Cartesian and polar with applications

3. Vector calculus:

3.1 Vector Differentiation with simple applications

3.2 Operators Grad, div and curl with properties

3.3 Applications to Physics

3.4 Vector Integration(three famous theorems)

3.5 Applications to areas and volumes

4. Complex Differentiation:

4.1 Analytic functions, C-R equations and properties

4.2 Harmonic Functions and Milne-Thompson Method

4.3 Applications to flow problems

4.4Conformal maps and bilinear transformations

4.5 Applications of the bilinear transformations

5. Complex Integration:

5.1 Cauchy's theorem and consequences

5.2 Evaluating integrals using Cauchy's integral formula

5.3 Taylor and Laurent expansions

5.4 Singularities, poles and Cauchy residue theorem

5.5 Contour integration using unit circle and semicircular contours

Syllabus

Functions of Several Variables: Partial derivatives and Jacobians, Total differentiation and applications, Lagrangian Multiplier method, Applications to Maxima and Minima **Multiple Integrals:** Double integrals and areas, Triple integrals and volumes, Change of order of integration, Beta and Gamma functions with applications, Change of variables between Cartesian and polar with applications **Vector calculus:** Vector Differentiation with simple applications, Operators Grad, div and curl with properties, Applications to Physics, Vector Integration(three famous theorems), Applications to areas and volumes **Complex Differentiation:** Analytic functions, C-R equations and properties, Harmonic Functions and Milne-Thompson Method, Applications to flow problems, Conformal maps and bilinear transformations, Applications of the bilinear transformations **Complex Integration:** Cauchy's theorem and consequences, Evaluating integrals using Cauchy's integral formula, Taylor and Laurent expansions, Singularities, poles and Cauchy residue theorem, Contour integration using unit circle and semicircular contours

Text Book

B.S. Grewal: Higher Engineering Mathematics, 39th Edn.,Khanna Publishers, New Delhi

References

1. Lecture Notes by the faculty of Department of Mathematics, TCE, Madurai
2. Veerarajan t.: Engineering Mathematics, 3rd Edn., Tata McGraw Hill, New Delhi, 2004
3. Venkataraman M.K.: Multiple Integrals and Gamma, Beta functions, National Publishing Co., 2004
4. Manish Goyal: Textbook of Engineering Mathematics, Lakshmi Publishing, New Delhi
5. Venkataraman M.K., Engineering Mathematics (Vol. II), ational Publishing, Chennai
6. Kreyszig E., Advanced Engineering Mathematics, 8th Edn. John Wiley & Sons

Lecture Schedule

No.	Topic	No. of Lectures
1.	**Functions of Several Variables**	
1.1	Partial derivatives and Jacobians	2
1.2	Total differentiation and applications	2
1.3	Lagrangian Multiplier method	2
1.4	Applications to Maxima and Minima	2
2.	**Multiple Integrals**	
2.1	Double integrals and areas	1
2.2	Triple integrals and volumes	1
2.3	Change of order of integration	2
2.4	Beta and Gamma functions with applications	2
2.5	Change of variables between Cartesian and polar with applications	1
3	**Vector calculus**	
3.1	Vector Differentiation with simple applications	1
3.2	Operators Grad, div and curl with properties	3
3.3	Applications to Physics	1

No.	Topic	No. of Lectures
3.4	Vector Integration(three famous theorems)	4
3.5	Applications to areas and volumes	3
4	**Complex Differentiation:**	
4.1	Analytic functions, C-R equations and properties	3
4.2	Harmonic Functions and Milne-Thompson Method	2
4.3	Applications to flow problems	1
4.4	Conformal maps and bilinear transformations	2
4.5	Applications of the bilinear transformations	2
5.	**Complex Integration**	
5.1	Cauchy's theorem and consequences	2
5.2	Evaluating integrals using Cauchy's integral formula	2
5.3	Taylor and Laurent expansions	2
5.4	Singularities, poles and Cauchy residue theorem	2
5.5	Contour integration using unit circle and semicircular contours	4

Course Designers:

1. V. Mohan vmohan@tce.edu
2. N. Kannannkmat@tce.edu
3. M. Kameswarimkmat@tce.edu
4. K. Angaleeswarikamat@tce.edu
5. P. Subramanianpsmat@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T22	3	0	--	3

T22 Information System Design 3:0

Preamble

This course aims to emphasis the need for Information system and provides coverage of various phases in building Information systems. It provides an overview of different kinds of Information Systems, describes how it relates to other computing disciplines. The goal is to help students understand the diverse application of Information Systems and the challenges inherent in the diffusion of Information systems.

Program outcomes addressed

Graduates will demonstrate

a. An ability to apply knowledge of engineering, information technology, mathematics and science.

c. An ability to design a system, component or Process to meet desired needs within realistic constraints such as economic ,environmental, social, ethical, health and safety, manufacturability, and sustainability.

e. An ability to identify, formulate and solve engineering problems.

h. An ability to consider social, environmental, economic and ethical impact of engineering activities in a given context.

Competencies

Students will be able to

1. Differentiate components of Information Systems like data, Information and System.
2. Understand the System Requirement Specification document.
3. Develop high level design for stated requirements using Data flow Diagram, Process description Tool, E-R Diagram.
4. Comprehend Transaction processing system and Management Reporting system and apply to different types of Organisations
5. Understand the components of Information system to support the different Business functions and perform a feasibility analysis on selection of implementation of Information Systems.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3	End-semester examination
1	Remember	50	40	40	40
2	Understand	50	30	30	30
3	Apply	0	20	20	20
4	Analyze	0	10	10	10
5	Evaluate	0	0	0	0
6	Create	0	0	0	0

Course Level Learning Objectives

Remember

1. State the difference between data and information.
2. Identify the various information systems.
3. What is a data flow diagram?
4. Define Data Dictionary.
5. What is meant by Cost-Benefit analysis?
6. What is feasibility study of a solution?

Understand

1. What would be the strategic, tactical, operational information required for an educational institution?
2. Enlist the differences between online transaction processing and Batch processing.
3. Differentiate between a flowchart and Data flow diagram.
4. When should questionnaires be used and be not used in requirement gathering?
5. Can the name of a person be used as a code in a bank account? Justify.
6. Why are input data records divided into batches for offline entry?
7. Distinguish between technical, operational and economic feasibility.

Apply

1. Illustrate when is batch processing preferred over on-line transaction processing?
2. A university administrator calls a systems analyst to improve the administration of sponsored research projects. The main problems are delay in giving latest financial

position to project coordinators, reconciliation of advances given to coordinators, prompt demands not sent to sponsors to collect promised grants and lack of information to answer following questions:

Which areas of research get maximum grants?

Which agency aids which type of projects?

What trends can be seen in the nature of grants?

Classify the above problems into missing functions, unsatisfactory Performance and excessive cost of operation.

Set the goals to meet the deficiencies and quantify them.

3. Admission procedure in a University is as follows: An advertisement is issued giving essential qualifications for the course, the last date for receipt of application, and the fee to be enclosed with the application. A clerk in the Registrar's office checks the received applications to see if mark sheet and fee are enclosed and sends valid applications to the concerned academic department. The department checks the application in detail and decides the applicants to be omitted, those to be put in the waiting list, and those rejected. Appropriate letters are sent to the Registrar's office which intimates the applicant. Give physical and logical Data Flow Diagrams corresponding to the above problem.
4. A magazine is published monthly and is sent by post to its subscribers. Two months before the expiry of subscription, a reminder is sent to the subscribers. If subscription is not received within a month, another reminder is sent. If renewal subscription is not received up to two weeks before the expiry of the subscription, the subscriber's name is removed from the mailing list and the subscriber informed. Obtain logical Data Flow Diagrams for this problem.
5. An organization maintains an employee file in which each record has following data: {Employee No., employee name, employee gross pay}. It has been decided to increase the pay as per the following formula:

For pay of Rs. 1000 or less increase 15%.

For pay of more than Rs. 1000 but up to Rs. 2500 increase 10%.

For pay over Rs. 2500 increase 5%.

(i) Write a structured English processing rule corresponding to the above policies.

(ii) Express the policies as a decision table.

6. Develop E-R diagram for the following:

(i) Customer withdraws money from his account.

(ii) Students write examinations.

(iii) Students attend classes.

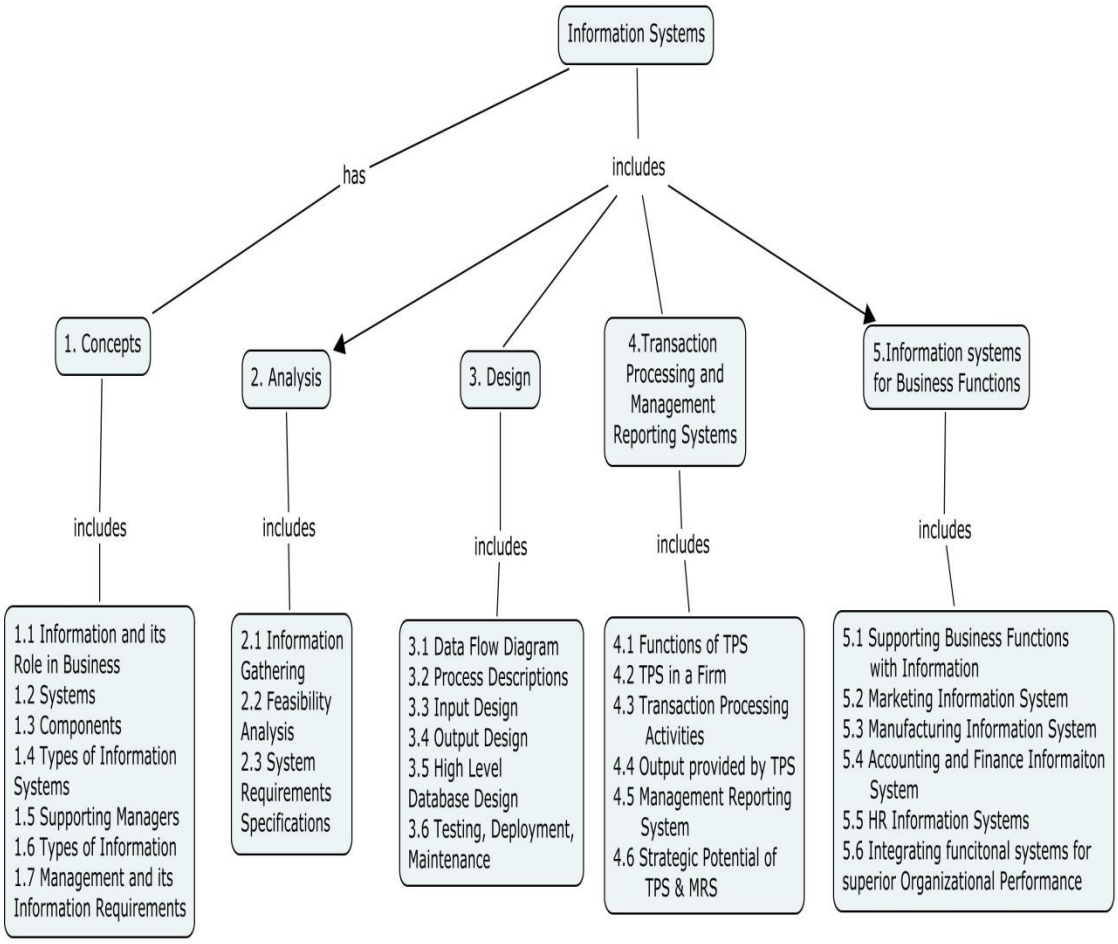
(iv) Professors write books.

(v) Driver drives a car.

Analyze

1. In an approach to automate a travels reservation operation, it was worked out that the capital cost was Rs. 2 lakhs. The project will be completed by 4 months and the salary for the people hired for this works out to Rs. 30,000 per month. The cost due of maintenance is Rs. 1500/- per month. The expected electricity charges are Rs. 6000 for two months. Analyze how the ROI will work out if they can save Rs.35000/month using:
    a. Simple pay back method
    b. Simple payback method with interest if interest is 2% per month
    c. Present value method with interest as 2 % per month
2. The Process Description of a DFD Process block was described using a) Structured English and b) using Decision Table. Compare and contrast which one is better if the programmer involved is a novice person. Substantiate your stand.
3. The Process Description of a DFD Process block was described using a) Structured English b) Decision Tree and c) using Decision Table. Compare and contrast which one is better if the documentation is targeted towards the management. Justify
4. Can a relation be in 2NF but not in 3NF? Give example and justify
5. Compare the logical DFD with a physical DFD. Which of these two will you use to represent the current manual system "as-is" while preparing report to the management. Justify
6. While studying the information system in a textile industry with an aim of modernizing, the management gives the configuration of their existing hardware. Is it possible to develop information system considering the available hardware and defining only new software to meet their requirements. Justify giving example scenarios.

Concept Map



Syllabus:

Concepts of Information Systems - Information and Its Role in Business – Systems - Components of Information Systems - Types of Information Systems - Supporting Managers with Information Systems - Types of Information - Management and Information Requirements - **Information Systems Analysis -** Overview of Design of an Information System - Information Gathering - Feasibility Analysis - System Requirement Specification - **Information System Design -** Dataflow Diagram - Process Description - Input Design - Output Design - Overview of Database Design - Testing, Deployment and Maintenance – **Transaction Processing and Management Reporting Systems –** Functions of Transaction Processing System – Transaction Processing Subsystem in a Firm – Transaction Processing Activities – Output Provided by Transaction Processing System – Management

reporting System – Strategic Potential of Transaction Processing and Management Reporting Systems - **Information Systems For Business Functions –** Supporting Business Functions in an Enterprise with Information – Marketing Information System – Manufacturing Information System – Accounting and Financial Information System – Human Resource information System **-** Integrating Functional Systems for Superior Organizational Performance. Case study on the above

Text Books:

1. V.Rajaraman : Analysis and Design of Information Systems, Second edition, Prentice Hall of India, 2002
2. Vladimir Zwass: Foundations of Information Systems, Irwin / McGraw Hill International Edition, 1998

Reference Books:

1. Shouhong Wang, Hai Wang, Information Systems Analysis and Design, Universal Publishers, 2012
2. Gary B. Shelly, Harry J. Rosenblatt – 2011, System Analysis and Design, ninth edition, Cengage Learning, 2011

3. James A. Senn, Analysis and design of information systems McGraw Hill International Edition, 1989

Course Contents and Lectures schedule

Sl. No	Topics	No of Sessions
1	**Concepts of Information Systems**	
1.1	Information and Its Role in Business	1
1.2	Systems	1
1.3	Components of Information Systems	1
1.4	Types of Information Systems	3
1.5	Supporting Managers with Information Systems	1
1.6	Types of Information	1
1.7	Management and Its Information Requirements	1
2	**Information Systems Analysis**	
2.0	Overview of Design of an Information System	2

2.2	Information Gathering	1
2.3	Feasibility Analysis	3
2.4	System Requirement Specification	1
3	**Information System Design**	
3.1	Dataflow Diagram	3
3.2	Process Description	3
3.3	Input Design	2
3.4	Output Design	2
3.5	High Level Database Design	3
3.6	Testing, Deployment and Maintenance	2
4	**Transaction Processing and Management Reporting Systems**	
4.1	Functions of Transaction processing system	1
4.2	Transaction Processing Subsystem in a Firm	1
4.3	Transaction Processing Activities	1
4.4	Output Provided By Transaction Processing System	1
4.5	Management Reporting System	2
4.6	Strategic Potential of Transaction Processing and Management Reporting Systems	1
5	**Information Systems for Business Functions**	
5.1	Supporting Business Functions in an Enterprise with Information	1
5.2	Marketing Information System	2
5.3	Manufacturing Information System	1
5.4	Accounting and Financial Information System	1
5.5	Human Resource Information Systems	1
5.6	Integrating Functional Systems for Superior Organizational Performance	1
	Total Lectures	45

Course Designers:

1. N.Sivaram nsit@tce.edu
2. U. Nadarajan unit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T23	3	1	-	4

T23 Digital System Design 3:1

Preamble: This course covers the basic tools for the design of digital circuits and provides fundamental concepts used in the design of digital systems. The main objective of this subject is to make the students to understand the basic building blocks of computers, logic gates, combinational and sequential circuits, and to provide the discussion of the memory and programmable logic. This course acts as prerequisite for Computer Organization.

Program Outcomes addressed:

Graduates will demonstrate

a. An ability to apply knowledge of engineering, information technology, mathematics, and science.

c. An ability to design a system or component, or process to meet stated specifications.

d. An ability to identify, formulate and solve engineering problems.

Competencies:

Students will be able to

1. Present the various binary systems suitable for representing data in digital systems.
2. Show the correlation between Boolean expressions and their corresponding logic diagrams by understanding the basic postulates of Boolean algebra.
3. Simplify digital circuits constructed with logic gates by simplifying Boolean expressions using a)MAP method b)VEM.
4. Design the basic components like adders, multipliers and code converters used in the design of digital systems.
5. Design and analyze the performance of the basic building blocks that are used to construct complex digital systems including flip-flops, registers and counters.
6. Design the memory and programmable logic by studying various memories like ROM, RAM, Cache memory and virtual memory.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3	End-semester examination
1	Remember	20	20	20	20
2	Understand	20	20	20	10
3	Apply	20	20	20	30
4	Analyze	20	20	20	20
5	Evaluate	0	0	0	0
6	Create	20	20	20	20

Course Level Learning Objectives

Remember

1. What is meant by Gray code?
2. Define a flip flop.
3. State Demorgan's theorem
4. List out all the applications of a register.
5. Name the circuit that compares two numbers.
6. Write the role decoding circuits in selecting the memory word.

Understand

1. Explain the basic theorems and properties of Boolean algebra.
2. Discuss NAND-NOR implementation of Boolean expressions.
3. Describe how does a decimal adder adds two numbers.
4. Interpret the role of synchronous counters in designing the digital systems.
5. Compare Programmable Logic Array and Programmable Array Logic
6. Explain why J-K flip-flops produce more don't-care terms than the other flip- flop types, even when all of the states are specified.

Apply

1. Do the following conversions
    a. $(934.89)_{10}$ to Binary
    b. $(101.111)_2$ to decimal
    c. $(939AB)_{16}$ to decimal
    d. $(86.37)_{10}$ into octal

2. Simplify the following Boolean Expression using K Map and construct thelogical circuit
   F(a,b,c,d) = Σ (1, 2, 4, 5, 6, 7,8, 11, 12, 14) + Σ (3, 10, 13)
3. Design a synchronous circuit that has a single input variable and single output variable. The input data are received serially. The output Z is to change only when three consecutive input bits have the same value.
4. Simplify the following Boolean Expression using Boolean theorems and postulates and construct the logical circuit

f(p,q,r,s) = pqrs + p'qrs+ pq'rs'+ pq'rs + p'q'rs' + pqrs' + pqr's + p'q'r's'

5. Design a modulo-16 counter.
6. Examine the number of 256k * 8 bits RAM chips that are needed to construct a memory of 3M * 32 bits capacity.

Create

1. Simplify the following using QuineMc-Cluskymethod.. Construct the logic circuit for the simplified expression.

$$f(w,x,y,z) = \sum 0,1,2,4,5,7,9,12,14,15 + \sum_{\phi} 3,10,11$$

2. Implement the following four Boolean expressions with three half adders

   D = A XOR B XOR C

   E = A'BC + AB'C

   F = ABC' + (A'+B)C

   G = ABC

3. Design a simple 4 bit register with D flip-flops.
4. A combinational circuit is defined by the following three Boolean functions:

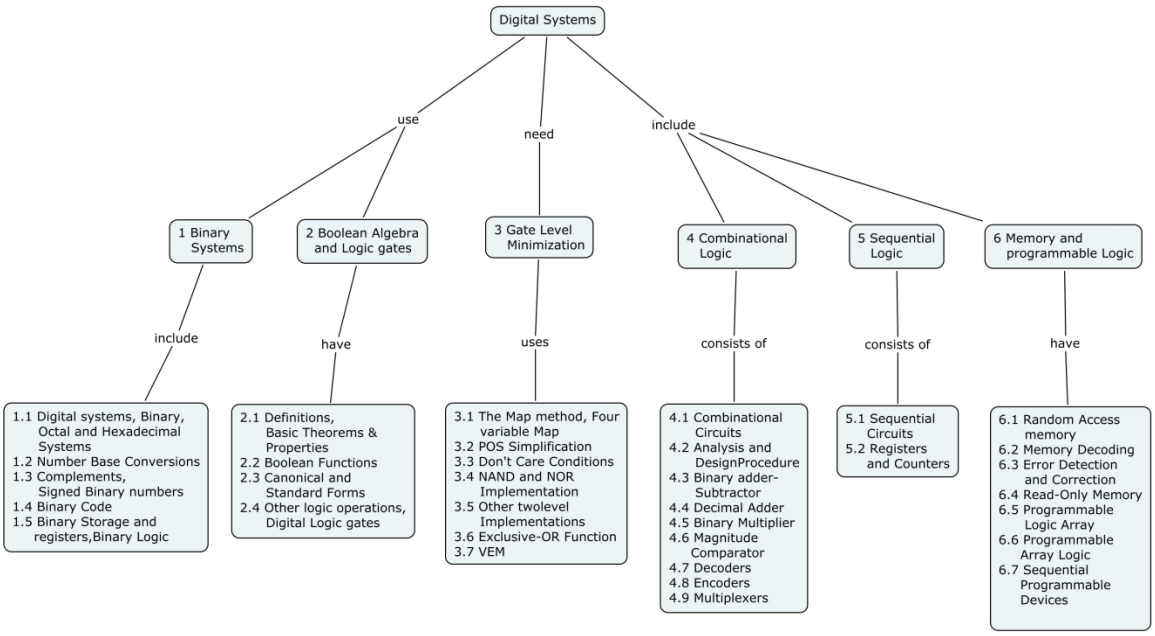
   F1=x'y'z'+xz

   F2=xy'z'+x'y

   F3=x'y'z'+xy

   Construct the circuit with a decoder.
5. Design a JK flip-flop using a D flip-flop, a 2-1 line multiplexer and an inverter.
6. Design a serial 2's complementer with a shift register and a flip-flop.

Concept Map



Syllabus

Binary Systems: Digital Systems - Binary, Octal and Hexadecimal Numbers- Number Base Conversions – Complements - Signed Binary Numbers - Binary Codes - Binary Storage and Registers - Binary Logic - **Boolean Algebra and Logic Gates:** Definitions of Boolean Algebra - Basic Theorems and Properties of Boolean Algebra - Boolean Functions - Canonical and Standard forms- other Logic Operations - Digital Logic Gates - **Gate Level Minimization:** The Map Method - Four Variable Map - Product of Sums(POS) simplification - Don't care conditions - AND and NOR Implementation - Other Two level Implementations - Exclusive–OR Function - Variable Entered mapping(VEM) **- Combinational Logic:** Combinational Circuits - Analysis and Design Procedure - Binary Adder- Subtractor - Decimal Adder - Binary Multiplier - Magnitude Comparator - Decoders and Encoders – Multiplexers - **Sequential Logic:** Sequential Circuits - Latches- Flip-flops- Analysis of clocked sequential Circuits - Design Procedure - Registers and Counters -Registers - Shift Registers - Ripple Counters - Synchronous Counters - Other Counters - **Memory and Programmable Logic:** Random Access Memory - Memory Decoding - Error Detection and Correction - Read-only Memory - Programmable Logic Array - Programmable Array Logic - Sequential Programmable Devices.

Text Book:

1. M.Morris Mano: Digital Design, Third Edition, Pearson Education, 2006

Reference Book:

1. Floyd Thomas L, Jain: Digital Fundamentals, Eighth Edition, Pearson Education, India, 2009
2. Ronald J.Tocci,NealS.Widmer, Gregory L.Moss: Digital Systems: Principles and Applications, Eleventh Edition, Prentice Hall of India Private Limited, New Delhi, 2010

Course contents and Lecture schedule

No.	Topic	No. of Lectures
1.	**Binary Systems**	
1.1	Digital Systems - Binary, Octal and Hexadecimal Numbers	1
1.2	Number Base Conversions	2
1.3	Complements - Signed Binary Numbers	1
1.4	Binary Codes	1
1.5	Binary Storage and Registers - Binary Logic	1
2.	**Boolean Algebra and Logic Gates**	
2.1	Definitions of Boolean Algebra - Basic Theorems and Properties of Boolean Algebra	1
2.2	Boolean Functions	1
2.3	Canonical and Standard forms	1
2.4	Other Logic Operations - Digital Logic Gates	1
3.	**Gate Level Minimization**	
3.1	The Map Method - Four Variable Map	2
3.2	Product of Sums(POS) simplification	1
3.3	Don’t care conditions	1
3.4	NAND and NOR Implementation	1
3.5	Other Two level Implementations	1

3.6	Exclusive –OR Function	1
3.7	Variable Entered mapping(VEM)	2
4	**Combinational Logic**	
4.1	Combinational Circuits	1
4.2	Analysis and Design Procedure	1
4.3	Binary Adder-Subtractor	2
4.4	Decimal Adder	1
4.5	Binary Multiplier	2
4.6	Magnitude Comparator	1
4.7	Decoders and Encoders	2
4.8	Multiplexers	1
5.	**Sequential Logic**	
5.1	**Sequential Circuits**	
5.1.1	Latches - Flip-flops	2
5.1.2	Design Procedure	2
5.2	**Registers and Counters**	
5.2.1	Registers	1
5.2.2	Shift Registers	2
5.2.3	Ripple Counters	1
5.2.4	Synchronous Counters	1
5.2.5	Other Counters	2
6.	**Memory and Programmable Logic**	
6.1	Random Access Memory	1
6.2	Memory Decoding	1
6.3	Error Detection and Correction	1

6.4	Read-only Memory	1
6.5	Programmable Logic Array	1
6.6	Programmable Array Logic	1
6.7	Sequential Programmable Devices	1
	Total Lectures	**48**

Course Designers

1. R.MuthuSelvi rmsit@tce.edu

2. G.Prakashprakashguru@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T24	3	0	--	3

T24 Computer Programming -I 3:0

Preamble: The course on Computer Programming is intended to introduce the students to computational thinking, the methodology of programming with emphasis on modularity and the coding of computer programs using C language. Upon completion of the course, the students would be able to master the principles of structured programming and demonstrate significant experience in C Program development environment.

Program Outcomes addressed

Graduates will demonstrate

a. An ability to apply knowledge of engineering, information technology, mathematics and science.
b. An ability to design and conduct experiments, as well as to analyze and interpret data.
c. An ability to design a system, component or process as per needs and specifications.
d. An ability to identify, formulate and solve engineering problems.

Competencies

Students will be able to

1. Comprehend the following terms in the context of problem solving by a computer: Problem specification, input-output analysis, algorithm, flowchart, pseudo-code, High level language, assembly language, machine language, and compilation and Execution.
2. Understand the fundamentals of programming languages like constants and variables, declarations, data types, operators and expressions, arrays, input and output statements, control and iterative statements.
3. Analyze the reasons for different constructs for iteration, such as "for" loops and "do...while" loops.
4. Understand Modular Programming technique and develop computer programs using functions.
5. Distinguish between iteration and recursion and decide where iteration is better than recursion and where recursion is better than iteration in problem solving.
6. Select appropriate storage class specifications such as auto, static, extern and register variables in developing application programs.
7. Understand the concept of pointers to variables, one dimensional and multi-dimensional arrays, structures and functions.

8. Understand the syntax and semantics of structures and unions and apply it in development of simple projects like payroll management, Grade calculation etc.
9. Deploy the concept of Dynamic memory allocation for simple programs like matrix manipulation, stack implementation etc.
10. Differentiate the ways in which Macros and functions are handled during Compilation and use preprocessor directives for efficient programming.
11. Utilize the extensive set of library functions available in C for creating and processing data files.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3	End-semester examination
1	Remember	20	10	10	10
2	Understand	20	20	20	20
3	Apply	60	50	50	50
4	Analyze	0	20	20	20
5	Evaluate	0	0	0	0
6	Create	0	0	0	0

Course Level Learning Objectives

Remember

1. What is structured programming?
2. Define Operator precedence. What are the relative precedence's of Arithmetic Operators?
3. What are pre-processor directives?
4. Enlist some of the bitwise operators available in C.
5. Give the Syntax of switch-case statement?
6. What is a pointer?

Understand:

1. Compare while loop with do – while Loop.
2. Distinguish between call by value and call by reference with an example.
3. Explain how recursive functions affect the run time efficiency.
4. Differentiate between Structure and Union in C.
5. Explain the various storage classes in C.

6. How is garbage collection done in C?

Apply

1. Write an interactive C code that will encode and decode a line of text. To encode a line of text, proceed as follows:
    a. Convert each character, including blank spaces in to ASCII equivalent.
    b. Generate a positive random integer. Add this integer to the ASCII equivalent of each character.
    c. Display the characters that correspond to the encoded ASCII values.

Reverse the procedure for decoding the line of text.

2. Write a program to sort a list of strings into alphabetical order using an array of pointers.
3. Write a program to count the number of occurrences of any two vowels in succession in a line of text. For Example, in the sentence "THIAGARAJAR COLLEGE OF ENGINEERING" such occurrences are EE,IA.
4. Write a program to read a file and count the number of characters, spaces, tabs and new lines present in it.
5. Create a structure to specify data on students given below:

Roll number, Name, Department, Course, Year of Joining

Assume there are not more than 450 students in the college,

    1. Write a function to print names of all students who joined in a particular year.
    2. Write a function to print the data of a student whose roll number is given.

6. A factory has 3 divisions and stocks 4 categories of products. An inventory table is updated for each division and for each product as they are received. There are three independent suppliers of products to the factory:
    a. Design a data format to represent each transaction.
    b. Write a program to take a transaction and update the inventory.
    c. If the cost per item is also given, write a program to calculate the inventory sales.

Analyze:

1. Determine the hierarchy of operations and evaluate the expression k=3/2*4+3/8+3 and convert the equation $z = \frac{\frac{8.8(a+b)2}{c} - 0.5 + 2a/(q+r)}{(a+b)*(\frac{i}{m})}$

2. Rewrite the following program using conditional operators.

```
main()
{
floatsal;
printf("ENTER SALARY");
scanf("%f",&sal);
if(sal<40000 &&sal>25000)
printf("MANAGER");
else
if(sal<25000 &&sal>15000)
printf("ACCOUNTANT");
else
printf("CLERK");
}
```

3.Evaluate the output of the following code:

```
main()
{
int x=4,y=0,z;
while (x>=0)
{
x--;
y++;
if (x==y)
continue;
else
printf("\n%d%d",x,y);
}
}
```

4.Consider the following code segment:

```
inti,j=25;
int*pi,pj=&j;
……
……
*pj=j+5;
```

```
j=*pj+5;
pj=pj;
*pi=i+j;
```

Each integer occupies 2 bytes of memory. The value assigned to i begin at the hexadecimal address F9C and the value assigned to j begins at address F9E. Compute the values of following:

(a) &i (b)&j (c)pj (d) *pj (e)i (f) pi (g)*pi (h) (pi+2)
(i) (*pi+2) (j) *(pi+2)

5. How many bytes in memory would be occupied by the following array of pointers to strings? How many bytes would be required to store the same strings, if they are stored in a two dimensional character array?

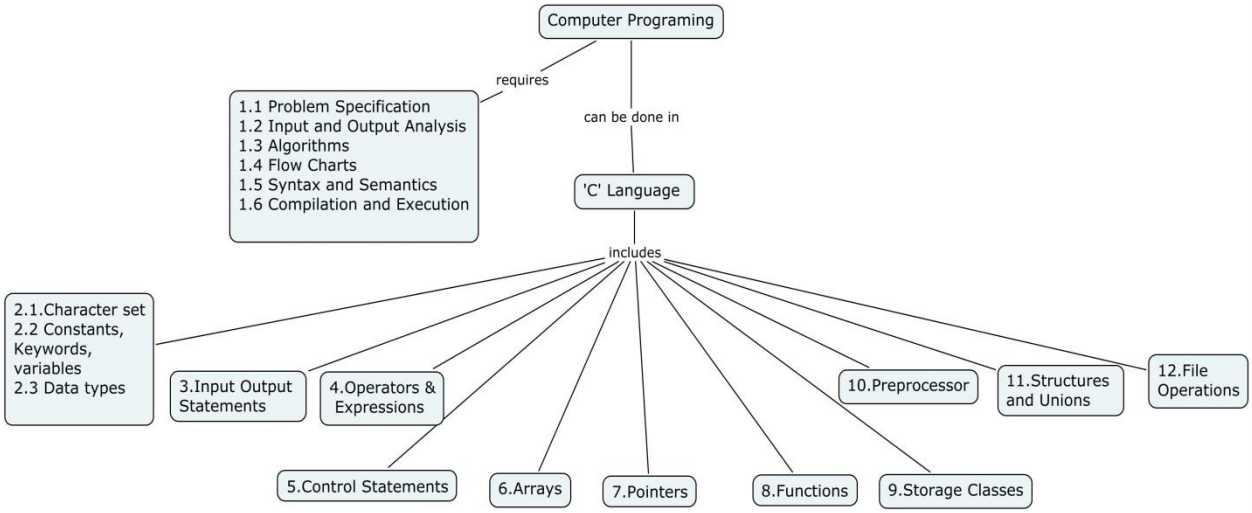
Char *mess[] = {"DEPARTMENT""OF""INFORMATION""TECHNOLOGY"};

Can an array of pointers to strings be used to collect strings from the keyboard? Justify your answer.

6. Analyze the meaning of the following declarations involving pointers:

in (*p[10]) (char a); int *(*p[10]) (char a); int *(*p[10]) (char *a);
int *(*p) (char(*a)[]); int *p (char *a[]));

Concept Map:



Syllabus:

Computer Programming - Problem specification, input-output analysis, algorithm, flowchart, Preprocessing, Compilation and Execution, Syntax and semantics.**Programming in C** – Significance of C Language, C Character set, Keywords, Constants and variables, data types, Declaration statements. **Input and Output statements –** getchar, putchar, scanf, printf, gets, puts functions **Operators and Expressions –**Arithmetic, Relational, Logical, Conditional operators and Bitwise operators, Operator Precedence, Type Conversion, Type casting. **Control Statements** –**Branching** –if-else and if- else ladder, switch case. **Looping** –while loop, do –while loop, for loop, Break, continue and Goto statements **Arrays** –Single and Multidimensional Arrays, Strings **Pointers –** Declaration, Operations on Pointers, Pointers and one dimensional Arrays, Pointers and Multidimensional Arrays, Array of pointers, Dynamic Memory allocation. **Functions –** Function Definition, Function Prototypes, Passing arguments to a function, Pointers to functions, Recursion. **Storage Classes** – automatic, external, static and register **C Preprocessor** – Preprocessor directives, Macro Expansion, Conditional Compilation. **Structures and Unions** – Definition, Processing a structure, Array of structures, Pointers to structures, Passing structures to Functions and Returning structure variables from functions, Self Referential Structures, Unions and Bit Fields. **Files** – Reading and writing a file, Processing a data file, Unformatted Files and Binary Files.

Textbooks

1. Byron S.Gottfried, "Programming with C", Tata MC Graw Hill, Second Edition -2006.
2. Al Kelley and Ira Pohl, "A Book On C", Addison-Wesley , Fourth Edition, ISBN 0-201-18399-4.

References

1. Donald.E.Knuth, "The Art of Computer Programming : Volume 1- Fundamental Algorithms" Addison-Wesley, Third Edition, ISBN 0-201-03801-3.
2. YashavantKanetkar, "Programming in ANSI C", 2nd Edition-BPB Publications.
3. YashavantKanetkar, "Understanding Pointers in C", 2nd Edition BPB Publications.

Course Contents and Lecture Schedule

No	Topic	Hours
1	**Introduction to Computer Programming**	
1.1 1.2	Problem Specification Input Output Analysis	1

1.3	Algorithm	1
1.4	Flow Chart	2
1.5 1.6 1.7	Preprocessing Compilation and Execution Syntax and semantics	1
2	**Fundamentals of C** Significance of C Language	1
2.1 2.2	Character set Constants, Variables and Key words	1
2.3	Data types and Declarations	1
3	**Input and Output Statements**	2
4	**Operators and Expressions**	
4.1	Arithmetic, Relational, Logical and Conditional Operators, Bit wise Operators	1
4.2	Operator Precedence	1
4.3 4.4	Type Conversion Type Casting	1
5	**Control Statements**	
5.1	Branching	1
5.2	Looping	1
5.3	Break, Continue and Goto statements	1
6	**Arrays**	
6.1	Single Dimensional Arrays	1
6.2	Multi-Dimensional Arrays	2
6.3	Strings	2
7	**Pointers**	
7.1	Pointer to variables -Declaration and Operations	1
7.2 7.3	Pointers and one dimensional Arrays Pointers to Strings	1
7.4	Pointers and Multi-dimensional Arrays	1
7.5	Array of Pointers	1

7.6	Dynamic Memory allocation	2
8	**Functions**	
8.1	Function Definition and Function prototypes	1
8.2	Passing Arguments to a function	2
8.3	Pointers to Functions	2
8.4	Recursion	2
9	**Storage Classes**	1
10	**C Preprocessor**	
10.1	Preprocessor Directives	1
10.2 10.3	Macro Expansion Conditional Compilation	1
10.4	Multiple File Inclusions	1
11	**Structures and Unions**	
11.1	Definition and Processing of Structure	1
11.2 11.3	Array of Structures Pointers to Structures	1
11.4 11.5	Passing Structures to Functions, Returning structure variables from functions Self-Referential Structures	1
11.6 11.7	Unions Bit Fields	1
12	**Files**	
12.1	Reading and writing to a file	2
12.2 12.3	Processing a Data file Unformatted Files and Binary files	2
	Total Lectures	**45**

Course Designers:

1. C.Jeyamala jeyamala@tce.edu
2. P.Karthikeyankarthikit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T 25	3	0	--	3

T25 Operating Systems: Configuration and Use 3:0

(Common to CSE and IT, C34, T25)

Preamble: In the BE Computer Science and BE Information Technology program, the Course Operating Systems: Configuration and Use is offered as a common subject. One of the main assumptions made is that the course on "configuring and Using Operating Systems" would be studied from the viewpoint of the user-stakeholder and from a practical perspective of using the operating system.

In a course like this it is necessary to decide the "vehicle" OS through which the course will be conducted. In view of the growing base and popularity of the Open Source Linux, it is proposed to use it heavily. Windows XP Professional could be a viable choice, but the course could become dated when the Windows 7 OS is released shortly.

Program Outcomes addressed

b. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

d. Graduates will demonstrate an ability to design a system, component or process as per needs and specifications.

f. Graduate will demonstrate skills to use modern engineering tools, softwares and equipment to analyze problems.

k. Graduate who can participate and succeed in competitive examinations.

Competencies

1. Explain why the leading open source operating system Linux is powerful and popular in the community of users
2. Make an informed choice about the particular implementation, version, etc. for installation and use in a home or business setting
3. Install the Linux OS and make an initial configuration usable
4. Navigate GNOME and KDE to make the initial settings of the OS
5. Perform the various tasks of user management such as creating shell accounts, user accounts, restricted user accounts, software accounts, group user accounts, etc.
6. Create Linux file systems
7. Perform the various tasks of file system management such as configuring permissions, umask, groups, backing and restoring file systems, and navigating the file systems
8. Perform the tasks of package management dealing with compressed files
9. Perform GUI management, configure the GUI components, X server, etc.

10. Perform the various tasks of network management such as setting up IP addresses, configuring a domain, determining & applying subnets, configuring routing & masquerading, and setting up daemons
11. Configure and use Domain Name Services
12. Configure and make available the Internet mail services
13. Configure the Apache Web server for use
14. Setting up other services such as FTP, etc.
15. Use text processing tools such as vi and emacs
16. Write simple shell scripts to perform typical administrative tasks and simple applications

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test3/End semester examination
1	Remember	20	20	20
2	Understand	30	40	30
3	Apply	10	10	10
4	Analyze	20	10	10
5	Evaluate	10	10	20
6	Create	10	10	10

Course Level Learning Objectives:

Remember

1. List the various Linux Distributions
2. Which is the first Linux Distribution?
3. List the Linux Distributions Built on RedHat Linux Kernel
4. List the Linux Distributions Built on Debian Linux Kernal
5. What are the configuration tools used in RedHat Linux and SuSeLinux ?
6. Which is the most popular Linux distribution in the world?
7. List the Linux partitions for Linux Installation
8. Define a swap partition
9. What is meant by a Shell?
10. List the alternate web servers available for Linux
11. List Xwindow system commands and Configuration Files
12. List the Database Management Systems for Linux

13. Write the methods of managing user environments
14. List the file system Hierarchy Standards
15. What are the journaling file system types?
16. What are the entries in fstab files?
17. List the GNOME office applications
18. What is SQL Anywhere?

Understand

1. How to install Linux in a computer where Windows is already installed?
2. How to use the free space on the Hard Disk for Linux Installation?
3. Describe the fdiskutility ?
4. Explain the reasons for choosing the MBR/Boot sector/floppy disk for LILO
5. How is Authentication ensured when starting Linux?
6. How to run jobs in the background?
7. How chkconfig works?
8. How to login using anonymous FTP login?
9. Explain the available file system types
10. Explain the /proc File System
11. Explain about the Emacs and Vi Editors
12. Explain the GUI User Management tools?
13. How to Run MS Office on Linux?

Apply

1. Configure the Graphical system using XFree86 graphical software
2. Reconfigure the graphical system using command line
3. Configure the Apache web server
4. Configure the client browsers
5. Perform the postfix configuration
6. Perform the sendmail configuration
7. Set up and manage the MySQL database servers
8. Configure the C Shell and TCSH shell
9. Configure the Xorg Server
10. Upload an Image using F-Spot

Analyze

1. Differentiate the IDE and SCSI Interface partitions
2. Write the Disk Druid utility in Red Hat Linux

3. Analyze the cause, if Linux installation program does not start
4. Analyze the details of setting up the Linux Graphical Interface
5. Analyze the tasks performed on the system using syslog files
6. Analyze the changes made when Creating or deleting the new /etc/fstab entries
7. Analyze the mounting and unmounting of various devices using mount/umount commands
8. Analyze the system run levels in inittab
9. Analyze the configuration of Various Shells
10. Analyze the Special Features of Bash Shell

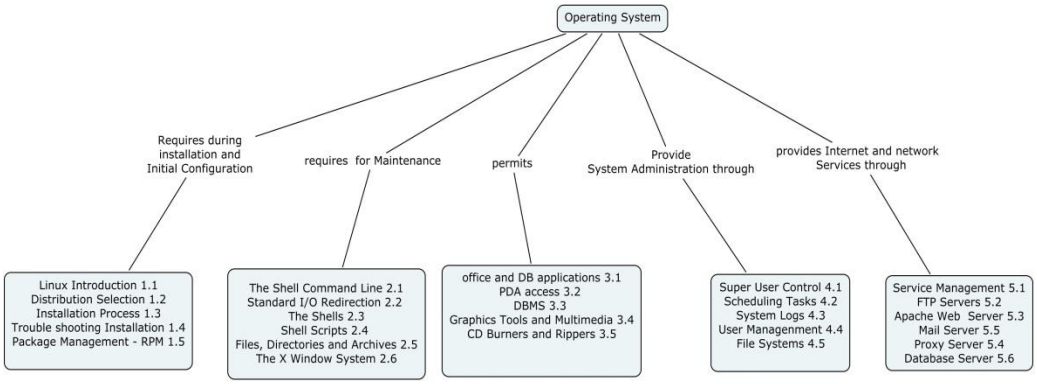
Evaluate

1. Compare the Drive letter assignments in Windows and Subdirectory mount points in Linux
2. Obtain the list of available system directories
3. Compare the Linux Desktop Editors
4. Compare the Graphics Tools for Linux
5. Compare the Multimedia and Sound Applications
6. Compare Video and DVD Projects and Applications
7. Compare the C shell and Z shell
8. Compare the Various Display Managers
9. Evaluate the Local Proxy Server Configuration
10. Compare the different Database Management Systems available on Linux

Create

1. Give the example mount points for a multiple partition installation
2. Create the boot disk to use when installing Linux
3. Create the default grub entry to Linux Operating system by editing the grub.conf file
4. Create file systems using various utilities
5. Create a new scheduled task using cron utilities
6. list the steps to install and uninstall RPM packages
7. Install software applications from compressed archive
8. Perform power and arithmetic operations using functions in openoffice.calc
9. Generate a chart in openoffice.writer using data in aopenoffice.calc created by any equation.
10. Create a presentation about the college using Openoffice.impress
11. Create a New Image using GIMP

Concept Map:



Course Contents:

1. Linux Installation

1.1 Introduction to Linux

1.2 Linux distributions

1.3 The Installation Process

1.4 Troubleshooting Installation

1.5 Software package Management with RPM

2. Linux Shell, Files and Xwindow

2.1 Shell command Line

2.2 Standard I/O redirection

2.3 The Shells

2.4 Shell scripts

2.5 Linux Files and Directories

2.6 X window system

3 Linux Software Management

3.1 Office and Database Applications

3.2 PDA access

3.3 Database Management Systems

3.4 Graphics tools and Multimedia

3.5 CD Burners and rippers

4 System Administration

4.1 Super user control

4.2 Scheduling tasks

4.3 System logs

4.4 User management

4.5 File systems

5 Internet and Network Services

5.1 Managing services

5.2 FTP Servers

5.3 Apache Web server

5.4 Proxy server

5.5 Mail server

5.6 Database servers

Syllabus

Linux Installation: Introduction to Linux , Linux distributions , The Installation Process , Troubleshooting Installation ,Software package Management with RPM

Linux Shell, Files and Xwindow: The Shell command line , Standard I/O redirection , The various shells, Shell scripts , Linux Files Directories and Archives , The X window system **Linux Software Management**: Office and Database Applications , PDA access , DBMS , Graphics tools and Multimedia , CD Burners and rippers **System Administration**: Super user control , Scheduling tasks , System logs , User Management, File systems, **Internet and Network Services** Managing services , FTP Servers , Apache Web server, Proxy server , Mail Server , Database servers.

Course Textbooks:

1. Richard Petersen, The Complete Reference Linux Sixth Edition, Tata McGraw-Hill, 2008
2. Nicholas Wells, Guide to Linux Installation and Administration, Vikas Publishing House, 2000

Reference Book :

1. Matt Welsh, Matthias KalleDalheimer, Terry Dawson, Lar Kaufman ”Running Linux”, Fourth Edition, December 2002

Lecture Schedule:

No	Topic	No. of Lectures
1	**Linux Installation**	
1.1	Introduction to Linux	1
1.2	Linux distributions	1
1.3	The Installation Process	2
1.4	Troubleshooting Installation	1

1.5	Software package Management with RPM	1
2	**Linux Shell, Files and Xwindow**	
2.1	The Shell command line	1
2.2	Standard I/O redirection	1
2.3	The Shells	2
2.4	Shell scripts	2
2.5	Linux Files Directories and Archives	1
2.6	The X window system	1
3	**Linux Software Management**	
3.1	Office and Database Applications	2
3.2	PDA access	1
3.3	DBMS	2
3.4	Graphics tools and Multimedia	1
3.5	CD Burners and rippers	2
4	**System Administration**	
4.1	Super user control	2
4.2	Scheduling tasks	1
4.3	System logs	2
4.4	User management	1
4.5	File systems	2
5	**Internet and Network Services**	
5.1	Managing services	2
5.2	FTP Servers	1
5.3	Apache Web server	1
5.4	Proxy server	2
5.5	Mail server	2
5.6	Database servers	2
	Total No of Hours	40

Course Designers:

1. K. Sundarakanthamkskcse@tce.edu
2. T. Subbulakshmisubbulakshmi@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T26	2	-	-	2

T26 Environment and Ecology **2:0**

(Common to all branches of Engineering B46, D26, E26, G36, C26)

Preamble: Environmental science is a subject carries too much of importance in the modern world.The basic knowledge on environment is an absolute necessity for all citizens. This course work aims at imparting the fundamental knowledge on environmental science which includes natural resources and their exploitation, Ecosystem, Biodiversity and its conversation Environmental pollution.

Programme outcomes addressed

- Graduates will demonstrate understanding of the societal, health, safety, legal and cultural issues and consequent responsibilities relevant to engineering practice.
- Graduates will understand the impact of engineering solutions in the societal contest and demonstrate knowledge of and need for sustainable development.
- Graduate will understand and commit to professional ethics and responsibilities and norms of engineering practice.

Competencies:

At the end of the course the student should be able to

(i) Understand the significance of conversation of natural reasources.

(ii) Realize that who is also one of the components of the ecosystem like any other species.

(iii) Appreciate the values of biodiversity and necessity of its conversation.

(iv) Find out the various sources, effects and control measures of environmental pollution.

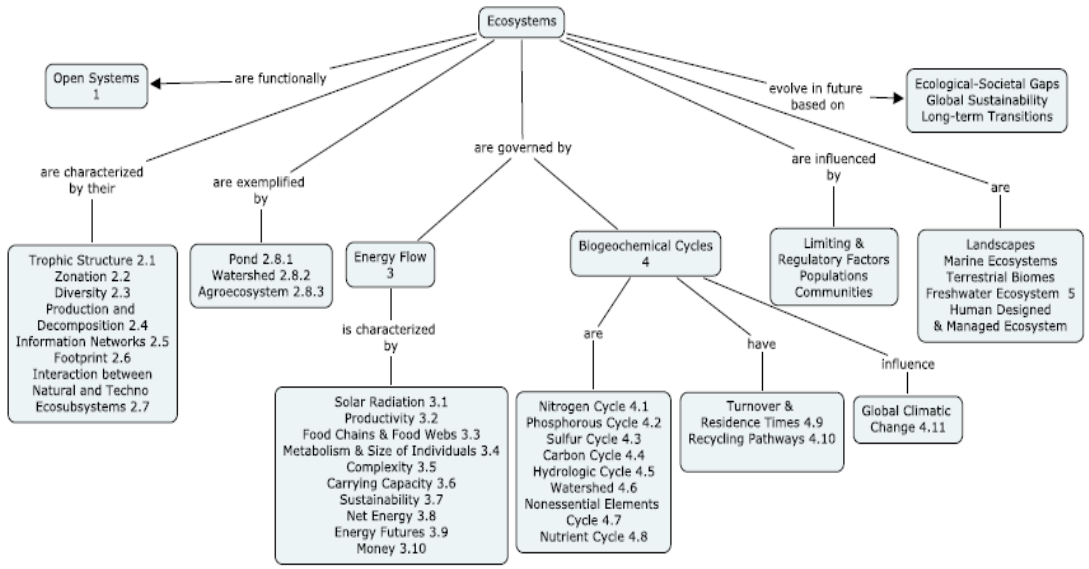
(v) Deliver the best individually towards environment sustainability.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	End-semester examination
1	Remember	30	20	20
2	Understand	40	40	40

3	Apply	30	40	40
4	Analyze	-	-	-
5	Evaluate	-	-	-
6	Create	-	-	-

Concept Map



Course contents and Lecture Schedule

S.No.	Topics	Periods
1	**Natural Resources**	
1.0	Multidisciplinary nature of Environmental studies & public awareness	1
1.1	Forest – uses, over exploitation, deforestation	1
1.2	Water – uses, over utilization, floods,droughts	**1**
1.3	Food – agriculture overgrazing, modern agriculture. Fertilizer – pesticide problems , water logging and	1

S.No.	Topics	Periods
	salinity	
1.4	Energy – renewable , non renewable energy, use of alternate energy	1
1.5	Land – degradation, land slides, soil erosion, desertification.	1
2	**Ecosystem**	1
2.1	Ecosystem – Concept , structure, function, components	1
2.2	Laws of Ecology	1
2.3	Energy flow in ecosystem – Food chains,food webs	1
2.4	Ecological pyramids – Ecological succession	1
2.5	Types of ecosystem – Forest grass land, desert, aquatic ecosystem	1
2.6	Loss of Ecosystem and its Estimation	1
3	**Biodiversity and its conservation**	
3.1	Types of Biodiversity, biogeographical and classification	1
3.2	Values of Biodiversity	1
3.3	Hotspots of Biodiversity and Threats to Biodiversity	1
3.4	Biodiversity indices	1
3.5	Endangered and endemic species-conservation of	1

S.No.	Topics	Periods
	biodiversity	
4	**Environmental pollution and control**	
4.1	Causes ,effects and control measures of air pollution	1
4.2	Causes ,effects and control measures of water pollution	1
4.3	Causes ,effects and control measures of soil pollution	1
4.4	Causes ,effects and control measures of noise pollution	1
4.5	Solid Waste Management – Causes effects and control measures	1
4.6	Water conservation – Rainwater harvesting – Global Warming	1
4.6.1	Climate change and its effect on environment	1
4.7	Ozone layer depletion – legal provisions	1

Syllabus

General - Multidisciplinary nature of environmental studies –need for public awareness. Natural reasorces – Forest – uses, over exploitation deforestation and their effects. Water – uses, over utilization , floods , droughts. Food – architecture overgrazing, modern agriculture. Fertilizer – pesticide problems, water logging and salinity. Energy – growing needs , renewable , non renewable energy, use of alternate energy. Land – degradation land slides ,soil erosion ,desertification.

Ecosystem - Concept, structure , function, components. Laws of Ecology, Energy flow in ecesytem. –Food chains, food webs Ecological pyramids – Ecological sucession. Types of ecosystem – Forest grass land,desert,aquatic ecosyatem, Loss of ecosystem and its Estimation.

Biodiversity and its conversation – biodiversity types, biogeographical classification. Values of biodiversity – Hot spots of biodiversity – threats to biodiversity – Biodiversity Indices – Endangered and endemic species – conservation of biodiversity.

Environmental pollution – Causes , effects and control measurements of air pollution – water pollution – soil pollution – Noise pollution. Solid waste management – causes effects and control measures.

Water conservation - Rainwater harvesting – global warming – Climate Change and its effect on Environment – ozone layer depletion – legal provisions.

Text Book

1. Kaushik and Kaushik, 'Environmental Science and Engineering', 2nd edition, New age International publishers, New Delhi, 2006

Reference Books

1. Wright and Nebel, 'Environmental science towards a sustainable future', Prentice Hall of India Ltdd, 2000.

2. S.K. Garg and Garg, 'Ecological and Environmental studies', khanna publishers, Delhi, 2006.

3. Gillbert M.Masters, 'Introduction to Environmental Engineering and Science',Second edition, Pearson Education publication,Delhi,2004.

Course Designers

1. T. Velrajan tvciv@tce.edu

2. Mr.S.Chandran schandran@tce.edu

Subject Code	Lecture	Tutorial	Practical	Credit
T27	**0**	**0**	**1**	**1**

T27 Operating System Lab 0:1

List of Experiments

Part I : Linux Essentials

1. Installation of Linux Operating Systems and Run Level Configuration.
2. Implement the commonly used Linux commands like cal, passwd,history and ls.
3. Explore the Linux filesystem and identify First level directories and symbolic links.
4. Use the umask command to display current default permissions of files and assign the following permission to your file.
   - all - r/w to owner, group, and others.
   - group - r/w to owner and group, r/o to others.
   - owner - r/w to owner, r/o to group, none to others.
5. Design a poster using GIMP
6. OpenOffice.org
   a. Create your resume using openoffice.org writer
   b. Creation of a presentation using openoffice.org Impress.
   c. Schedule the tasks for accomplishing a miniproject using Project Management Tool.
   d. Draw charts using excel data.
7. Pipes, Filters and IO Redirection
   a. Execute two commands simultaneously.
   b. Use cut,paste,sort,grep and awk utilities to manipulate a file.
   c. Change the default input, output and error files.
8. Text Editor

   Create a file classmates.dat using vi and emacs editor and perform the following tasks using commands
   a. Use command mode to make changes in the file.
   b. Move around in the document .
   c. Move to the top or bottom of the file.
   d. Making changes in words or lines.
   e. Copying or moving lines to other locations in a file.

Part II: Shell Script

1. Write shell script to show various system configuration like
   Currently logged user and his logname
   current shell
   home directory
   Operating system type
   Current path setting
2. Write a shell script to search the etc/passwd file for a specified user. If the users name is found in /etc/passwd, print a message that the user exists, else print an error message that the user does not exist.
3. Write a shell script to change a text string (each character passed as a shell parameter) into a telephone number. Test each parameter for legality (A-Z). if parameter is invalid, print an appropriate error message and exit the script; Otherwise, process each parameter and print the resultant telephone number. Assign the numerical to Character in the following manner:0=ABC, 1=DEF, 2=GHI, 3=JKL, 4=MNO, 5=PQR, 6=STU, 7=VWX, 8=YZ
4. Write a shell script to list all files in the current directory ending in .c. For each file found, print the name of the file.
5. Write a script to move a list of files in the current directory to another directory. How can you move all your files to another directory?
6. Write a shell program that sends a note to the several people on your system. Include the following information in the note:
   a. lines at the top that include the current date and the words ``Dear friends''
   b. the body of the note (stored in an existing file)
   c. a closing statement
7. Write a shell script that is based on chmod, but uses the keywords add and remove rather than + and - for permission rights. The keywords execute, read, and write are used instead of rwx. Its format would be similar to
   mychmod add execute file-directory-name
8. Write shell script to convert file names from UPPERCASE to lowercase file names or vice versa
9. Write a shell script named dirrep.sh that collects the following information about a directory.

Count the total number of directories that are in the given directory (not that these are subdirectories).

Count the total number of files in the given directory.

Count the number of items (files/directories) in the current directory that are readable.

Count the number of items (files/directories) in the current directory that are writable.

Count the number of items (files/directories) in the current directory that are executable.

10. Write a C program that makes use of shared libraries.
11. Implement the commonly used linux commands like cp,mv,rm,ls and date using C language.
12. Create an addressbook program using the shell script. It should use functions to perform the following tasks.

    Search address book .

    Add entries .

    Remove / edit entries and

    Display.

Course Designers:

1. S.Srinivasan ssnit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T28	0	0	1	1

T28 Computer Programming Lab 0:1

Preamble:

The Computer Programming lab using C under windows and Linux environmnet aims at providing hands on for Computer Programming –I. Students will gain a practical insight in to structured programming concepts and improve their problem solving and programming skills.

List of Experiments

1. Simple Programs
    a. Generation of Fibonacci Series
    b. Calculation of factorial of number
    c. Computation of Sum of set of numbers
    d. Generation of prime numbers
2. Sorting of Numbers and Strings
3. Matrix Manipulations
    a. Addition of two matrices
    b. Subtraction of two matrices
    c. Multiplication of two matrices
    d. Determinant Calculation
4. String Manipulations
    a. Encoding and Decoding
    b. Case Conversions
    c. String Reversal, Substring Extraction, String Concatenation
    d. Palindrome Checking
5. Pointers
    a. Pointers to Strings
    b. Array of Pointers
6. Matrix Manipulation using Dynamic Memory Allocation
7. Program Development using Functions
    a. Illustration of call by value and call by Reference
    b.Recursive Functions
8. Macro Expansion with Conditional Compilation and Multiple File Inclusions

9. Structures
    a. Program Development involving Array of Structures
    b. Passing Structures to Functions and returning Structure variables from Functions
10. File Operations
11. Manipulation of Structures with Files.
12. Implementation of Stack

Course Designers:

1. C.Jeyamala jeyamala@tce.edu
2. P.Karthikeyankarthikit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T29	0	0	1	1

T29 Work Shop **0:1**

(Common to all branches of Engineering B29, C29, D29, E29, G29, T29)

Objective: The students of all branches of engineering would get exposure to basic practices in a mechanical workshop. The students get trained to acquire skills at basic level in fitting, carpentry, joining, metal forming and plumbing.

List of Exercises

I Fitting (Any four exercises)

1. Fitting tools and practice
2. Joining of two different metals with adhesives
3. Preparation of single step joint
4. Preparation of 'V' joint
5. Preparation of Gauge joint
6. Preparation of Taper sep joint

II Carpentry (Any four exercises)

1. Carpentry tools and practice
2. Joining different types of wood with adhesives
3. Preparation of Half joint
4. Preparation of Dovetail joint
5. Preparation of T-brittle joint
6. Turning on wood lathe

III Demonstration on Tools and Practice (Any four exercises)

1. Welding
2. Soldering
3. Brazing
4. Foundry and Moulding practice
5. Smithy forging
6. Plumbing
7. House Wiring
8. Press work

Terminal Examination: Students are tested in fitting and carpentry trades

REVISED CURRICULUM AND DETAILED SYLLABI

FOR

B.Tech DEGREE (Information Technology) PROGRAM

THIRD SEMESTER

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2012-2013

THIAGARAJARCOLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to AnnaUniversity)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Approved in BOS Meeting on 13.10.2012 Approved in 45th Academic Council Meeting on 24.11.2012

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015

B.Tech. Degree (Information Technology) Program

SUBJECTS OF STUDY

(For the candidates admitted from 2012-2013 onwards)

THIRD SEMESTER

Subject code	Name of the subject	Category	No. of Hours / Week	credits
			L T P	
THEORY				
T31	Engineering Mathematics III	BS	4 - -	4
T32	Object Oriented Programming	DC	3 - -	3
T33	Computer Programming -II	DC	3 - -	3
T34	Data Structures	DC	3 - -	3
T35	Data Bases: Practice	DC	3 - -	3
T36	Computer Organization	DC	3 - -	3
PRACTICAL				
T37	Object Oriented Programming Lab	DC	- - 3	1
T38	Data Bases Lab	DC	- - 3	1
T39	Data Structures Lab	DC	- - 3	1
Total			**19 - 9**	**22**

BS : Basic Science
HSS : Humanities and Social Science
ES : Engineering Science
DC : Department Core

L : Lecture
T : Tutorial
P : Practical

Note:

1 Hour Lecture/Tutorial is equivalent to 1 credit
2/3 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015

B.Tech. Degree (Information Technology) Program

SCHEME OF EXAMINATIONS

(For the candidates admitted from 2012-2013 onwards)

THIRD SEMESTER

S.No.	Sub. code	Name of the subject	Duration of Terminal Exam. in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	T31	Engineering Mathematics III	3	50	50	100	25	50
2	T32	Object Oriented Programming	3	50	50	100	25	50
3	T33	Computer Programming - II	3	50	50	100	25	50
4	T34	Data Structures	3	50	50	100	25	50
5	T35	Data Bases: Practice	3	50	50	100	25	50
6	T36	Computer Organization	3	50	50	100	25	50
PRACTICAL								
7	T37	Object Oriented Programming Lab	3	50	50	100	25	50
8	T38	Data Bases Lab	3	50	50	100	25	50
9	T39	Data Structures Lab	3	50	50	100	25	50

* Continuous Assessment evaluation pattern will differ from subject to subject and for different tests. This will have to be declared in advance to students. The department will put a process in place to ensure that the actual test paper follow the declared pattern.

** Terminal Examination will be conducted for maximum marks of 100 and subsequently be reduced to 50 marks for the award of terminal examination marks.

Sub Code	Lectures	Tutorial	Practical	Credit
T 31	4	0	--	4

T31 Engineering Mathematics III 4:0

(Common to all branches of Engineering, B31,C31,D31,E31,G31,T31)

Preamble:

An engineering student needs to have some basic mathematical tools and techniques. This emphasizes the development of rigorous logical thinking and analytical skills of the student and appraises him the complete procedure for solving different kinds of problems that occur in engineering. Based on this the course aims at giving the adequate exposure in the theory and applications of Fourier series, Fourier Transforms, PDE's and BVP

Program Outcomes addressed

a. Graduates will demonstrate knowledge of mathematics, science and engineering.

b. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

j. Graduate will develop confidence for self-education and ability for life-long learning.

Competencies

At the end of the course the student should be able to

1. Express the periodic functions arising in the study of engineering problems as Fourier series of Sines and Cosines.
2. Find the Fourier series for the typical waveforms.
3. Find the Fourier series for discrete data using Harmonic Analysis.
4. To study some of the well-known integral transforms (like Fourier, Fourier Sine and Cosine) and properties.
5. Formulate simple Engineering problems as Partial Differential Equations and state the boundary conditions.
6. Solve Partial Differential Equations, linear, nonlinear, homogeneous and non-homogeneous, by various methods.
7. Solve the standard Partial Differential Equations arising in engineering problems like Wave equation, Heat flow equation (one dimensional and two dimensional, Cartesian and polar coordinates) by Fourier series.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3/End-semester examination
1	Remember	10	10	0
2	Understand	30	30	30
3	Apply	60	60	70
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	0	0	0

Course Level Learning Objectives

Remember

1. Define Periodic function.
2. Show that f(x)= x^3 is an odd function.
3. State the Fourier Series for the function f(x) in the interval (1, 3).
4. Identify the Kernel for Fourier Cosine and Sine Transforms.
5. State Parseval's Identity.
6. State Convolution Theorem.

Understand

1. Distinguish between Odd and Even functions.
2. Use the Fourier series expansion of $x^2 = \frac{\pi^2}{3} + 4\sum_1^\infty (-1)^n \frac{\cos nx}{n^2}$, $-\pi < x < \pi$ to predict the value of $\sum \frac{1}{n^2}$.
3. Discuss harmonic analysis.
4. Discuss Fourier Series in Complex form.
5. Interpret the result $F[f(ax)] = \frac{1}{a} F\left(\frac{s}{a}\right)$.
6. Interpret the usage of Parseval's theorem.
7. Discuss the two methods of forming partial differential equations.
8. Discuss the solution of $\frac{\partial^2 z}{\partial x \partial y} = x^2 y$ by direct integration.
9. Discuss the working rule of solving the Lagrange's linear equation.
10. Discuss the working rule of solving f(p,q)=0.
11. Discuss the working rule of solving f(z,p,q)=0.

Apply

1. Find the Fourier transform of $e^{-a^2x^2}$. Hence prove that $e^{-\frac{x^2}{2}}$ is self-reciprocal with respect to Fourier transforms and (i) Find the Fourier Cosine transform of e^{-x^2} .
2. Solve the equation $z^2(p^2+q^2+1)=c^2$ where c is a constant.
3. Obtain the first three harmonics in the Fourier series expansion in (0,12) for the function y=f(x) defined by the table given below:

X:	0	1	2	3	4	5	6	7	8	9	10	11
Y:	1.8	1.1	0.3	0.1	0.5	1.5	2.16	1.88	1.25	1.30	1.76	2

4. Find the Fourier transform of f(x), if $f(x)=\{1-|x|, for|x|<1 and 0, for|x|>1\}$.

Syllabus

Fourier Series: Dirichlet's conditions, General Fourier Series, Half range Sine and Cosine series, Parseval's Identity, Harmonic Analysis, Complex form of Fourier Series.**Fourier Transformation:** Fourier Integral Theorem, Fourier Transform, Fourier Sine and Cosine Transforms, Convolution Theorem, properties, Parseval's Identity, Discrete Fourier Transform, Discrete Time Fourier Transform,Demonstration of Fourier transforms and its properties using MATLAB (Tutorial).**Partial Differential Equations**: Formation, Solution of standard types of first order equations, Lagrange's linear equation, Linear partial differential equations of second and higher order with constant coefficient. **Boundary Value Problem**: Classification of Second Order linear partial differential equations, One-dimensional Wave equation, One dimensional heat equation, Solution by Fourier Series, Steady State Solution of two dimensional heat equation in Cartesian Co-ordinates, Laplace equation in Polar Co-ordinates, Solution by Fourier Series method.

Text Book

B.S. Grewal: Higher Engineering Mathematics, 39thEdn. , Khanna Publishers, New Delhi, 2007.

References

1. Lecture Notes by the faculty of Department of Mathematics, TCE ,Madurai.

2. Veerarajan .T: Engineering Mathematics, 3rd Edition. , Tata McGraw Hill, NewDelhi, 2004

3. Kreyszig,E., "Advanced Engineering Mathematics", John wiley and sons,(Asia)Pte Ltd., Singapore.2006.

Course contents and Lecture Schedule

No.	Topic	No. of Lectures
1.	**Fourier Series**	
1.1	Dirichlet's conditions, General Fourier Series	3
1.2	Half range Sine and Cosine series	3
1.3	Parseval's Identity	1
1.4	Harmonic Analysis	2
1.5	Complex form of Fourier Series	2
2.	**Fourier Transformation**	
2.1	Fourier Integral Theorem, Fourier Transform	2
2.2	Fourier Sine and Cosine Transforms	2
2.3	Convolution Theorem	1
2.4	Properties, Parseval's Identity	2
2.5	Discrete Fourier Transform, Discrete time Fourier Transform	2
2.6	Demonstration of Fourier transforms and its properties using MATLAB (Tutorial)	3
3	**Partial Differential Equations**	
3.1	Formation	2
3.2	Solution of standard types of first order equations	3
3.3	Lagrange's linear equation	2
3.4	Linear partial differential equations of second and higher order with constant coefficient	3
4	**Boundary Value Problems**	
4.1	Classification of Second Order linear partial differential equations	1

4.2	One-dimensional Wave equation, Solution by Fourier Series	4
4.3	One dimensional heat equation, Solution by Fourier Series	4
4.4	Steady State Solution of two dimensional heat equation in Cartesian Co-ordinates, Solution by Fourier Series	4
4.5	Laplace equation in Polar Co-ordinates, Solution by Fourier Series	4

Course Designers

1. V.Mohanvmohan@tce.edu
2. N. Kannannkmat@tce.edu
3. N. Chitrancmat@tce.edu
4. M.SivanandhaSaraswathysivanandha@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T 32	3	0	--	3

T32 Object Oriented Programming 3:0

Preamble: In the object-oriented view of software, programs are considered to be collections of objects that interact by sending messages to one another and reacting to the answers to those messages. These ideas are at the forefront of modern software development. The course takes an 'objects first', approach to teaching:

- The fundamental concepts associated with object-oriented programming (object, class, hierarchy, inheritance, encapsulation, polymorphism, collaboration, etc.)
- To apply and extend the knowledge of programming concepts within an object-oriented programming context.

This course provides a comprehensive introduction to the object-oriented approach through the widely-used Java programming language. In the process, this course will cover how to create OO applications from start to finish, following correct syntax, coding style, and avoiding common errors. Concentrating on aspects of Java that best demonstrate object-oriented principles and good practice, the student will gain a solid basis for further study of the object-oriented software development. Lectures will cover selected predefined Java classes and methods, and how to create user-defined classes, methods, and packages. Programming projects will focus on practicing OOP.

Program Outcomes addressed

a. An ability to design a system or component, or process to meet stated specifications

b. An ability to identify, formulate and solve engineering problems

c. An ability to design and conduct experiments, analyze and interpret data.

Competencies

1. Understand the differences between a structured and object-oriented programming, and the major benefits of object-oriented programming.
2. Comprehend and implement the basic constructs of OOP including abstraction, objects and classes, polymorphism, and inheritance.
3. Understand the OOP concept involved in inter-process communication and remote method invocation.
4. Implement exception handling mechanisms, packages, interfaces and multithreading.
5. Design GUI for simple applications using AWT controls.

6. Develop object-orienteddesigns for real world problems.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test3/ End semester Examination
1	Remember	20	10	0
2	Understand	30	30	20
3	Apply	50	40	40
4	Analyze	0	0	0
5	Evaluate	0	0	10
6	Create	0	20	30

Learning Objectives

Remember

1. List down the striking features of object oriented programming.
2. List any five event classes with definition.
3. List any 4 methods of InputStreamclass?
4. List any 4 methods of Writer class?
5. What are the various *awt* classes associated with controls?

Understand

1. Write short notes on constructors and destructors?
2. Differentiate static and dynamic binding?
3. Write about final and abstract keywords in Java?
4. Explain briefly about method overriding?
5. Differentiate Exception and Error?
6. Write about the Thread Priorities with example?
7. Write short notes on byte streams and character streams?
8. What is the difference between method overloading and overriding?
9. How will you handle an exception?
10. What do you mean by stub and skeleton?

Apply

1. Write a Java program to implement the multithreading concept?
2. Write a Java program to implement a single inheritance in interface?
3. Write a java program to create a new exception class for voters' eligibility?
4. Write a java program to implement synchronization concept in threads?
5. Write a RMI program to implement a distributed application for subtracting two numbers?

6. Create a package with interface for students mark system and import the package to find the results
7. Write a Java program to implement the following concepts for Bank Transactions.
    a. Implement the hierarchical inheritance.
       Super class: Bank - get the basic inputs.
       Sub class1: FixedDeposit – Calculate the interests
       Sub class2: Transactions –Deposit and Withdraw
    b. Implement method Overloading (interest calculation based on the balance amount)
    c. Include the package and Interface concepts.
8. Write a Java program to implement the following concepts for Employee Payroll.
    - Implement single inheritance with method overriding.
    - Calculate the Dearness Pay, Dearness Allowance and House Rent Allowance for the basic pay which is more than Rs.5000/=
    - Overload the binary + operator for calculating the salary.
    - Create a new exception class for checking the basic pay which is less than Rs.5000/=
9. Write a Java program to implement the following concepts for Railway Ticket Reservation System.
    - Create two packages namely Train and Passenger.
    - In Train package using single inheritance get the train details like TrainNo, TrainName, ArrivalTime, DepartureTime, Source, Destination and AvailableSeats.
    - In Passenger package, get the details of the passenger who are going to book the ticket based on the availability. Get the input like Name, Age, Address, Date of Journey, TrainNo.
    - Import these two packages in common a class and Show the list of train details as well as passenger details in date of journey order.
    - Create an exception class for checking the age eligibility.
10. Write a Java program to implement the following concepts for a Book Shop.
    - Implement the hierarchical inheritance.
        - Super class: Book – Get the book details (Book_id, Title, Author, Publishers, Price, Category and Available copies).
        - Sub class: Sales – Get the details (Date, Book_id, Title and No. of Copies).

-
  - Sub class: Stock – Get the detail (Book_id, Remaining copies)
- Display the Stock details based on the date.
- Display the Sales details based on the date.
- Use super, final and abstract keywords where ever it is possible.
- Create an exception class for the no of copies sales exceeds the available copies.

Evaluate

1. Evaluate the Structured approach and Object Oriented Systems Development approaches for developing a "Railway Reservation System" based on the factors like modularity, re-usability, development time, debugging cost, scalability and ability to support abstraction. Suggest suitable scenarios under which each of them may be more suitable than the other?
2. Compare and contrast the characteristics exhibited by the procedure and object oriented programming based on the factors like modularity, re-usabilty, development time, debugging cost, scalability and ability to support abstraction. Suggest suitable scenarios under which each of them may be more suitable than the other?

Create

1. Create a base class called shape Use this class store two double type value that could be used to compute the area of figure. Derive two specific classes called triangle and rectangle from the base shape. Add to the base class, as a member function get_data() to initialize base class data members and another member function display_area() to compute and display the area of figures. Using these three classes design a program that will accept dimensions of a triangle or rectangle interactively and display the area. The two input values will be treated as lengths of two sides in the case of rectangle and as base and height in the case of triangle and used as follows:

   Area of rectangle = x * y

   Area of triangle = ½ * x * y
2. Consider the case of "Library management automation System". Provide the details on the following perspectives:
   i. Classes and Roles.
   ii. Abstract Classes.
   iii. Functionalities involved.
   iv. Hierarchy implementation.
3. Consider the case of "Banking process". Provide the details on the following perspectives:

4.

i. Classes and Roles.
ii. Functionalities involved.
iii. Hierarchy implementation.

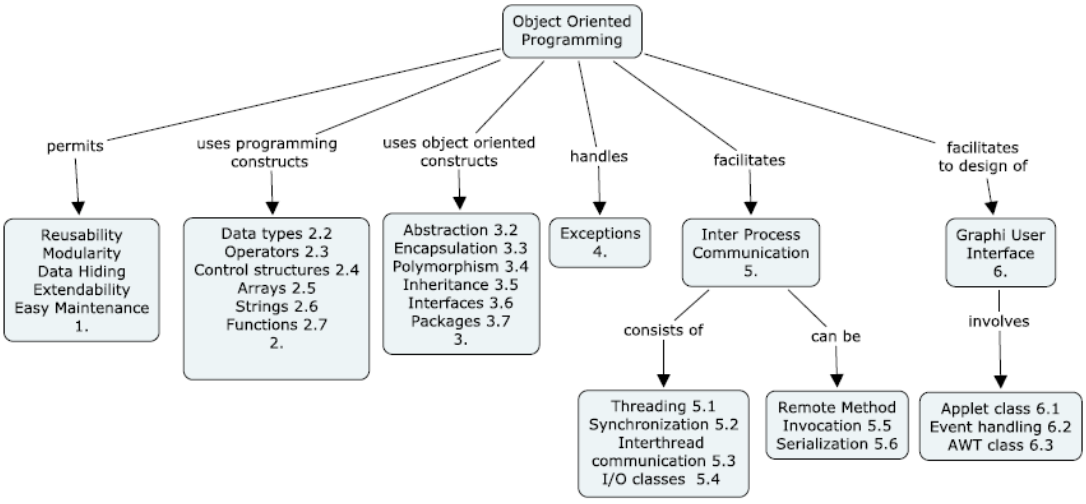
5. An electricity board charges the following rates to domestic users to discourage large consumption of energy.

For the first 100 units	Rs.1.50 per unit
For the next 200 units	Rs. 3.00 per unit
Beyond 300 units	Rs.5.00 per unit.

All users are charged a minimum of Rs.100. If the total cost exceeds Rs.250/=, then an additional surcharge of 15% is added. Write a program to read the names of users and number of units consumed and print out the charges with names.

5. Create two packages of a student. One package contains the academic details of a student and the other one contains the extracurricular activities, sports etc. Link these two packages and display all the details of a student.

Concept Map



Course Contents and Lectures schedule

1	**Introduction to Object Oriented Programming** (Reusability, modularity, data hiding, extendibility and easy maintenance	2
2.	**Programming Constructs**	
2.1	What are programming constructs	1
2.2	Data types	
2.3	Operators	

2.4	Control structures	1
2.5	Arrays	1
2.6	Strings	1
2.7	Functions	1
3	**Object Oriented Constructs**	
3.1	Object oriented paradigm	1
3.2	Abstraction	1
3.3	Encapsulation	1
3.4	Polymorphism	1
3.5	Inheritance	2
3.6	Interface	2
3.7	Packages	2
4	**Exception Handling**	
4.1	Fundamentals of exceptions	2
4.2	Built-in Exceptions	1
4.3	Creating Exception subclasses	2
5	**Inter Process Communication**	
5.1	Creating single and multiple threads	2
5.2	Thread priorities & Synchronization	2
5.3	Interthread Communication	2
5.4	I/O classes	3
5.5	Remote Method Invocation	1
5.6	Serialization	2
6	**Graphical User Interface**	
6.1	Applet class	2
6.2	Event Handling	2
6.3	AWT controls	2

Syllabus

Features of OOP: Reusability, modularity, data hiding, extendibility and easy maintenance **OO Programming Constructs**: Data types, operators, control structures, arrays, strings and functions **Object Oriented Constructs:** Abstraction, encapsulation, polymorphism, inheritance, interface and packages**Exception Handling:** Fundamentals, built-in exceptions, creating exception subclasses **Inter Process Communication:** Creating single and multiple threads, thread priorities, interthread Communication, I/O classes, RMI, Serialization **Graphical User Interface:** Applet class, event handling, AWT controls

Text Book

1. Herbert Schildt: The complete reference – Java2, Fifth Edition, McGraw Hill Companies, 2002.
2. C.Thomas Wu: An introduction to object oriented programming with Java, Third Edition, McGraw Hill Companies, 2004.

Course designers

1. S. Geethasgeetha@tce.edu
2. P. Karthikeyankarthikit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T33	3	0	--	3

T33 Computer Programming-II 3:0

Preamble:

In this course, the students learn a language that has many practical uses in the real world. The course introduces C++ syntax and functions not found in the traditional C. The fundamental concepts of the object oriented paradigm are introduced and object oriented programming is stressed in place of traditional structured programming. Focused topics include class definitions, overloading functions and operators, single and multiple inheritance, virtual functions and abstract classes, templates, exceptions, and the standard template library. Prerequisites: a working knowledge of C, including structures and pointers, is assumed; substantial experience in C is preferred.

Program Outcomes addressed

Graduates will demonstrate

b. An ability to design and conduct experiments, as well as to analyze and interpret data.

c. An ability to design a system or component, or process to meet stated specifications

d. An ability to identify, formulate and solve engineering problems/

Competencies

The Students will be able to

1. Understand the differences between a structured and object-oriented programming, and the major benefits of object-oriented programming.
2. Explore the more extensive features like Native types and statements, functions and pointers and structures for implementing ADT.
3. Interpret C++ Object Oriented Constructs, Data hiding and member functions, Object Creation and destruction.
4. Comprehend and Implement the Object Oriented Programming including polymorphism, inheritance, abstraction, template, generic programming and exceptions, input/output, STL.
5. Design real life examples using C++.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3	End Semester Examination
1	Remember	20	20	10	10
2	Understand	30	30	30	30
3	Apply	40	40	40	40
4	Analyze	10	10	20	20
5	Evaluate	0	0	0	0
6	Create	0	0	0	0

Course Level Learning Objectives

Remember

1. Name the operators available in C++ which is not available in C.
2. Give the difference between a structure and a class.
3. What is namespace?
4. List the memory allocation operators.
5. Differentiate between constructor and destructor.
6. Write the rules for operator overloading?
7. Differentiate implicit and explicit call.
8. What is the difference between function template and class template?
9. What is meant by exception?
10. What is STL?

Understand

1. Give the structure of a C++ program.
2. Explain C++ data types and operators in C++ in detail.
3. Explain control structures available in C++ with syntax and examples.
4. Explain the different parameter passing mechanisms in C++ with example.
5. What are the different kinds of polymorphism?
6. What is a class? What are the parts of class specification
7. Explain the declaration of a class in C++, access specifiers and different ways of defining a member function in a class with example.
8. What are inline functions explain with example? What are their advantages?
9. Differentiate overloaded and overridden functions.
10. Define inheritance. State the features and advantages and syntax of inheritance.

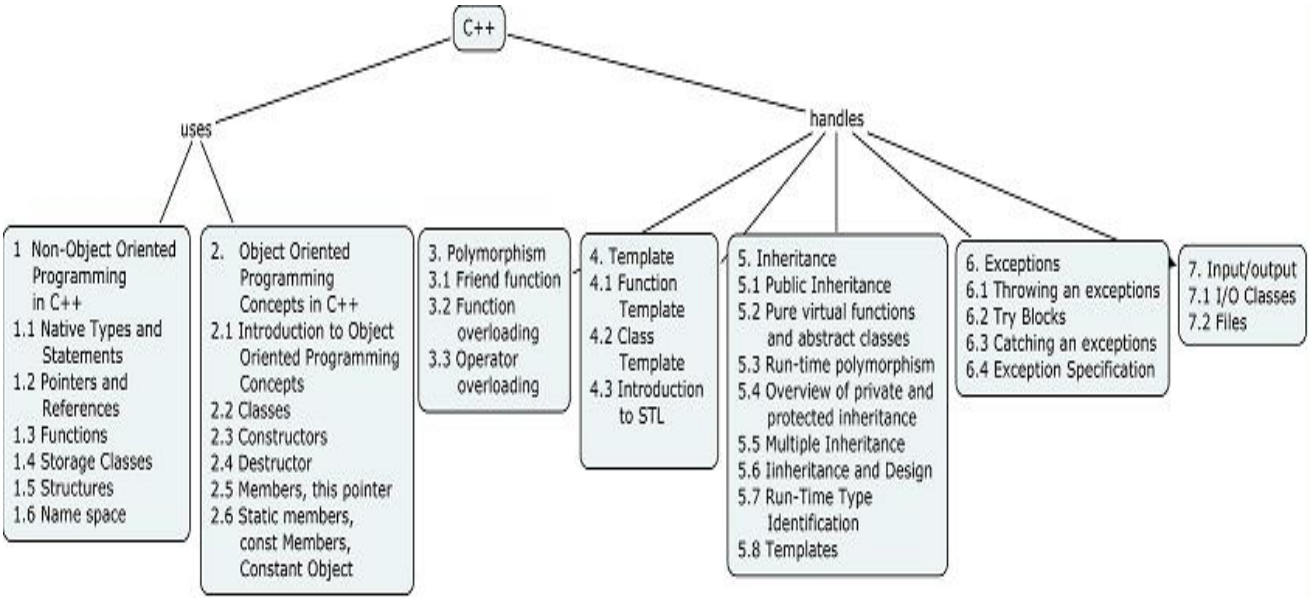
Apply

1. Write a program that performs string reversal.Assume that s1 ends up with the reverse of the string s2, and that s1 points at enough store that is adequate for reversal.
2. Write a C++ program using function template for sorting an array(assume for float array and int array)
3. Write a C++ file program to store the details of 10 students .Get inputs from user (name,m1,m2,m3).Find total & average and store it in a file called studin.dat
4. Write a C++ file program to store the details of employee in a file called emp.dat (empno, empname., deptno, salary). Read the contents of file emp.dat and if deptno =10 increment salary by 10% and store the new salary along with empname, empno and deptno in a new file called sal.dat.
5. Write a c++ program that throws an exception in one function and catches it in another.
6. Write a generic function that requires that two vector<T>s of different types be swapped. Assume that both array types have elements that are assignment-convertible.

Analyze

1. Compare parameter passing mechanism using pointers and reference as arguments. Discuss the pros and cons
2. Is it possible to mix new,delete with malloc and free while writing programs.Discuss its impact.
3. Compare inline functions with macros.Discuss the merits and demerits using both.
4. Can a C++ and C structure be treated as same.Justify
5. Can a structure in C++ consider nothing but class with members as public by default. Justify
6. Clearly differentiate
    a. Runtime polymorphism and Runtime Type identification
    b. Virtual function and virtual inheritance
7. Compare the traditional exception handling techniques with try-catch-throw. List merits and demerit
8. Differentiate specializing template functions by explicit specification with automatic invocation based on type of argument.Justify its usage.
9. How does C++ file I/O library compare with its counterpart in C.Discuss how object orientation is achieved in the former?

Concept Map



Syllabus

Enhancing knowledge of C language- Non-Object Oriented Programming Concepts in C++ - Data types, Input/Output, Casting – **Pointers and References –** Dynamic memory allocation, Use of new and delete operators**,** References. **Functions –** Parameter passing mechanism – Call by reference - Default Arguments – Overloading functions - in-lining, **Storage Classes, Structures, Name-space. Object Oriented Concept in C++:** Introduction to Object Oriented Programming Concepts **–** Classes, Constructors: Parameterized constructors, Default constructors, Copy constructors, and Conversion constructors – Destructor – Members – this pointer – Static members – const Members – Constant Object. **Polymorphism:** Friend function – function overloading – Operator overloading: Normal operators, Special operators. **Templates:** Function Template, Class Template, Introduction to STL. **Inheritance:** Public Inheritance – Run-time polymorphism – Pure virtual functions and abstract classes – Overview of private and protected inheritance – Multiple inheritance – inheritance and Design - Run-Time Type Identification – Templates and inheritance. **Exceptions:** Throwing an exception - Try Blocks –catching an exception - Exception Specification. **Input/output:** Output Class - Formatted Output - Input Class, Files.

Text Book

Ira Pohl: Oriented Programming using C++, Second Edition, Pearson Education 2001.

Reference Books

1. BjarneStourstrup: The C++ programming language, Third Edition 2001.

2. Stanley B.Lippman: C++ Primer, Third Edition 2002.

Course Contents and Lectures schedule

No.	Topic	No. of Lectures
1	**Non-Object Oriented Programming Concepts in C++**	
1.1	**Native types and statements**	
1.1.1	Data types, Input/output, Casting	1
1.2	**Pointers and References**	
1.2.1	Dynamic memory allocation, Use of new and delete operators;	1
1.2.2	References	1
1.3	Functions	
1.3.1	Parameter passing mechanism - Call by reference	1
1.3.2	Default Arguments	1
1.3.3	Overloading functions	2
1.3.4	In-lining	1
1.4	Storage Classes	1
1.5	Structures	1
1.6	Name-space	1
2	**Object Oriented Programming Concepts in C++**	
2.1	Introduction to Object Oriented Programming Concepts	1
2.2	Classes	1
2.3	Constructors	1
2.3.1	Parameterized constructors	1
2.3.2	Default constructors	1
2.3.3	Copy constructors	1
2.3.4	Conversion constructors	1
2.4	Destructor	1
2.5	Members, this pointer	1
2.6	Static members, const Members, Constant	2

No.	Topic	No. of Lectures
	Object	
3	**Polymorphism**	
3.1	Friend function	1
3.2	Function overloading	1
3.3	Operator overloading - Normal operators, Special operators	2
4	**Template**	
4.1	Function Template,	1
4.2	Class Template	1
4.3	Introduction to STL	1
5	**Inheritance**	
5.1	Public Inheritance	1
5.2	Pure virtual functions and abstract classes	2
5.3	Run-time polymorphism	1
5.4	Overview of private and protected inheritance	1
5.5	Multiple Inheritance	1
5.6	Inheritance and Design	1
5.7	Run-Time Type Identification	1
5.8	Templates and inheritance	1
6	**Exceptions**	
6.1	Throwing an exceptions	0.5
6.2	Try Blocks	0.5
6.3	catching an exception	1
6.4	Exception Specification	1
7	**Input/output**	
7.1	I/O Classes	1
7.2	Files	3
Total Lectures		45

Course designers

1. S.Gilbert Nancy gilbertnancyl@tce.edu

2. R.Leena Srirlsit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T34	3	0	--	3

T34 Data Structures 3:0

(Common to all branches of Engineering B35, C25, D35, E35, G35)

Program Outcomes addressed

a. An ability to apply knowledge of engineering, information technology, mathematics, and science

c. An ability to design a system or component, or process to meet stated specifications

d. An ability to identify, formulate and solve engineering problems

Competencies

1. Ability to identify and implement appropriate data structure for a given application
2. Comprehend the terms "data abstraction", "abstract data type", and "data structures", and how data structures and algorithms have to be blended carefully to obtain efficient implementations.
3. Explain the notion of time complexity and the asymptotic notions of "Big Oh" with non-trivial examples.
4. Explain the difference between worst case complexity and best case complexity. Justify with an example algorithm for each of the complexities: O(n), O(n*2), O(n*3), O(2**n), O(n log n), O(n*2 log n),O(log n), O(log log n), O(sqrt(n)).
5. Identify all the trade-offs involved in choosing static versus dynamic data structures
6. In the context of searching, identify the trade-offs involved in selecting the most efficient data structure.
7. In the context of sorting, identify the trade-offs involved in selecting: (a) bubble-sort (b) insertion sort (c) selection sort (d) quick sort (e) merge sort (f) heap sort.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	End-semester examination
1	Remember	30	20	10
2	Understand	30	20	10
3	Apply	20	30	30
4	Analyze	10	20	20
5	Evaluate	10	10	30
6	Create	0	0	0

Course Level Learning Objectives:

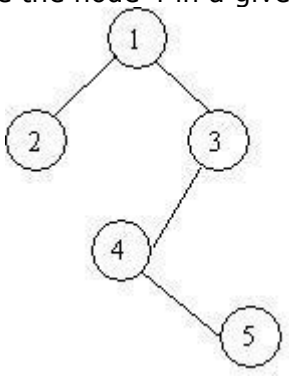
Remember

1. What is data structure?
2. List out the areas in which data structures are applied extensively?
3. What are the major data structures used in the following areas: RDBMS, Network data model and Hierarchical data model?
4. What are the notations used in Evaluation of Arithmetic Expressions using prefix and postfix forms?
5. List out few of the applications of tree data-structure?
6. List out few of the applications that make use of Multilinked Structures?
7. What is the bucket size, when the overlapping and collision occur at same time?
8. What are the Collision Resolution Techniques and the methods used in each of the type?
9. Draw a hash table with open addressing and a size of 9. Use the hash function "k%9". Insert the keys: 5, 29, 20, 0, 27 and 18 into your table (in that order).
10. Suppose that an open-address hash table has a capacity of 811 and it contains 81 elements. What is the table's load factor? (An approximation is fine)

Understand

1. If you are using C language to implement the heterogeneous linked list, what pointer type will you use?
2. What is the minimum number of queues needed to implement the priority queue?
3. How many null branches are there in a binary tree with 20 nodes?
4. How many different trees are possible with 10 nodes?
5. What is the condition for balancing to be done in an AVL tree?

6. How do you traverse a given tree using Inorder, Preorder and Postorder traversals.
7. What is the suitable efficient data structure for constructing a tree?
8. There are 8, 15, 13, 14 nodes were there in 4 different trees. Which of them could have formed a full binary tree?
9. At what location can you store the node 4 in a given binary tree using array?



10. Sort the given values using Quick Sort?

65	70	75	80	85	60	55	50	45

11. Classify the Hashing Functions based on the methods by which the key value is found.
12. What are the steps to inserting a new item at the head of a linked list? Use one short English sentence for each step.
13. Suppose that p is a reference to an IntNode in a linked list, and it is not the tail node. What are the steps to removing the node after p? Use one short English sentence for each step.
14. Write a class definition that could be used to define a node in a doubly linked list. Include only the instance variables, not the methods. Also write one sentence to describe a situation when a doubly linked list is appropriate.
15. Describe a situation where storing items in an array is clearly better than storing items on a linked list.
16. Describe why it is a bad idea to implement a linked list version a queue which uses the head of the list as the rear of the queue.

Apply

1. Convert the expression ((A + B) * C - (D - E) ^ (F + G)) to equivalent Prefix and Postfix notations.
2. Draw the B-tree of order 3 created by inserting the following data arriving in sequence - 92 24 6 7 11 8 22 4 5 16 19 20 78

3. Draw a binary Tree for the expression : A * B - (C + D) * (P / Q)
4. Is a Linked List a linear or non-linear data structure?
5. Suppose we are using the usual IntNode class (with instance variables called data and link). Your program is using an IntNode variable called head to refer to the first node of a linked list (or head is null for the empty list). Write a few lines of C++ code that will print all the double numbers on the list?
6. Suppose we are using the usual IntNode class (with instance variables called data and link), and that locate is referring to a node in a linked list. Write an assignment statement that will make locate refer to the next node in the list (if there is one). If there is no next node, then your assignment statement should set locate to null.
7. Suppose that p, q, and r are all references to nodes in a linked list with 15 nodes. The variable p refers to the first node, q refers to the 8th node, and r refers to the last node. Write a few lines of code that will make a new copy of the list. Your code should set THREE new variables called x, y, and z so that: x refers to the first node of the copy, y refers to the 8th node of the copy, and z refers to the last node of the copy. Your code may NOT contain any loops, but it can use the other IntNode methods.

Analyze

1. Why is the order of an algorithm generally more important than the speed of the processor?
2. Convert each time formula to the best possible big-O notation. Do not include any spurious constants in your big-O answer.

Time Formula	**Big-O**
10n	.
$2n^2$	.
3 times log (base 2) of n	.
$2n^2 + 10n$	.

3. Which of these is the correct big-O expression for 1+2+3+...+n?

   A. O(log n)
   B. O(n)
   C. O(n log n)
   D. $O(n^2)$

4. Which of the following formulas in big-O notation best represent the expression $n^2+35n+6$?

5.

A. $O(n^3)$
B. $O(n^2)$
C. O(n)
D. O(42)

6. Answer true or false for this statement: For all possible inputs, a linear algorithm to solve a problem must perform faster than a quadratic algorithm to solve the same problem.
   - TRUE
   - FALSE

7. Answer true or false for this statement: True or false: An algorithm with worst case time behavior of 3n takes at least 30 operations for every input of size n=10.
   - TRUE
   - FALSE

8. What term is used to describe an O(n) algorithm.

   A. Constant
   B. Linear
   C. Logarithmic
   D. Quadratic

9. Here is some code for an *integer* variable n:

```
while (n > 0)
  {
    n = n/10; // Use integer division
  }
```

What is the worst-case time analysis for the above loop?

A. O(1)
B. O(log n)
C. O(n)
D. $O(n^2)$

10. Express the formula (n - 2)*(n - 4) using big-O notation:

    A. O(1)
    B. O(8)
    C. O(log n)
    D. O(n)
    E. None of the above

11. Fill in the following table for the times to sort an array of n items. Use only big-O notation, and do not have any extraneous constants in your expressions.

	Worst Case	Average Case
Binary search of a sorted array	.	.
Insertion sort	.	.
Merge sort	.	.
Quick sort without "median of three" pivot selection	.	.
Quick sort with "median of three" pivot selection	.	.
Selection sort	.	.
Heap sort	.	.

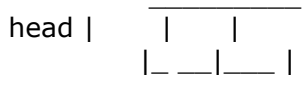
Evaluate

1. Compare the worst-case big-O time analysis for these two methods: The add method for the Bag that is implemented using an array, and the add method for the Bag that is implemented using a linked list.
2. Compare the worst-case big-O time analysis for these two methods: The remove method for the Bag that is implemented using a fixed-sized array, and the remove method for the Bag that is implemented using a linked list.
3. Compare the worst-case big-O time analysis for these two methods: The addBefore method for the Sequence that is implemented using an array, and the addBefore method for the Sequence that is implemented using a linked list.
4. Compare the worst-case big-O time analysis for these two methods: The remove method for the Sequence that is implemented using an array, and the remove method for the Sequence that is implemented using a linked list.

5. I am going to execute this code with THREE pushes and ONE pop:

```
IntStack s = new IntStack( );
s.push(1);
s.push(2);
s.push(3);
System.out.println(s.pop( ));
```

Suppose that s is represented by a linked list. Draw the state of the private member variables of s after the above code:



6. Implement the following method. You may use the IntStack class and the Stack operations of push, pop, peek, isEmpty, and size. The parameter, in, is an EasyReader from Appendix B of the text and it is already attached to some kind of input. You may use the methods:

   `in.isEOLN()` -- returns true when the end of line is reached.
   `in.peek()` -- returns the next input character without actually reading it.
   `in.ignore()` -- reads and throws away the next input character.
   `in.intInput()` -- reads and returns an integer value from the EasyReader.

   This should be used only if you know that the next input characters form a valid integer value.

   The method specification is:

   public static intevaluatePostfix(EasyReader in)
   Precondition (Which is not checked): The next input line of in is a properly formed postfix expression consisting of integers, the binary operations + and -, and spaces.
   Postcondition: The method has read the next input line (including the newline) and returned the value of the postfix expression.

7. Consider the usual algorithm to convert an infix expression to a postfix expression. Suppose that you have read 10 input characters during a conversion and that the stack now contains these symbols: +(top), (, *(bottom)Now, suppose that you read and process the 11th symbol of the input. Draw the stack for the case where the 11th symbol is:

   A. A number:

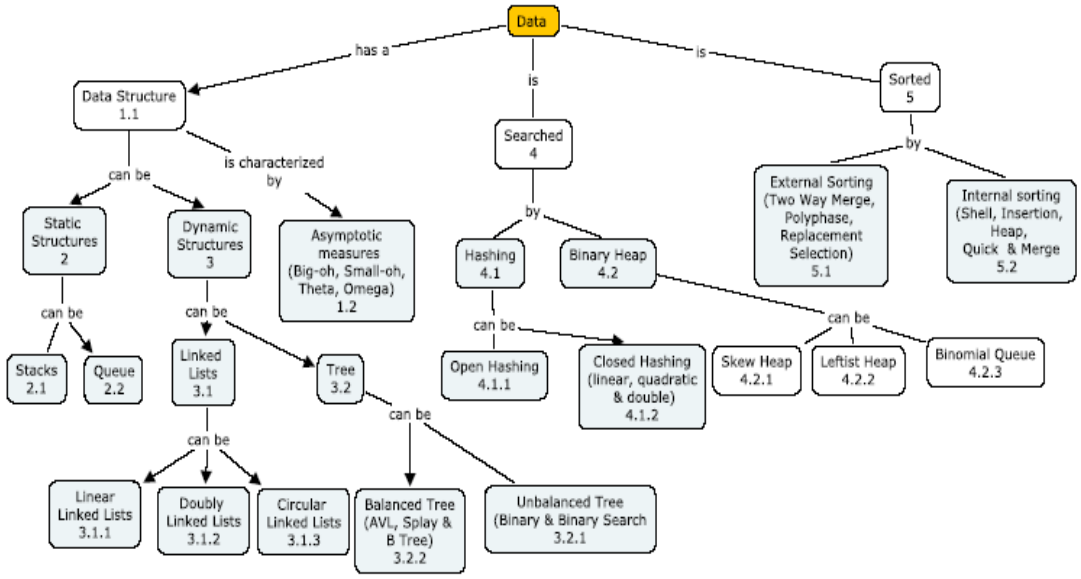
   B. A left parenthesis:

   C. A right parenthesis:

   D. A minus sign:

   E. A division sign:

Concept Map



Course Contents:

1. Data

1.1 Data Structure

1.2 Asymptotic Measures

2. Static Data Structures

2.1 Stacks

2.2 Queues

3. Dynamic Data Structures

3.1 Linked Lists

- 3.1.1 Linear Linked Lists
- 3.1.2 Doubly Linked Lists
- 3.1.3 Circular Linked Lists

3.2 Trees

- 3.2.1 Unbalanced Trees
- 3.2.2 Balanced Trees

4. Data Search

4.1 Hashing

- 4.1.1 Open Hashing
- 4.1.2 Closed Hashing

4.2Heap

4.2.1 Skew Heap

4.2.2 Leftist Heap

4.2.3 Binomial Queue

5 Data Sorting

5.1 Internal Sorting

5.2 External Sorting

Syllabus

Data: Data Structure, Asymptotic Measures **Static Data Structures:** Stacks, Queues **Dynamic Data Structures:** Linked Lists: Linear Linked Lists, Doubly Linked Lists and Circular Linked Lists, Trees: Unbalanced and Balanced Trees, **Data Search:** Hashing: Open Hashing and Closed Hashing; Heap: Skew Heap, Leftist Heap, Binomial Queue **Data Sorting:** Internal Sorting: Insertion sorting, Shell sorting, Quick sorting, Merge sorting and Heap sorting; External Sorting

Textbook

1. Richard F. Gilberg , Behrouz A. Forouzan: Data Structures: A Pseudocode Approach With C, 2nd Edition, Thomson Learning, 2003

Reference

1. Mark Allen Weiss: Data Structures and Algorithms in C, Addison-Wesley, 1997

Lecture Schedule

No.	Topic	No. of Lectures
1	**Data**	
1.1	Data Structure	0.5
1.2	Asymptotic Measures	1
2	**Static Data Structures**	
2.1	Stacks	3
2.2	Queues	2
3	**Dynamic Data Structures**	
3.1	Linked Lists	0.5
3.1.1	Linear Linked Lists	2

3.1.2	Doubly Linked Lists	1.5
3.1.3	Circular Linked Lists	1.5
3.2	Trees	1
3.2.1	Unbalanced Trees	3
3.2.2	Balanced Trees	6
4	**Data Search**	
4.1	Hashing	0.5
4.1.1	Open Hashing	1
4.1.2	Closed Hashing	2.5
4.2	Heap	2
4.2.1	Skew Heap	1.5
4.2.2	Leftist Heap	1.5
4.2.3	Binomial Queue	1
5	**Data Sorting**	
5.1	Internal Sorting	
5.1.1	Insertion sorting	1
5.1.2	Shell sorting	1
5.1.3	Quick sorting	1
5.1.4	Merge sorting	1
5.1.5	Heap sorting	2
5.2	External Sorting	2

Course Designers:

1. M.K. Kavitha Devi mkkdit@tce.edu
2. S. Sudhassj@tce.edu
3. S. Geetha Sgeetha@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T 35	3	0	--	3

T35 Databases: Practice 3:0

Preamble: This course aims at facilitating the student to understand the various functionalities of DBMS software and perform many operations related to creating, using and maintaining databases for Real-world applications and introducing emerging technologies in Databases.

Programme Outcomes addressed

a. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

b. Graduates will demonstrate an ability to design a system, component or process as per needs and specifications.

c. Graduate will demonstrate skills to use modern engineering tools, softwares and equipment to analyze problems.

Competencies: At the end of the course the student will be able to

1. Retrieve data from Data bases including selecting rows, limiting the selection, and single-row functions
2. Display data from multiple tables, and using group functions, subqueries and runtime libraries.
3. Alter the tables and constraints, create sequences and views, and control user access.
4. Develop Stored Procedures, interact with the Oracle database, control process flow, handle cursor, and handle errors.
5. Demonstrate an understanding of the architecture of a DBMS, start and stop a Database instance, and create a Database.
6. Access and update data, and manage transaction concurrency, database structure and storage allocation.Manage rollback segments, tables and indexes, clusters, and data integrity constraints.
7. Analyze Large-Scale data using Data mining Techniques.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	End-semester examination
1	Remember	20	20	20
2	Understand	30	30	30
3	Apply	50	50	50
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	0	0	0

Course Level Learning Objectives

Remember

1. What is database management system?
2. What is a schema?
3. What is a cursor?
4. What is functional dependency?
5. What is a table join?
6. What is integrity constraint? What are the five types of integrity constraints?
7. What is a composite index?
8. What is cardinality?
9. What is called query processing?
10. What is Market-Basket analysis?
11. What are the two steps in Classification?
12. What are the limitations of Kmeans clustering?

Understand

1. Explain the structure of a DBMS
2. Identify some of the character, number and date functions available in SQL. What are two functions that allow the user to transform column values regardless of the data type?
3. How can constraints be created and enforced on views?
4. How are views dropped?
5. How are rollback statements created and altered?
6. Explain the need of triggers in a database.

7. Explain string operations and aggregate functions used in SQL
8. What are the three ways of inserting data into a table?
9. Distinguish Classification from Prediction
10. How do you evaluate the accuracy of a Classifier?

Apply

1. Develop ER diagram for the following
   Customer withdraws money from his account
   Student writes Examinations
   Driver drives a car
2. The DBA defines an integrity constraint associated with a table, which fails on creation. What can be done to determine which rows in a table violate an integrity constraint?
3. (a) Design an ER diagram for an IT training group database that will meet the information needs for its training program. Clearly indicate the entities, relationships and the key constraints. The description of the environment is as follows :
   The company has 12 instructors and can handle up to 100 trainees for each training session. The company offers 5 Advanced technology courses, each of which is taught by a team of 2 or more instructors Each instructor is assigned to a maximum of two teaching teams or may be assigned to do research Each trainee undertakes one Advanced technology course per training session.

b) Examine the table shown below :

Staff No.	Branch No.	Branch Address	Name	Position	Hrs/Week
E101	B02	Sun Plaza, Delhi, 110001	Ram	Assistant	16
E101	B04	2/3 UT, Delhi, 110111	Ram	Assistant	9
E122	B02	Sun Plaza, Delhi, 110001	Mohan	Assistant	14
E122	B04	2/3 UT, Delhi, 110111	Mohan	Assistant	10

(i)Why is the table above not in 2NF ?

(ii) Describe the process of normalizing the data shown in the table above to third normal

form (3NF).

(iii) Identify the primary and foreign keys in your 3NF relations.

(d) Explain the following integrity rules of a relational model with the help of an example

(i)Entity Integrity

(ii) Referential Integrity

4) (a) Consider the following relations :

Employee (emplD, FirstName, LastName, address, DOB, sex, position, deptNo)

Department (dtptNo, deptName, mgr, empID)

Project (projNo, projName, deptNo)

Work on (empID, projNo, hours worked)

Write the SQL statements for the following :

(i) List the name and addresses of all employees who work for the IT department.

(ii) List the total hours worked by each employee, arranged in order of department number and within department, alphabetically by employee surname.

(iii) List the total number of employees in each department for those departments with more than 10 employees.

(iv) List the project number, project name and the number of employees who work on that project.

(b) What is a database model ? Explain any two types of data models with an example for each.

5. A table Employee with the following fields: EmpNo, Name, Designation, salary are maintained in a computer. Write SQL queries for the following.

a. Display the details for all employees

b. Find average salary paid to employees

c. Display the details of all employees whose salary fall in the range of 10,000 and 50,000Rs.

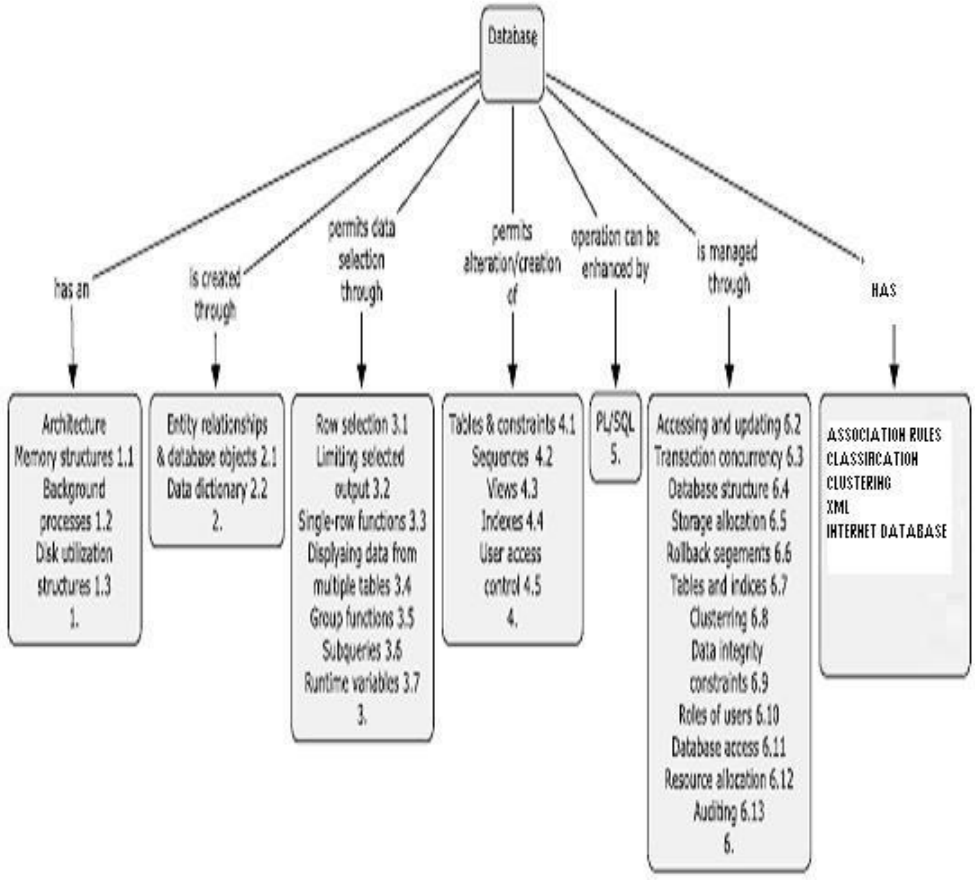
d. List the names of all employees whose name start with the letter "A"

6) Apply Frequent Itemset Mining to Market-Basket Data and illustrate the various steps in Apriori Algorithm.

7) Apply the ID3 algorithm for Decision tree Induction and generate the tree and classification rules.

8) Apply the Kmeans algorithm to Reuters Dataset and evaluate the accuracy of the resulting Clusters.

Concept Map



Syllabus

Introduction- DBMS, Characteristics of Database approach, Advantages of using the DBMS approach, Schema architecture, Data Independence, Database System Environment, Classification of DBMS **Schema Definitions, Basic Constraints and Queries-** SQL data definition and data types, Basic constraints in SQL, Schema Change Statements in SQL, Basic Queries in SQL, Complex SQL Queries, Insert, Delete and Update Statement in SQL, Views in SQL **Programming language extension to SQL-** Fundamentals, Control Structures, Data Manipulation, Cursors, Exceptions **Advanced Programming Concepts-** Records, Tables, Varrays, Procedures, Functions, Packages, Triggers, Data Dictionary Views **Emerging Technologies-** XML and Internet Databases, Data mining Concepts, Association rules, Classification, Clustering

Text Books:

1.RamezElmasri and ShamkantB.Navathe, " Fundamentals of Database Systems, 4th edition,Pearson Education,2004.

2.Nilesh Shah, " Database Systems using Oracle,2nd edition,Prentice Hall of India Pvt Ltd,2007.

Reference Books:

1.Ian H Witten and Eibe Frank, Practical Machine learning tools and techniques with java implementation", Morgan Kaufmann publications, 2000

2.AbrahamSilberschatz,HenryF.Korth and Sudarshan, "Database System Concepts" , 5th edition, Mcgraw-Hill,2006.

Course Contents

No.	Topic	No. of Lectures
1	**Introduction**	
1.1	DBMS	1
1.2	Characteristics of Database approach	1
1.3	Advantages of using the DBMS approach	1
1.4	Schema architecture	1
1.5	Data Independence	1
1.6	Database System Environment	2
1.7.	Classification of DBMS	1
2.	**Schema Definitions, Basic Constraints and Queries**	
2.1	SQL data definition and data types	1
2.2	Basic constraints in SQL	1
2.3	Schema Change Statements in SQL	1
2.4	Basic Queries in SQL	2
2.5	Complex SQL Queries	1

2.6	Insert, Delete and Update Statement in SQL	1
2.7	Views in SQL	1
3	**Programming language extension to SQL**	
3.1	Fundamentals	2
3.2	Control Structures	1
3.3	Data Manipulation	1
3.4	Cursors	2
3.5	Exceptions	2
4	**Advanced Programming Concepts**	
4.1	Records	1
4.2	Tables	1
4.3	Varrays	1
4.4	Procedures	1
4.5	Functions	1
4.6	Packages	1
4.7	Triggers	1
4.8	Data Dictionary Views	1
5	**Emerging Technologies**	
5.1	XML and Internet Databases	2
5.2	Data mining Concepts	1
5.3	Association rules	1
5.4	Classification	2
5.5	Clustering	2

COURSE DESIGNERS:

1. S.SenthamaraiKannansskit@tce.edu
2. M. Indra Devimidit@tce.edu
3. S. Appavu alias Balamuruganbit@tce.edu

Subject Code	Lecture	Tutorial	Practical	Credit
T36	3	--	--	3

T36 Computer Organization 3:0

Preamble

This course needs a prerequisite of Digital logic circuits. It covers the basic principles of Computer organization, operation and performance. The main objective of this subject is to cover the overall basic computer hardware structure including, the peripheral devices and to provide the discussion of the fundamentals of computer organization which relates these to contemporary design issues.

Program outcomes addressed

Graduates will demonstrate

a. An ability to apply the knowledge of engineering, information technology, mathematics and science.

c. An ability to design a system or component, or process to meet stated specifications.

e. an ability to identify, formulate and solve engineering problems.

Competencies

Students will be able to

1. Describe the structure of computers and design the Instruction set by identifying the instruction types and addressing modes.
2. Design the arithmetic logic units and control unit of a simple computer.
3. Describe the various memories like ROM, RAM, Cache memory and Virtual memory.
4. Demonstrate an understanding of the I/O data transfer.
5. Determine the performance improvement in the processor by the use of pipelining.

Assessment Pattern

S NO	Bloom's Category	Test 1	Test 2	Test3	End-semester examination
1	Remember	20	20	20	20
2	Understand	40	40	40	30
3	Apply	40	40	40	50
4	Analyze	0	0	0	0
5	Evaluate	0	0	0	0
6	Create	0	0	0	0

Course Level Learning Objectives

Remember

1. Identify the importance of MAR, MDR.
2. List the three phases in the execution of an instruction.
3. Show the role of micro instruction register.
4. Differentiate SRAMs and DRAMs.
5. Define locality of reference.
6. Define the hit ratio.
7. State the role of Memory Management Unit (MMU).
8. Mention the role of handshaking in asynchronous transfer.
9. Differentiate the static branch prediction and dynamic branch prediction.
10. Show the importance of reorder buffer.

Understand

1. Describe all the addressing modes
2. Explain the execution of a complete instruction
3. Discuss the working principle of a Hardwired control unit
4. Explain Booth's algorithm with suitable example.
5. Describe how read and write operations are done in a static memory
6. Translate the virtual address into a physical address
7. Illustrate how the devices interrupt the processor and how they are serviced
8. Explain how data transfer takes place using DMA
9. Illustrate how the pipelining improves the performance of computers.

Apply

1. Register R5 points to the top of the stack Write the sequence of instructions using the Auto increment, Auto decrement and Index addressing mode to perform the following task: Pop the top two items off the stack, add them and then push the result onto the stack.
2. Specify the addressing mode used in the following instruction: Move #200, R0
3. Write a program that can evaluate the expression A * B + C *D in a single accumulator processor.
4. Classify each of the following as Data movement, data manipulation or program control instruction

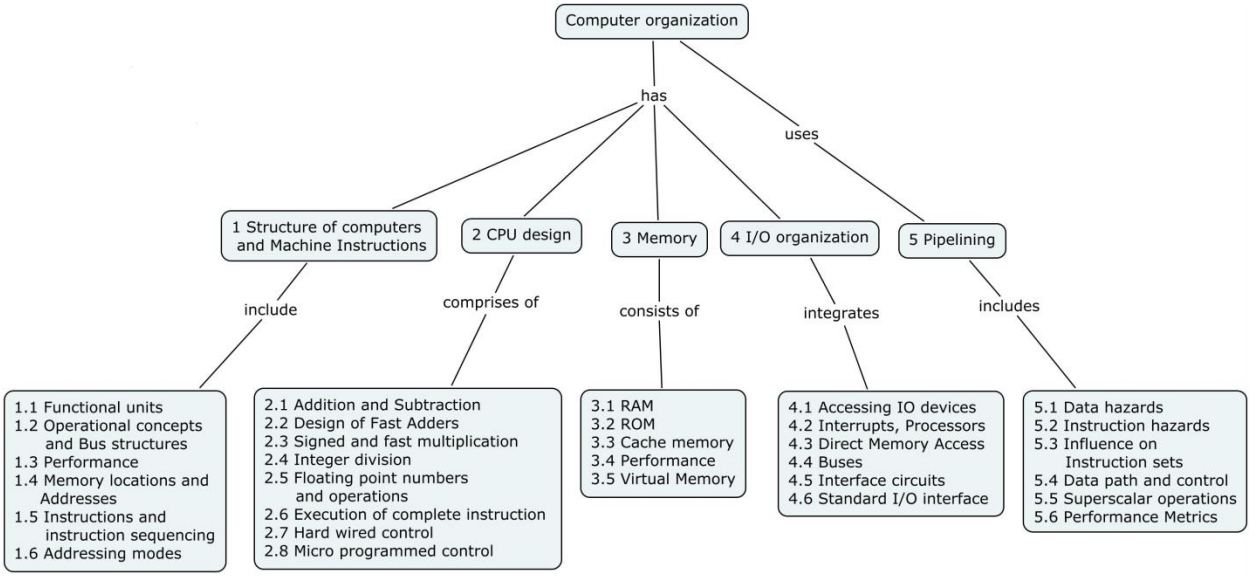
   (a)SUB (b) XOR (c) MOV R1, R2 (d) JMPZ 10H (e) LDAC 10 (f) JNC 200h(g) MVI A, 89h (h) LDA 8Ah (i) STA 2000h (j) JZ 23h

5. Consider a computer that has a byte addressable memory organized in 32 bit words according to the big-endian scheme. A program reads ASCII characters entered at a keyboard and stores them in successive byte locations, starting at location 1000. Show the contents of the two memory words at locations 1000 and 1004 after the name "Computer "has been entered
6. Consider the binary numbers in the following addition and subtraction problems to be signed, 6 bit values in the 2's complement representation. Perform the operations indicated, Specify whether or not arithmetic overflow occurs: 010110 111111
7. Divide 1000 by 11 using restoring division algorithm
8. A block-set-associative cache consists of a total of 64 blocks divided into 4-block sets. The main memory contains 4096 blocks, each consisting of 128 words.
    a. How many bits are there in the main memory address?
    b. How many bits are there in each of the TAG, SET and WORD fields?
9. The following code segment is to be executed on a RISC processor with the three stage instruction pipeline.

    1:R3←R1 + R2
    2:R5 ←R3 + R4
    3:R6 ←R1 + R5
    4: JMP 9
    5:R6←R6 + R7
    6:R7←R7 + R8
    9:R2←R1 + R3

    Show the final code and execution trace if branch conflicts are resolved by
    i. No-op insertion
    ii. Instruction reordering
    iii. Stall insertion
10. A byte addressable computer has a small data cache capable of holding 8 32-bitwords.Eaach cache block consists of one 32-bit word. When a given program is executed the processor reads data from the following sequence of hex addresses:200,204,208,2FO,200,204,218,21C,24C
    a. Show the contents of the cache at the end of each pass through this loop if a directly mapped cache is used. Compute the hit ratio.
    b. Repeat the problem for an associative-mapped cache that uses the LRU replacement algorithm.
    c. Repeat the problem for a four-way-set-associative cache.

Concept Map



Syllabus

Structure of computers & Machine Instructions - Functional units - Operational concepts - and Bus structures – Performance - Memory locations and Addresses - Instructions and Instruction Sequencing - Addressing modes - **CPU design -** Addition and Subtraction - Design of Fast Adders - Signed and fast multiplication - Integer division - Floating point numbers and operations - Execution of complete instruction - Hard wired control - Micro programmedcontrol - **Memory** – RAM – ROM - Cache memory – Performance - Virtual Memory - **I/O organization -** Accessing IO devices – Interrupts – Processors - Direct Memory Access – Buses - Interface circuits - Standard I/O interface - **Pipelining** – Data hazards - Instruction hazards - Influence on Instruction sets - Data path and control - Superscalar operations - Performance Metrics.

Text Book:

Carl Hamacher, ZvonkoVranesic And SafwatZaky, Computer Organization , Tata McGraw Hill, Fifth Edition.

Reference Books:

1. William Stallings, "Computer Organization and Architecture – Designing for Performance", Sixth Edition, Pearson Education, 2003.

2. David A. Patterson And John L. Hennessy, "Computer Organization and Design: The Hardware/Software Interface", Third Edition, Elsevier, 2005.
3. John P. Hayes, "Computer Architecture and Organization", Third Edition, Tata McGraw Hill, 1998.

Course content and Lecture Schedule

No.	Topic	No. of Lectures
1.	**Structure of computers and Machine Instructions**	
1.1	Functional units	1
1.2	Operational concepts and Bus structures	1
1.3	Performance	1
1.4	Memory locations and Addresses	1
1.5	Instructions and Instruction Sequencing	2
1.6	Addressing modes	2
2	**CPU design**	
2.1	Addition and Subtraction	1
2.2	Design of Fast Adders	1
2.3	Signed and fast multiplication	2
2.4	Integer division	1
2.5	Floating point numbers and operations	1
2.6	Processing unit	2
2.7	Execution of complete instruction	1
2.8	Hard wired control	1
2.9	Micro programmed control	2
3	**Memory**	
3.1	RAM	1
3.2	ROM	1
3.3	Cache memory	1
3.4	Performance	1
3.5	Virtual Memory	2
4	**I/O organization**	
4.1	Accessing I/O Devices	1
4.2	Interrupts	2
4.3	Processors	1
4.4	Direct Memory Access	1

4.5	Buses	1
4.6	Interface circuits	1
4.7	Standard I/O interface	1
5	**Pipelining**	
5.1	Data hazards	1
5.2	Instruction hazards	1
5.3	Influence on Instruction sets	1
5.4	Data path and control	1
5.5	Superscalar operations	1
5.6	Performance Metrics	1
Total Lectures		**40**

Course Designers:

1. R.MuthuSelvirmsit@tce.edu

2. M.Nirmala Devi mnit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T 37	--	--	1	1

T37 Object Oriented Programming Lab. 0:1

Object-oriented programming requires the programmer to view programs differently from the older procedural programming paradigm. It also requires a language that supports classes and objects. Java can be used for object-oriented programming. Object-oriented programming has been promoted as a more productive and natural way to view solutions to problems. Object-oriented programmers are more productive because the classes they create for one project may be reusable on another project. It is a natural way of viewing software development, because many problems are stated in terms of objects which then interact. The interaction is what causes the computation to progress.

The object-oriented paradigm requires that we examine the following concepts: classes, objects, and the creation process, packages, public, methods, data members or fields, static methods and static data members, extending a class to get a new class, a class that implements an interface, container classes, a class hierarchy, exception handling and more.

Upon completion of this course, the students will be able to:

- Understand object oriented programming principles, explain how OOP differs from structural programming, and discuss the advantages of OOP
- Utilize OOP in Java by designing and writing Java classes, encapsulating logic, reusing existing code through inheritance/polymorphism and composition, and modeling real-world relationships between objects
- Read as well as write Java syntax, including declarations, assignments, operators, flow-control structures, generics, annotations, enumerations, naming conventions, etc.
- Define and handle error conditions in Java through the use of exceptions

List of Experiments

1. Write a Java program to demonstrate the usage of classes and methods.
2. Write a Java program to demonstrate the implementation of Constructors.
3. Write a Java program to demonstrate the concept of Polymorphism in
    a. Method overloading.
    b. Constructor overloading.
4. Write a Java program to demonstrate the concept of
    a. Multilevel inheritance.
    b. Hierarchical inheritance.
5. Write a Java program to demonstrate the implementation of interfaces.

6. Write a Java program that builds a Package, imports and uses it.
7. Write a Java program to demonstrate the creation and usage of Exception subclass.
8. Write a Java program to demonstrate the implementation of multithreading.
9. Write a Java program that reads and writes the contents of a File using I/O classes.
10. Write a Java program to demonstrate the implementation of the Remote Method Invocation.
11. Write a Java program that designs an Applet window with Event Handling.
12. Mini Project.

Course designers

1. S. Geethasgeetha@tce.edu
2. P. Karthikeyankarthikit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T 38	--	--	1	1

T38 Data Bases Lab 0:1

Preamble:

This Lab aims at giving adequate exposure to the SQL and PL/SQL programming within the Oracle RDBMS environment.

LIST OF EXPERIMENTS:

1. Creation and Modification of relations
2. Integrity constraint enforcement
3. Creation and updation of views
4. Simple SQL Queries
5. Exercises using PL/SQL
6. Cursor management
7. Procedures, functions and packages
8. Creation of triggers
9. Declaration of PL/SQL tables
10. Association Rule Mining using WEKA
11. Classification & Clustering using WEKA
12. Mini Projects

REFERENCE

1. George Koch and Kevin Loney Oracle 8: The complete reference,McGraw Hill,1997

COURSE DESIGNERS:

1. S.SenthamaraiKannansskit@tce.edu
2. M. Indra Devimidit@tce.edu
3. S. Appavu alias Balamuruganbit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T39	0	0	1	1

T39 Data Structures Lab. 0:1

Preamble: With a dynamic learn-by-doing focus, this laboratory course encourages students to explore data structures by implementing them, a process through which students discover how data structures work and how they can be applied. This course challenges students to exercise their creativity in both programming and analysis.

Prerequisite: Proficiency in C

List of Exercises:

1. Implementation of all the basic operations like creation, insertion, deletion, find, display etc in a Linear linked list.
2. Polynomial addition and subtraction using Linked List
3. Josephus Problem using Circular List
4. Palindrome checking using Doubly linked List
5. Stack implementation using Array, Two way stack implementation
6. Expression evaluation using stack
7. Queue implementation
8. Round Robin Scheduling using circular Queue
9. Implementation of all the basic operations like creation, insertion, deletion, find, display etc in a Binary Search Tree.
10. Implementation of all the basic operations like creation, insertion, deletion, find, display etc in an AVL Tree.
11. Implementation of all the basic operations like creation, insertion, deletion, find, display etc in a Splay Tree.
12. Implementation of open hashing and closed hashing for searching
13. Design a suitable application for analyzing the performance of Insertion Sort, Merge Sort, Shell Sort, Quick Sort and Heap Sort Algorithms.

REVISED CURRICULUM AND DETAILED SYLLABI

FOR

B.Tech DEGREE (Information Technology) PROGRAM

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2012-2013

THIAGARAJARCOLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to AnnaUniversity)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Approved in BOS Meeting on 13.10.2012 Approved in 45th Academic Council Meeting on 24.11.2012

Scheduling of Courses for students admitted in the Academic year 2012-2013

Semester							Laboratory/Project Work		
8(21)	Elective 6 3:0	Elective 7 3:0	Elective 8 3:0				T84 Project 0:12		
7(22)	T71 Management Theory and Practice 3:0	T72 Wireless and Mobile Communication 3:1	Elective 3 3:0	Elective 4 3:0	Elective 5 3:0		T78 Project 0:6		
6(22)	T61 Operations Research 3:1	T62 Accounting and Finance 3:0	T63 Data Warehousing and Mining 3:1	T64 Web Programming 3:0	Elective 1 3:0	Elective 2 3:0	T67 Web Programming Lab 0:1	T68 Software tools Lab 0:1	
5 (25)	T51 Discrete Mathematics 4:0	T52 Software Engineering Methodologies 3:1	T53 Information Storage and Management 3:0	T54 Information Security and Assurance 4:0	T55 Graphics and Visualization 3:1	T56 Platform Technologies 4:0	T57 Platform Technologies Lab 0:1	T58 Network Security Lab 0:1	
4(25)	T41 Engineering Mathematics - IV 4:0	T42 Algorithms: Design Principles 3:0	T43 Operating Systems: Principles 3:1	T44 Computer Networks 3:0	T45 Data bases: Principles and Design 4:0	T46 System and Data Centre Administration 3:0	T47 Network Programming Lab 0:1	T48 System Admin Lab 0:1	T49 Professional Communication 1:1
3(22)	T31 Engineering Mathematics –III 4:0	T32 Object Oriented Programming 3:0	T33 Human Computer Interaction 3:0	T34 Data Structures 3:0	T35 Data Bases: Practice 3:0	T36 Information Systems Design 3:0	T37 Object Oriented Programming Lab 0:1	T38 Data Bases Lab 0:1	T39 Data Structures Lab 0:1
2 (22)	T21 Engineering Mathematics – II 4:0	T22 Information Technology 3:0	T23 Computer Organization and Design 3:1	T24 Computers and Programming 3:0	T25 Operating Systems : Configuration and use 3:0	T26 Environment and Ecology 2:0	T27 Operating System Lab 0:1	T28 Computer Programming Lab 0:1	T29 Workshop 0:1
1 (25)	H11 Engineering Mathematics – I 4:0	H12 Physics 3:0	H13 Chemistry 3:0	H14 English 3:0	H15 Basics of EEE 4:0	H16 Basics of ME and CE 4:0	H17 Physics Laboratory 0:1	H18 Chemistry Laboratory 0:1	H19 Engineering Graphics 0:2

Approved in BOS Meeting on 13.10.2012 Approved in 45th Academic Council Meeting on 24.11.2012

REVISED CURRICULUM AND DETAILED SYLLABI

FOR

B.Tech DEGREE (Information Technology) PROGRAM

FOURTH SEMESTER

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2012-2013

THIAGARAJARCOLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to AnnaUniversity)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Approved in BOS Meeting on 13.10.2012 Approved in 45th Academic Council Meeting on 24.11.2012

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015

B.Tech. Degree (Information Technology) Program

SUBJECTS OF STUDY

(For the candidates admitted from 2012-2013)

FOURTH SEMESTER

Subject code	Name of the subject	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
T41	Engineering Mathematics – IV	BS	4	-	-	4
T42	Algorithms: Design Principles	DC	3	-	-	3
T43	Operating Systems: Principles	DC	3	1	-	4
T44	Computer Networks	DC	3	-	-	3
T45	Data Bases: Principles and Design	DC	4	-	-	4
T46	System and DataCenter Administration	DC	3	-	-	3
PRACTICAL						
T47	Network Programming Lab	DC	-	-	3	1
T48	System Admin Lab	DC	-	-	3	1
T49	Professional Communication	HSS	-	-	3	2
Total			**20**	**1**	**9**	**25**

BS : Basic Science
HSS : Humanities and Social Science
ES : Engineering Science
DC : Department Core

L : Lecture
T : Tutorial
P : Practical

Note:

1 Hour Lecture/Tutorial is equivalent to 1 credit
2/3 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015

B.Tech. Degree (Information Technology) Program

SCHEME OF EXAMINATIONS

(For the candidates admitted from 2012-2013)

FOURTH SEMESTER

S.No.	Sub. code	Name of the subject	Duration of Terminal Exam. in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	T41	Engineering Mathematics-IV	3	50	50	100	25	50
2	T42	Algorithms: Design Principles	3	50	50	100	25	50
3	T43	Operating Systems: Principles	3	50	50	100	25	50
4	T44	Computer Networks	3	50	50	100	25	50
5	T45	Data Bases: Principles and Design	3	50	50	100	25	50
6	T46	System and DataCenter Administration	3	50	50	100	25	50
PRACTICAL								
7	T47	Network Programming Lab	3	50	50	100	25	50
8	T48	System Admin Lab	3	50	50	100	25	50
9	T49	Professional Communication	3	50	50	100	25	50

* Continuous Assessment evaluation pattern will differ from subject to subject and for different tests. This will have to be declared in advance to students. The department will put a process in place to ensure that the actual test paper follow the declared pattern.

** Terminal Examination will be conducted for maximum marks of 100 and subsequently be reduced to 50 marks for the award of terminal examination marks.

Sub code	Lectures	Tutorial	Practical	Credit
T41	4	0	-	4

T41 Engineering Mathematics IV **4:0**

(Common to CSE and IT,C41,T41)

Preamble: An engineering student needs to have some basic mathematical tools and techniques. This emphasizes the development of rigorous logical thinking and analytical skills of the student and appraises him the complete procedure for solving different kinds of problems that occur in engineering. This course aims at giving adequate exposure in the theory and applications of Statistics, Probability, Sampling and Graph Theory.

Program Outcomes addressed

a. Graduates will demonstrate knowledge of Mathematics, Science and Engineering.
b.Graduates will demonstrate an ability to identify, formulate and solve engineering problems.
c.Graduates will develop confidence for self education and ability for life-long learning.

Competencies

1. Determine correlations among variables (linear, non-linear) for regression lines.
2. Express the probability distributions arising in the study of Engineering problems and their applications.
3. Construct the various tests essentially needed for testing of samples for testing for different attributes.
4. Understand the basic concepts of graph theory used in computers application.
5. Construct the minimal spanning tree by using various algorithms.

Assessment Pattern

	Bloom's category	Test 1	Test 2	Test 3 / End Semester Examinations
1	Remember	10	10	0
2	Understand	30	30	30
3	Apply	60	60	70
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	0	0	0

Course level learning objectives

Remember

1. Define multiple correlation and Non linear regression.

2. Define Discrete and Continuous Random Variables.
3. State the properties of probability distribution function.
4. Give the PDF of a Normal Distribution.
5. What do you mean by test of Hypotheses?
6. Distinguish between critical region and acceptance region.
7. Evaluate the number of edges in the complete graph with n vertices.
8. Define connected and disconnected graph.
9. Distinguish between Euler graphs and Hamiltonian graphs.
10. Define adjacency matrix and incidence matrix.

Understand

1. Discuss the properties of correlation and regression coefficients.
2. Differentiate between linear and non-linear regession.
3. Discuss the properties of a Normal probability distribution.
4. Predict the mean and variance of chi-square distribution.
5. Estimate the probability mass function of Hyper geometric distribution.
6. Discuss the uses of t – distribution.
7. Discuss the procedure for testing of hypothesis.
8. Interpret the level of significance and degrees of freedom for a chi square variant.
9. 9. Discuss the properties of tree.
10. Evaluate the centre for P_n.

Apply

1. Calculate the correlation coefficient for the following data and also find the line of regression by the method of least squares.

X:	5	7	9	24	15	16	29
Y:	20	12	4	15	9	10	8

2. A continuous random variable that can assume values between x = 2 and x = 5 has a density function given by f(x) = 2(1+x)/27. Find P(3<x<4).
3. State and prove the memory less property of exponential distribution.

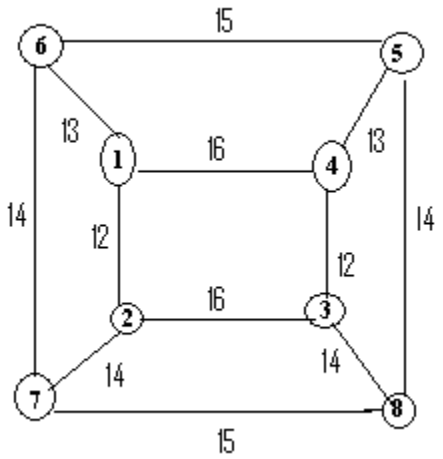
4. A coin is tossed 900 times and heads appeared 490 times. Would you conclude that the coin is a biased one?

5. Two random samples gave the following

$n_1 = 10; \sum (x_i - \bar{x})^2 = 90$; $n_2 = 12$; $\sum (y_i - \bar{y})^2 = 108$. Test whether the samples came from the same population.

6.Prove that a connected graph G is an Euler iff all the vertices of G are of even degree

7. Obtain a minimum spanning tree in the graph using prim's algorithm



8. State and prove any two characterizations for a tree with n vertices.

Syllabus

Statistics: Linear correlation and regression, Curve fitting – Method of least squares, Multiple Regression, Non linear regression.

Probability distributions: Discrete and Continuous distributions- Binomial, poisson, Geometric and Hyper Geometric distributions, Uniform, Exponential, Gamma, Weibull and Normal distributions, Using Excel for sampling and Distribution.

Test of Hypothesis: Testing hypothesis involving means and proportions, small samples-t-test,F-test and large samples -z-test, Test of goodness of fit, Chi square test.

Graph Theory: Basic definitions in graphs, walk, path and circuits, connected graphs, disconnected graphs and components, Euler graphs, operations on graphs, Hamiltonian graphs.

Trees: Properties of trees, distance and centres in a tree, rooted and binary trees, spanning trees. Algorithm-spanning tree algorithm, Kruskal's algorithm and Prism's algorithm, adjacency matrix and incidence matrix.

Text Books

1. S.C Gupta and V.K.Kapoor, "Fundamentals of mathematical statistics", Sultan Chand & Co, 2002
2. Veerarajan .T "Probability and Random Processes" TMH , 2006
3. NarsinghDeo, "Graph Theory", Prentice-Hall of India, 2004

References

1. Lecture Notes by the faculty of department of Mathematics, TCE, Madurai.
2. P.Kandasamy, K.Thilagavathy, K.Gunavathy: 'Probability Random Variable and Random Processes' S. Chand & Co , 2004.
3. V.K.Balakrishnan: "Theory and Problems of Graph Theory" Schaum's outlines, Tata McGraw-Hill Publishing Company Limited, New Delhi,2004.

Course content and lecture schedule

No	Topic	No.of Lectures
1	**Statistics**	
1.1	Linear correlation and regression	3
1.2	Curve fitting – Method of least squares	3
1.3	Multiple Regression	2
1.4	Non linear regression	2
2	**Distributions**	
2.1	Discrete and Continuous distributions	1
2.2	Binomial, Poisson	2
2.3	Geometric and Hyper Geometric distributions, Uniform	3
2.4	Exponential, Gamma,Weibull and Normaldistributions,.	3

2.5	Using Excel for sampling and Distribution	1
3	**Test of Hypothesis**	
3.1	Testing Hypothesis involving means and proportions	1
3.2	small samples t-test	2
3.3	F test	2
3.4	large samples- z-test	3
3.5	Test of goodness of fit, Chi square test	2
4	**Graph Theory**	
4.1	Basic definitions in graphs,walk,path,circuits	3
4.2	Connected and disconnected, components	2
4.3	Euler graphs, operations on graph	3
4.4	Hamiltonian graphs	2
5	**Trees**	
5.1	Properties of trees, Distance and centers in a tree	3
5.2	Rooted and binary trees, spanning trees	3
5.3	Spanning tree algorithm, Kruskal's algorithm, Prism's algorithm	3
5.4	Adjacency matrix, incidence matrix.	1

Course Designers

1. V.Mohan vmohan@tce.edu
2. G. Jothilakshmi gjlmat@tce.edu
3. A.P. Pushpalatha appmat@tce.edu
4. K. Angaleeswari kangaleeswari@tce.edu

Subject Code	Lecture	Tutorial	Practical	Credit
T42	3	0	--	3

T42 ALGORITHMS: DESIGN PRINCIPLES 3:0

Preamble

This subject will enable students to choose suitable data structure to store the information part of the program and use of efficient algorithms for developing a programming solution of a given problem. It will enhance the problem solving skills of the learner.

Programming Outcomes Addressed

Graduates will demonstrate

a. An ability to apply knowledge of engineering, information technology, mathematics and science

d. An ability to identify, formulate and solve engineering problems.

Competencies

The students will be able to

1. Analyze algorithms using amortized analysis, when appropriate. Recite analyses in terms of time and space complexity of simple algorithms that employ this method of analysis. Describe different strategies for amortized analysis, including the accounting method and the potential method.
2. Identify NP-hard and NP-complete problems.
3. Solve recurrence relation using homogeneous method, inhomogeneous method, change of variable method or master theorem.
4. Explain the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it such as Binary search, Merge sort, quick sort, Selection, Strassen's matrix multiplications.
5. Describe the Greedy algorithm paradigm and explain when an algorithmic design situation calls for it such as Knapsack problem, Job sequence with deadlines, Minimum cost spanning trees, and Single source shortest paths.
6. Demonstrate the Dynamic programming paradigm and explain when an algorithmic design situation calls for it such as Multistage graphs, All pair shortest paths, chained matrix multiplication, 0/1 Knapsack, The travelling sales person problem.
7. Understand the Graph algorithms paradigm and explain when an algorithmic design situation calls for it such as 8 – Queens's problem, Graph coloring, Hamiltonian cycles.

8. Explain the Backtracking paradigm and explain when an algorithmic design situation calls for it such as 8 – Queens problem, Graph coloring, Hamiltonian cycles

Assessment Pattern

S.No.	Bloom's Category	Test 1	Test 2	Test 3	End semester Examination
1	Remember	20	20	20	10
2	Understand	20	20	20	10
3	Apply	40	40	40	60
4	Analyze	10	10	10	10
5	Evaluate	10	10	10	10
6	Create	0	0	0	0

Course Level Learning Objectives

Remember

1. State the need of studying algorithms?
2. What is merge sort?
3. Why do leaves in the state space tree represent?
4. Explain the Hamiltonian circuit.
5. How can the output of a backtracking algorithm be thought of?
6. List the additional features required in branch-and-bound when compared to backtracking?
7. What is the general principle of backtracking method?

Understand

1. Explain the improvement that can be applied to sequential search if the list is sorted?
2. Prove that f(n) 2 o(g(n)) for f(n) = pnlgpn and g(n) = n.
3. Explain activity selection problem
4. Describe a divide and conquer algorithm to alphabetically sort a list of names, and find the exact number of comparisons needed to perform the most efficient algorithm known on a list with 2m entries.
5. Find the method used to find the solution in n-queen problem by symmetry?
6. Describe a divide and conquer algorithm to efficiently multiply two large numbers, and estimate the number of single-digit multiplications needed to multiply two numbers with n digits. Compare your answer to the number of

single-digit multiplications needed when using the "normal way" to multiply numbers

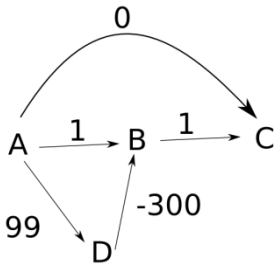
7. Compare DP with greedy? What are their similarities, and what are the differences? (See supplementary notes in the web.)
8. Explain how analysis of linear search is done with a suitable illustration.
9. Distinguish between Quick sort and Merge sort, and arrange the following numbers in increasing order using merge sort. (18, 29, 68, 32, 43,37, 87, 24, 47, 50)

Apply

1. Apply tight asymptotic bounds for each of the following recurrences using the Master Theorem if possible. Justify your answers.
   a) T(n) = 3T(bn^2 c) + n2 (Hint: lg 3 > 1.6)
2. Solve the recurrence

$$t_n = \begin{cases} n & if\ n = 0,1, or\ 2 \\ 5t_{n-1} - 8t_{n-2} + 4t_{n-3}\ otherwise \end{cases}$$

3. Apply both recursive and iterative algorithm, to compute the binomial co-efficient $\binom{n}{m}$, where $\binom{n}{0} = \binom{n}{n} = 1$. Hint:- $\binom{n}{m} = \frac{n!}{m!(n-m)!}$
4. Find an optimal solution to the knapsack instance n = 7, m= 15 (p1, p2, p3, ….p7) = (10, 5, 15, 7, 6, 18, 3) and (w1, w2, w3, ... w7) (2, 3, 5, 7, 1, 4, 1)
5. Convert the given homogenous equation into inhomogenous equation and then solve the resultant. $t_n - 2t_{n-1} = 3^n$
6. Find the All pair shortest path in the given graph using dynamic programming.

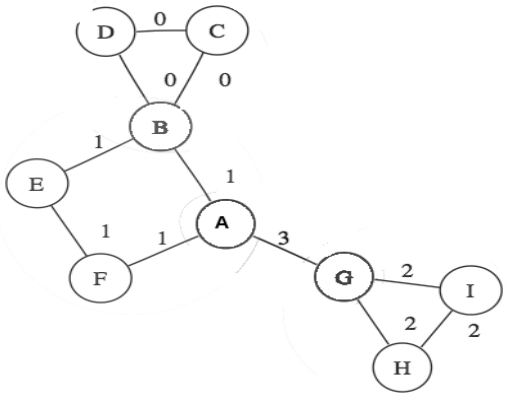


7. Using Pivotbias procedure, find the 4th smallest element in the array (4,5,2,3,1,7,1,2,3,8,4,6).

Analyze

1. Consider the following instance of "Job Sequencing problem with deadlines". Let n=4, (p1,p2,p3,p4)=(200,20,30,55) and (d1,d2,d3,d4)=(2,1,2,1). Describe all the feasible solutions for this problem. Identify if greedy approach succeeds in finding optimal solution.
2. Consider the following greedy choice for activity selection problem: Always take the activity that has earliest finish time. Prove that this greedy choice leads to an optimal solution.
3. Illustrate how the algorithm for finding the articulation points of an undirected graph works on the given graph, starting the search
   a. at node A
   b. at node D

   Analyze the result obtained by part (a) and (b).



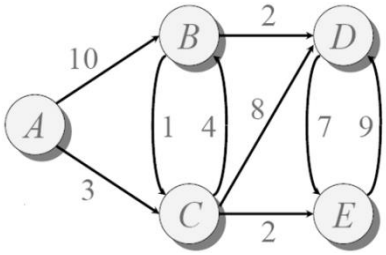
4. Use Strassen formula for multiplication for the given two matrices

$$A=\begin{matrix}3 & 2 & 4\\ 1 & 3 & 6\\ 9 & 8 & 2\end{matrix} \text{ and } B=\begin{matrix}1 & 4 & 5\\ 6 & 2 & 7\\ 3 & 4 & 3\end{matrix}$$

   Analyze its performance with the conventional method using Master theorem.
5. Prove that any weighted connected graph with distinct weights has exactly one minimum spanning tree.
6. Consider the following Matrices M1: 13*2 M2: 2*23 M3: 23*5 M4: 5*35. Find the minimum number of scalar multiplications needed to multiply them.
7. Given a set S of unit time tasks, |S| = n, with a deadline and a penalty for missing the deadline, {(di, pi)}, where di is the deadline for the ith task, 1 £ di £ n, and pi is the penalty incurred for missing this deadline. Write a greedy algorithm for assigning each of the n tasks to n time slots so that the total penalty incurred is a minimum. Analyze the time complexity for worst,average and best cases.

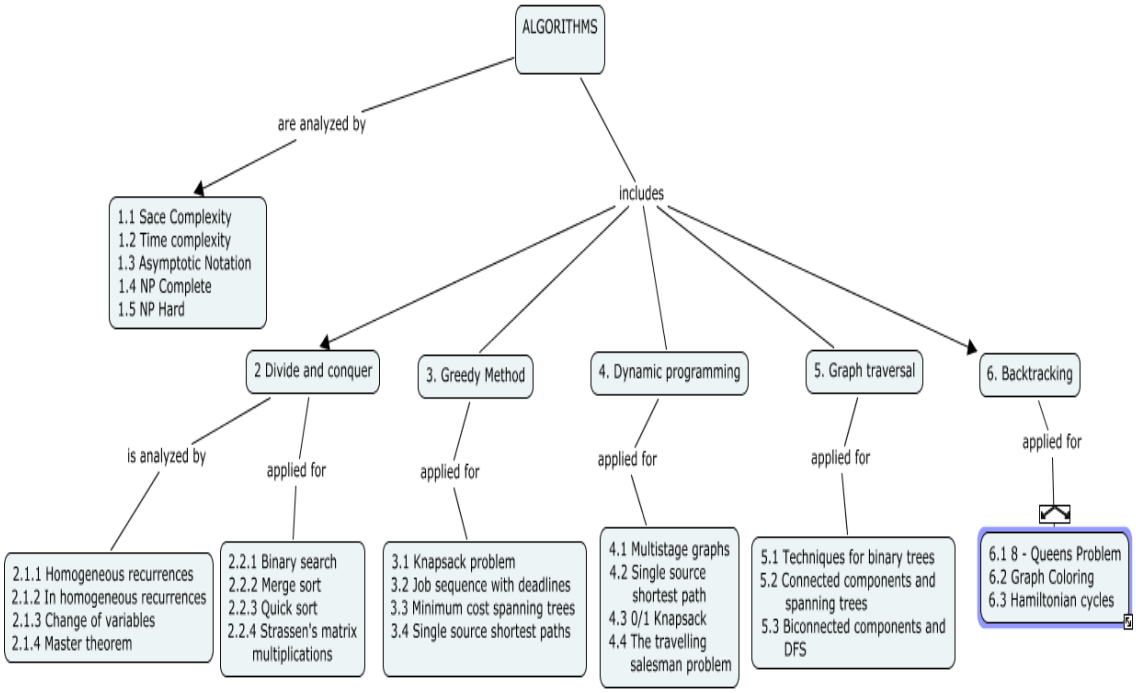
Evaluate

1. Compare the various algorithms in sorting techniques to sort a set of numbers either in ascending order or in random order.
2. Find the shortest path in the given graph using greedy and dynamic programming approaches. Evaluate the results and the time complexity taken.



3. Consider four matrices: A is 13 x 5, B is 5 x 89, C is 89 x 3, and D is 3 x 34. Count the number of scalar multiplication involved.
4. Evaluate two different Divide and Conquer approaches for multiplying the integers 4567 and 1234 and give the algorithm for the same.
5. Assume two matrix a and b of size n* n. Give an algorithm to compute matrix multiplication c= a * b. Calculate the time complexity by using Step count method and express in Asymptotic notation.
6. Given the dimension vector p = <5, 10, 3, 12, 5, 50, 6> for a chain of 6 matrices, construct a table and simulate the execution of the dynamic programming matrix-chain multiplication algorithm by calculating and filling in by hand the respective table entries.

Concept Map



Syllabus

Introduction : Space complexity, Time complexity, Asymptotic notation, Basics of NP-Complete, NP-Hard **Solving recurrences using thecharacteristics equation :** Homogeneous recurrences, In homogeneous recurrences, Change of variables, Master theorem**, The Greedy method :** General method, Knapsack problem, Job sequence with deadlines, Minimum cost spanning trees, Single source shortest paths , **Divide and conquer :** general method, Binary search, Merge sort, quick sort, Selection, Strassen's matrix multiplications**, Dynamic Programming :** general method, Multistage graphs, All pair shortest paths, chained matrix multiplication, 0/1 Knapsack, The travelling sales person problem, **Basic traversal and search techniques :** Techniques for binary trees, Connected components and spanning trees, Biconnected components and DFS, **Backtracking :** The general method, 8 – Queens problem, Graph coloring, Hamiltonian cycles

Text Books

1. Ellis Horowitz, SartajSahni, SanguthevarRajasekaran, "Computer Algorithms", Silicon Press Publication, 2nd edition, 2007
2. Gilles Brassard and Paul Bratley, "Fundamentals of Algorithmics" , Prentice Hall, Inc Publication, 2nd edition, 1997.

References

1. Thomas H. Cormen , Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", The MIT Press; 3rd edition, 2009.
2. Robert Sedgewick, Kevin Wayne, "Algorithms", Addison-Wesley Professional Publication, 4th edition, 2011.
3. Gayle Laakmann McDowell, "Cracking the Coding Interview: 150 Programming Questions and Solutions", CareerCup Publication, 5th Revised & enlarged edition, 2011.

Course Contents and Lecture schedule

Sl.No	Topic	No.Of Lectures
1.	**Introduction**	
1.1	Space complexity	1
1.2	Time complexity	1
1.3	Asymptotic notation	1
1.4	NP Complete	1
1.5	NP Hard	1
2	**Divide and conquer**	
2.1	Recurrence Relations	1
2.1.1	Homogeneous recurrences	
2.1.2	In homogeneous recurrences	1
2.1.3	Change of variables	1
2.1.4	Master theorem	1
2.2	Applications	1
2.2.1	Binary search	
2.2.2	Merge sort	1
2.2.3	Quick sort	1
2.2.4	Strassen's matrix multiplications	1
3	**The Greedy method -** General method	1
3.1	Knapsack problem	2
3.2	Job sequence with deadlines	1
3.3	Minimum cost spanning trees	2
3.4	Single source shortest paths	1
4	**Dynamic Programming -** General method	1

4.1	Multistage graphs	2
4.2	All Pair shortest paths	1
4.3	0/1 Knapsack	1
4.4	The travelling sales man problem	1
5	**Basic traversal and search techniques**	
5.1	Techniques for binary trees	2
5.2	Connected components and spanning trees	2
5.3	Biconnected components and DFS	2
6	**Backtracking**	
6.1	The general method	1
6.2	8 – Queens problem	2
6.3	Graph coloring	1
6.4	Hamiltonian cycles	2
	Total Lectures	**40**

Course Designers

1. M.K.Kavitha Devi mkkdit@tce.edu
2. S.Karthigaskait@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T43	3	1	--	4

T43 Operating Systems: Principles 3:1

Preamble: This course describes the fundamental concepts behind operating systems for general-purpose computers and covers the major components of operating systems.Particular emphasis is given to three major OS subsystems: process management (processes, threads, CPU scheduling, synchronization, and deadlock), memory management (segmentation, paging, swapping), and file systems. This course also covers the design of device manager. Students will be exposed to Linux, a modern, real time operating System.

Program Outcomes addressed

Graduates will demonstrate

a. an ability to apply knowledge of engineering, information technology, mathematics and science.

c. an ability to design a system or component, or process to meet stated specifications.

d. an ability to identify, formulate and solve engineering problems.

Competencies

Students will be able to

1. Manage the processes by scheduling the processes efficiently, modeling multithreads, synchronizing the processes and handling deadlocks and apply the knowledge in Linux system
2. Appraise the effectiveness of the different memory and disk management schemes and apply the knowledge in Linux system
3. Examine the performance issues of the I/O systems and apply the knowledge in Linux system
4. Implement a device driver
5. Specify the access controls to be imposed for various programs, processes, users to the resources
6. Enforce a secured system and apply the knowledge in Linux system

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3	End-semester examination
1	Remember	20	20	20	15
2	Understand	40	40	40	25
3	Apply	20	20	20	30
4	Analyze	20	20	20	30
5	Evaluate	0	0	0	0
6	Create	0	0	0	0

Course Level Learning Objectives:

Remember

1. List out three main purposes of an operating system?
2. Differentiate operating systems for mainframe computers and personal computers?
3. List the four steps that are necessary to run a program on a completely dedicated machine.
4. State the purpose of system calls.
5. Mention the three major activities of an operating system in regard to memory management.
6. Differentiate user-level threads and kernel-level threads.
7. Define preemptive and non-preemptive scheduling.
8. List three examples of deadlocks that are not related to a computer system environment.

Understand

1. Some early computers protected the operating system by placing it in a memory partition that could not be modified by either the user job or the operating system itself. Describe two difficulties that you think could arise with such a scheme.
2. The Sun UltraSPARC processor has multiple register sets. Describe the actions of a context switch if the new context is already loaded into one of the register sets. What else must happen if the new context is in memory rather than in a register set and all the register sets are in use?
3. Palm OS provides no means of concurrent processing. Discuss three major complications that concurrent processing adds to an operating system.

4. Assume an operating system maps user-level threads to the kernel using the many-to-many model and the mapping is done through LWPs. Furthermore, the system allows developers to create real-time threads. Is it necessary to bind a real-time thread to an LWP? Explain.
5. Describe the following memory allocation algorithms:
   a) First Fit
   b)Best Fit
   c)Worst Fit
6. Interpret the use of a device manager in Operating Systems.

Apply

1. Consider the interprocess-communication scheme where mailboxes are used.
   a)Suppose a process P wants to wait for two messages ,one from mailbox A and one from mailbox B. Illustrate the sequence of 'send' and 'receive' to be executed.
   b)Show the sequence of 'send' and 'receive' to be executed by 'P' if P wants to wait for two messages ,one from mailbox A and one from mailbox B?
2. Write a multithreaded Pthread program that generates the Fibonacci series. This program should work as follows: the user will run the program and will enter on the command line the number of Fibonacci numbers that the program is to generate. The program will then create a separate thread that will generate the Fibonacci numbers.
3. A CPU-scheduling algorithm determines an order for the execution of its scheduled processes. Given 'n' processes to be scheduled on one processor, how many different schedules are possible? Give a formula in terms of 'n'. Consider the following set of processes, with the length of the CPU-burst time given in milliseconds:

Process	Burst time	Priority
P1	10	3
P2	1	1
P3	2	3
P4	1	4
P5	5	2

The processes have arrived in the order P1, P2, P3, P4, P5 all at time 0.

a) Draw four Gantt charts illustrating the execution of these processes using FCFS, SJF, a nonpreemptive priority (a smaller priority implies a higher priority) and RR (quantum=1) scheduling.

b) Find out the turnaround time of each process for each of the scheduling algorithms in part a?

c) Find out the waiting time of each process for each of the scheduling algorithms in part a?

4. A barbershop consists of a waiting room with 'n' chairs and the barber room containing the barber chair. If there are no customers to be served, the barber goes to sleep. If a customer enters the barbershop and all chairs are occupied, then the customer leaves the shop. If the barber is busy and all chairs are available, then the customer sits in one of the chairs. If the barber is asleep, the customer wakes up the barber. Write a program to coordinate the barber and the customers.
5. Consider a system consisting of processes P1, P2, P3, ... Pn each of which has a unique priority number. Write a monitor that allocates three identical line printers to these processes, using the priority numbers for deciding the order of allocation.
6. Implement the deadlock-detection algorithm.
7. Consider a system consisting of four resources of the same type that are shared by the three processes, each of which needs at most two resources. Show that the system is deadlock-free.
8. Consider the system where a program can be separated into two parts: code and data. The CPU knows whether it wants an instruction (instruction fetch) or data (data fetch or store). Therefore, two base-limit register pairs are provided: one for instructions and one for data. The instruction base-limit register pair is automatically set to read only, so programs can be shared among different users. Examine the advantages and disadvantages of this scheme.

Analyze

1. A CPU-scheduling algorithm determines an order for the execution of its scheduled processes. Given 'n' processes to be scheduled on one processor, how many different schedules are possible? Give a formula in terms of 'n'. Consider the following set of processes, with the length of the CPU-burst time given in milliseconds:

Process	Burst time	Priority
P1	8	2
P2	4	1
P3	2	5
P4	1	4

P5 5 3

The processes have arrived in the order P1, P2, P3, P4, and P5 all at time 0.

Identify a schedule that results in the minimal average waiting time.

2. Consider a system consisting of four resources of the same type that are shared by three processes, each of which needs at most two resources. Investigate the deadlock freeness of the system.
3. Given memory partitions of 100KB,500KB,200KB,300KB and 600KB (in order), how would each of the first-fit, best-fit and worst-fit algorithms place processes of 212KB,417KB,112KB and 426KB (in order)?Which algorithm makes the most efficient use of memory? Justify your answer.
4. Consider the following page reference string:

1,2,3,4,2,1,5,6,2,1,2,3,7,6,3,2,1,2,3,6.

Examine the number of page faults which occur for the following replacement algorithms, assuming one, two, three, four, five, six, or seven frames?Remember all frames are initially empty, so your first unique pages will all costone fault each.

- LRU replacement
- FIFO replacement
- Optimal replacement

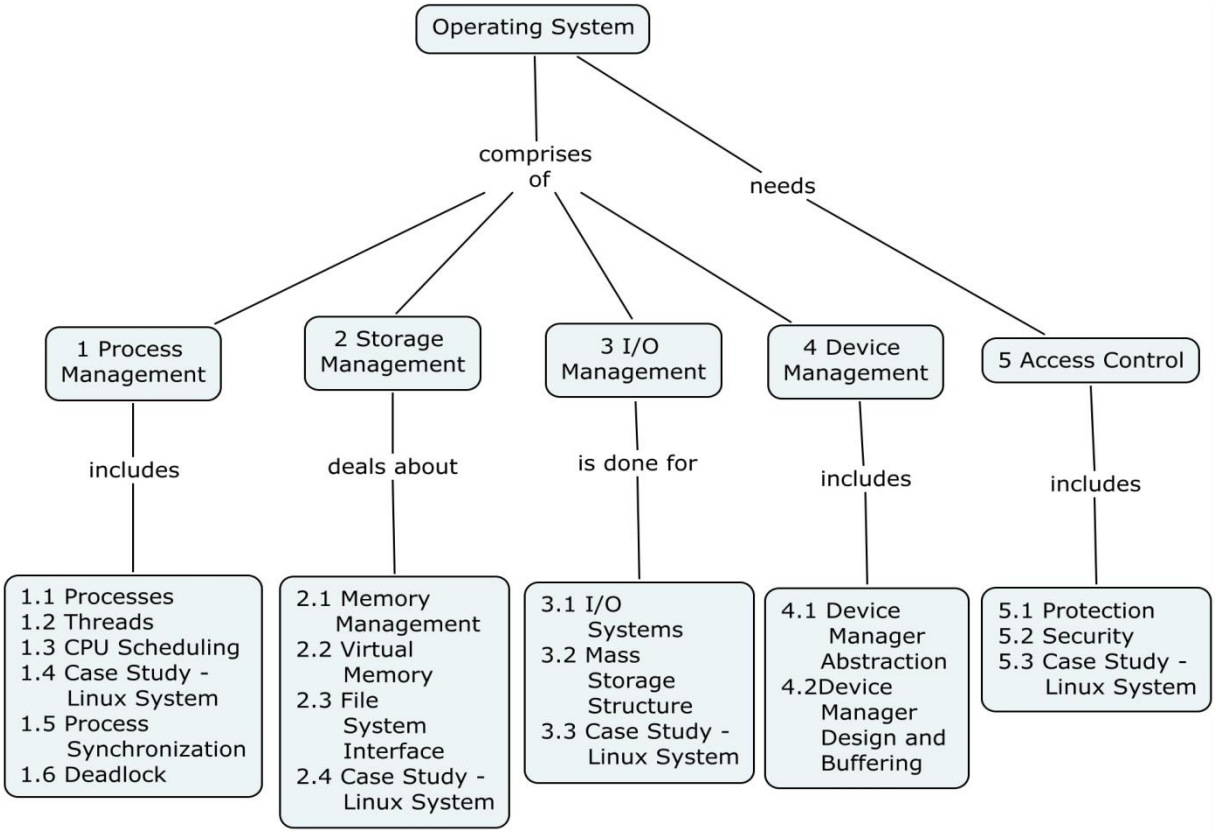
5. Consider a system that supports the strategies of contiguous, linked, and indexed allocation. Which strategy is best utilized for a file? Justify you answer
6. Suppose that a disk drive has 5000 cylinders, numbered 0 to 4999.the drive is currently serving a request at cylinder 143, and previous request was at cylinder 125,the queue of pending request in FIFO order is

86, 1470, 913, 1774, 948, 1509, 1022, 1750, 150

Compare and contrast the following algorithms with respect to the total distance (in cylinders) that the disk arm moves to satisfy the entire pending request (Assume that the disk arm starts from the current head position)

a. FCFS
b. SSTF
c. SCAN
d. LOOK
e. C-SCAN
f. C-LOOK

Concept Map



Syllabus:

Overview Types of Operating System **-** Operating System Structures **-** Operating System Components - System Services **- Process Management -** Processes – Process scheduling – Operations on processes – Threads - Multithreading models – Threading issues – CPU Scheduling - Scheduling criteria – Scheduling algorithms –Case Study-Linux System**-** Process Management-Scheduling-Process synchronization - The critical section problem – Semaphores – Classic problems of synchronization – Critical regions – Monitors – Deadlock - Deadlock characterization – Methods for handling deadlocks – Deadlock Prevention-Deadlock Avoidance-Deadlock Detection-Recovery from deadlock **Storage Management -** Memory Management – Swapping – Contiguous memory allocation – Paging – Segmentation – Segmentation with paging - Virtual Memory - Demand paging – Process creation – Page replacement – File system Interface - File concept, Access methods, Directory structureFile System Structure - File System Implementation - Directory ImplementationAllocation Methods - Free Space Management -Case Study - Linux System**-** Memory Management-File System - **I/O management -** I/O Systems **-** Kernel I/O subsystem – Mass-Storage Structure - Disk Structure-Disk scheduling – Disk management – Swap-Space management – Case Study - Linux System **-** Input

and Output - **Device management -** Device manager Abstraction - Device manager design and Buffering **– Access Control –** Protection **-** Goals of protection – Domain of Protection – Access Matrix – Implementation of Access Matrix **–** Security –The Security problem - User Authentication – Program Threats – System Threats **–** Security **-** Case Study-Linux System**-** Security. Operating System Standards

TEXT BOOK:

1. Silberschatz, Galvin and Gagne, "Operating System Concepts", Sixth Edition, JohnWiley & Sons Inc 2003.
2. Gary Nutt, "Operating Systems", Third Edition, Addison Wesley, 2004.

REFERENCES:

1. Andrew S. Tanenbaum, "Modern Operating Systems", Third Edition, Addison Wesley, 2008.
2. H M Deital, P J Deital and D R Choffnes, "Operating Systems", Third Edition,Pearson Education,2004.

Course Contents and Lecture schedule

S.No	**Topic**	**No of Lectures**
0	**Overview**	
0.1	**Types of Operating System**	1
0.2	**Operating System Structures**	
0.2.1	Operating System Components - System Services	1
1	**Process Management**	
1.1	**Processes**	
1.1.1	Process concept, Process scheduling, Operations on processes	2
1.2	**Threads**	
1.2.1	Multithreading models, Threading issues	1
1.3	**CPU Scheduling**	
1.3.1	Scheduling criteria, Scheduling algorithms	2
1.4	**Case Study-Linux System**	
1.4.1	Process Management - Scheduling	1

1.5	**Process Synchronization**	
1.5.1	The critical section problem, Semaphores	1
1.5.2	Classic problems of synchronization, Critical regions, Monitors	1
1.6	**Deadlock**	
1.6.1	Deadlock characterization, Methods for handling deadlocks	2
1.6.2	Deadlock Prevention-Deadlock Avoidance	2
1.6.3	Deadlock Detection-Recovery from deadlock	2
2	**Storage Management**	
2.1	**Memory Management**	
2.1.1	Swapping, Contiguous memory allocation	2
2.1.2	Paging, Segmentation, Segmentation with paging	2
2.2	**Virtual Memory**	
2.2.1	Demand paging	1
2.2.2	Process creation, Page replacement	1
2.3	**File system Interface**	
2.3.1	File concept, Access methods, Directory structure	1
2.3.2	File System Structure-File System Implementation	2
2.3.3	Directory Implementation	1
2.3.4	Allocation Methods-Free Space Management	2
2.4	**Case Study-Linux System**	
2.4.1	Memory Management - File Systems	1
3	**I/O Management**	
3.1	**I/O Systems**	
3.1.1	Kernel I/O subsystem	1
3.2	**Mass-Storage Structure**	

3.2.1	Disk Structure	1
3.2.2	Disk scheduling – Disk management	2
3.2.3	Swap-Space management	2
3.3	**Case Study-Linux System**	
3.3.1	Input and Output	1
4	**Device management**	
4.1	Device Manager Abstraction	1
4.2	Device Manager Design and Buffering	2
5	**Access Control**	
5.1	**Protection**	
5.1.1	Goals of protection – Domain of Protection	1
5.1.2	Access Matrix – Implementation of Access Matrix	2
5.2	**Security**	
5.2.1	The Security problem - User Authentication	1
5.2.2	Program Threats – System Threats	2
5.2.3	Security	1
5.3	**Case Study-Linux System**	1
5.3.1	Security and OS Standards	1
	Total Lectures	**46**

Course Designers:

1. R.Muthuselvi rmsit@tce.edu

2. R.Leena Sri rlsit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T44	3	0	--	3

T44 Computer Networks **3:0**

Preamble: The course aims at exploring the principles and design of computer networks featuring the Internet, covering aspects ranging from transmitting frames on a communication link and routing packets in a network to the design of network applications. An overview of wide area technologies, wired and wireless LANs is also provided.

Program Outcomes addressed

Graduates will demonstrate

a. An ability to apply knowledge of engineering, information technology, mathematics and science.
b. An ability to design and conduct experiments, as well as to analyze and interpret data.
c. An ability to design a system, component or process as per needs and specifications.
d. An ability to identify, formulate and solve engineering problems.
e. An ability to use techniques, skills, and modern engineering tools to implement and organize engineering works under given constraints.

Competencies

Students will be able to

1. Understand the fundamentals of computer networking, including protocol design, protocol layering, algorithm design, and performance evaluation.
2. Demonstrate an understanding of design principles and specific implemented protocols covering the application layer, transport layer, network layer, data link layer and Physical Layer of the Internet (TCP/IP) stack.
3. Demonstrate an understanding of wide area network technologies like Frame relay and Asynchronous transfer mode; and wired and wireless local area networks.
4. Write programs for simple network applications using socket programming.
5. Determine the performance of a given network by evaluating various Quality of Service metrics.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3	End-semester examination
1	Remember	30	20	20	10
2	Understand	30	20	20	20
3	Apply	40	30	40	40
4	Analyze	0	20	20	20
5	Evaluate	0	0	0	0
6	Create	0	10	0	10

Course Level Learning Objectives

Remember

1. What is framing?
2. Define round trip time of a packet.
3. State any two real time applications where UDP datagram service is preferred.
4. Enlist the functionalities of transport layer.
5. What is a sequence number?
6. Mention any few error messages generated by Internet Control Message Protocol.
7. Mention the physical devices operating at various layers of TCP/IP protocol suite.

Understand

1. Differentiate physical address, logical address and service point address.
2. Distinguish between connection oriented and connectionless services.
3. Explain the steps involved in distance vector routing to calculate the shortest path between two nodes.
4. Explain the protocol format of a TCP header with a neat sketch.
5. Explain the steps involved in Huffman encoding.
6. Discuss the functionalities of all layers in Frame relay network.

Apply

1. Use Cyclic Redundancy check to generate the code word for the data word 1010011010 and the divisor 10111. Verify the code word at the receiver side.
2. Compute the values of k and n in the Hamming code c(n,k) with d_{min} =3 for a data word of at least 16 bits.
3. Apply one's complement addition to generate the checksum for the message "INFORMATIONTECHNOLOGY".

4. In an IPV4 packet, if the value of HLEN is 1000 in binary, how many bytes of options are being carried away by this packet? An IPV4 packet has arrived with a few hexadecimal digits as follows 0x 450000280001000000102……Compute the number of hops the packet can travel before being dropped.
5. An organization is granted the address 211.17.180.01 /24, The administrator wants to create 32 subnets.
    a. Find the subnet mask.
    b. Find the number of addresses in each subnet.
    c. Find the first and last address in subnet 1.
    d. Find the first and last address in subnet 32.
6. A slotted ALOHA network transmits 200 bit frames with a shared channel with 200 kbps bandwidth. Calculate the throughput if the system produces
    a. 1000 frames/sec b. 500 frames/sec c.250 frames/sec.

Analyze

1. Does the efficiency of ATM using AAL3/4 depend on the size of the packet, assuming that there is no padding? Justify your answer.
2. Identify the changes to be made in ATM layer to accommodate an increase in data rate from 100 Mbps to 1 Gbps.
3. Examine whether a user can send data of 1 Mbps all the time in a frame relay network through T-1 line, with the following specifications: CIR = 1 Mbps; BC = 5 Million bits/5 sec and BE of 1 Million bits/5 seconds.
4. An ISP has a block of 1024 addresses. It needs to divide the addresses among 1024 customers. Is subnetting required? Explain your answer.
5. An IP datagram is carrying a TCP segment destined for the address 130.14.16.17/16. The destination port address is corrupted and it arrives at destination 130.14.16.19/16. How does the receiving TCP react to this error?
6. An IPv4 datagram has arrived with the following information in the header in hexadecimal 45 00 00 54 00 03 58 50 20 06 00 00 7C 4E 03 02 B4 0E 0F 02
    a. Examine the following
        i. Is the packet corrupted?
        ii. Are there any options?
        iii. Is the packet fragmented?
    b. How many routers can the packet still travel?
    c. What is the identification number and what is the type of service.

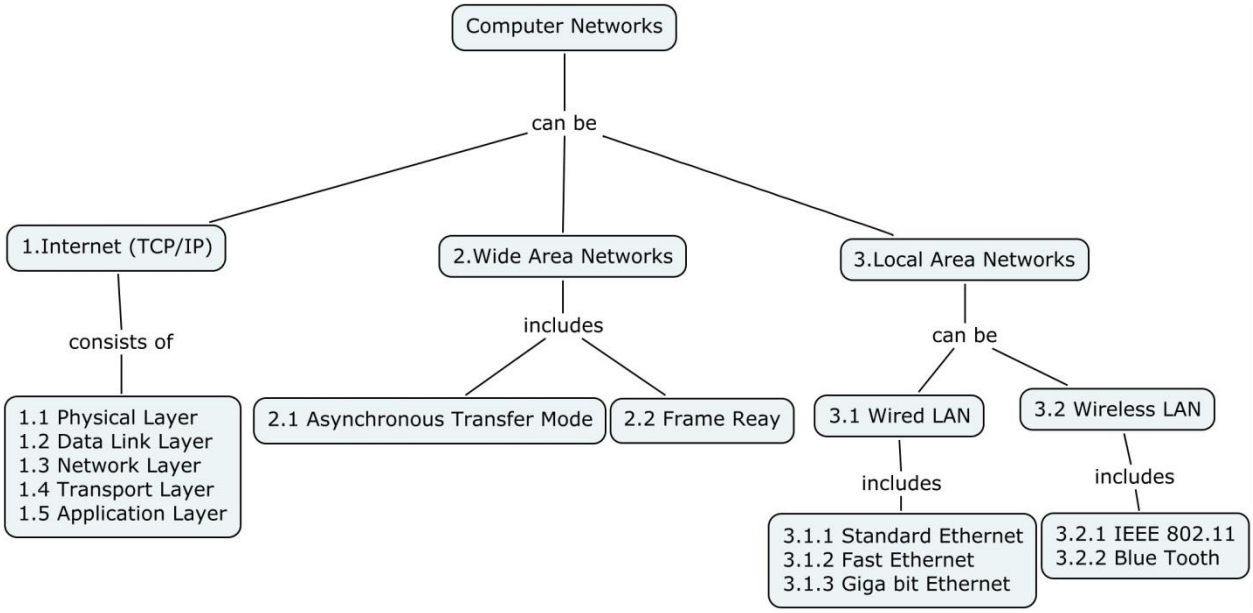
Create

1. An ISP is granted block of addresses 190.100.0.0 /16. The ISP needs to distribute these addresses to three groups of customers as follows:
    a. The first group has 64 sub-groups and each needs 256 addresses.
    b. The second group has 128 sub-groups and each needs 128 addresses.
    c. The third group has 128 sub-groups and each needs 64 addresses.

    Design sub blocks and find how many addresses are still available after these locations.
2. Design a bidirectional algorithm for GOBACK-N ARQ protocol using piggy backing.
3. Create a congestion window as a function of round trip time for the following scenario:

    Congestion control algorithm uses a linear increase and multiplicative decrease but not slow start. During communication packets 9,25,30,38 and 50 are lost. Assume a perfect time out mechanism that detects a lost packet exactly 1 RTT after it is transmitted.
4. Suggest suitable measures to address the limitations in sliding window Protocol.
5. Design hybrid architecture for a multimedia network using frame relay and ATM network.
6. Design an algorithm for bit stuffing to add and remove bits at the sender and receiver sites.

Concept Map



Syllabus:

Types of Computer Networks: Local Area Network, Wide Area Network, Internet, Protocols and standards. **Internet Model: TCP/IP Layers Physical Layer –** Data and signals, Modulation, Multiplexing, Switching, Topology, Transmission mode. **Data link Layer –** Framing, Physical Addressing, Flow control - noisy and noiseless channels, Error Control –Error detection and Error correction codes, Access control-Random Access, Controlled Access, Channelization Protocols. **Network Layer –** Logical Addressing , Routing – Distance Vector, Link state algorithms, Network Layer Protocols –ARP, RARP, ICMP, IGMP. **Transport Layer** – Service point addressing, segmentation and reassembly, Connection control – TCP and UDP, Flow control, TCP congestion control, Congestion avoidance mechanisms, Quality of service, Error control. **Application Layer** – Dialog control, Synchronization, Translation – presentation Formatting, Data Compression, Network security. **Local Area Networks: Wired LAN** –switched Ethernet, Fast Ethernet, Giga bit Ethernet **Wireless LAN** – IEEE 802.11, Blue tooth. **Wide area Technologies:** Asynchronous transfer mode – Physical layer, ATM layer, ATM adaptation layer. Frame relay- Physical Layer, Data link layer.

Textbook

1. BehrouzA.Foruzan, "Data Communication and Networking", Tata McGraw Hill, Fourth Edition, 2009.

Reference Books

1. Larry L.Peterson and Bruce S. Davie, "Computer Networks – A systems Approach" Fourth Edition, Morgan Kaufmann Publishers, 2007.
2. Andrew.S.Tanenbaum, "Computer Networks", Prentice Hall , Fourth Edition ,2009
3. William Stallings, "Data and Computer Communications", Pearson Education, Sixth Edition, 2007.
4. Todd Lammle, "CCNA Cisco Certified Network Associate Study Guide", Seventh Edition, 2011.

Course Contents and Lectures schedule

0.	Fundamentals of Computer Networks	1
0.1	ISO OSI Model	2
1.	Internet – TCP/IP model	1
1.1	**Physical Layer**	

1.1.1	Data and Signals, Transmission Mode, Topology	1
1.1.2	Multiplexing	1
1.1.3	Switching	1
1.1.4	Modulation	1
1.2	**Data Link Layer**	
1.2.1 1.2.2	Framing Physical Addressing	1
1.2.3	Flow Control	2
1.2.4	Error Control	2
1.2.5	Access control	2
1.3	**Network Layer**	
1.3.1	Logical Addressing	2
1.3.2	Routing	2
1.3.3	Network Layer Protocols –ARP,RARP, ICMP,IGMP	2
1.4	**Transport Layer**	
1.4.1 1.4.2	Service point Addressing Segmentation and reassembly	2
1.4.3	Connection Control –TCP, UDP	3
1.4.4	Flow Control – Congestion Control, Quality of service	2
1.4.5	Error Control	1
1.5	**Application Layer**	
1.5.1 1.5.2	Dialog control Synchronization	1
1.5.3	Data Compression	1
1.5.4	Network Security	1
2	**Wide Area Networks**	
2.1	Asynchronous Transfer Mode	3

2.2	Frame Relay	3
3	**Local Area Networks**	
3.1	**Wired LAN**	
3.1.1	Switched Ethernet	2
3.1.2 3.1.3	Fast Ethernet Giga bit Ethernet	1
3.2	**Wireless LAN**	
3.2.1	IEEE 802.11	2
3.2.2	Blue tooth	2
	Total Lectures	**45**

Course Designers:

1. S.Muthuramalingam smrit@tce.edu

2. C.Jeyamala jeyamala@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T 45	4	0	--	4

T45 Database Principles and Design 4:0

Preamble: The course on Database Principles and Design aims to emphasize the need for design of database systems and provide an in depth coverage of various principles of database systems.

Program Outcomes addressed

a. Graduates will demonstrate an ability to design and conduct experiments, as well as to analyze and interpret data

b.Graduates will demonstrate an ability to design a system or component or process to meet stated specifications

c. Graduates will demonstrate an ability to identify, formulate and solve engineering problems

Competencies

1. Explain different components of Query language that perform different database Systems.
2. Understand the relational model and relational algebra expressions
3. Perform Query Processing optimizations and Transaction Processing.
4. Resolve database system issues.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3/End-semester examination
1	Remember	20	20	10
2	Understand	40	20	20
3	Apply	40	40	40
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	0	20	30

Course Level Learning Objectives

Remember

1.What is meant by Tuple relational calculus?

2.What is meant by Domain Relational Calculus?

3. What is a transaction in Database Systems?

4. What is meant by weak entity sets?
5. What is meant by serializabilty?
6. What is meant by functional Dependency?
7. What is meant by PCNF?
8. What is meant by semi join? Give an example
9. What is meant by Multiplicity?
10. What is the purpose served by Normalization?
11. Why can we have at most one primary or clustering index on a file, but several Secondary indexes?
12. What is meant by degree of relation?
13. What is meant by catastrophic failure?
14. What is Query Optimization?

Understand

1. Explain how disk access can be parallelized using RAID Technology?
2. Explain two phase commit protocol in DBMS?
3. Explain how queries could be optimized in Relational Algebra?
4. Explain how semantic queries can be optimized using selectivity and Cost estimation?
5. Explain the differences among a relationship instance, a relationship type, and a relationship set.
6. Discuss the two notations for specifying constraints on n-ary relationships, and what each can be used for?
7. List the various data abstraction concepts and the corresponding modeling concepts in the EER model.
8. What are the main similarities and differences between conceptual database modeling techniques and knowledge representation techniques?
9. Discuss the mechanism used to read data from or write data to the disk.
10. Discuss the techniques for allowing a hash file to expand and shrink dynamically. What are the advantages and disadvantages of each?
11. How are buffering and caching techniques used by the recovery subsystem?
12. Describe the write-ahead logging protocol.
13. Describe the three phases of the ARIES recovery method.
14. Describe the two-phase commit protocol for multidatabase transactions.
15. Discuss how recovery from catastrophic failures is handled.
16. Describe the process of Query Optimization with an example.
17. Explain the various steps in processing a Query with an example.

Apply

1.Consider a disk with the following characteristics (these are not parameters of any particular disk unit): block size *B* = 512 bytes; interblock gap size *G* = 128 bytes; number of blocks per track = 20; number of tracks per surface = 400. A disk pack consists of 15 double-sided disks.

a. What is the total capacity of a track, and what is its useful capacity?

b. How many cylinders are there?

c. What are the total capacity and the useful capacity of a cylinder?

d. What are the total capacity and the useful capacity of a disk pack?

2. Consider the following two sets of functional dependencies: *F* = {*A.C, AC .D, E.AD, E .H*} and *G* = {*A.CD, E .AH*}. Check whether they are equivalent?

3. Suppose that we use the deferred update protocol for airline reservation system. Show how the log would be different in the case of deferred update by removing the unnecessary log entries; then describe the recovery process, using your modified log. Assume that only REDO operations are applied, and specify which operations in the log are redone and which are ignored.

4. PARTS file with Part# as key field includes records with the following P# values: 23, 65, 37, 60, 46, 92, 48, 71, 56, 59, 18, 21, 10, 74, 78, 15, 16, 20, 24, 28, 39, 43, 47, 50, 69, 75, 8, 49, 33, 38. Suppose that the search field values are inserted in the given order in a B-tree of order p = 4 and = 3; show how the tree will expand and what the final tree will look like.

5. Specify the following queries on the database schema using the relational operators in Relational Algebra. Also show the result of each query as it wouldapply to the database of relational operations.

   a. Retrieve the names of all employees in department 5 who work more than 10 hours per week on the 'ProductX' project.

   b. List the names of all employees who have a dependent with the same first name as themselves.

   c. Find the names of all employees who are directly supervised by 'Franklin Wong'.

   d. For each project, list the project name and the total hours per week (by all employees) spent on that project.

6. A PARTS file with Part# as hash key includes records with the following Part# values: 2369, 3760, 4692, 4871, 5659, 1821, 1074, 7115, 1620, 2428, 3943, 4750, 6975, 4981, 9208. The file uses eight buckets, numbered 0 to 7. Each

bucket is one disk block and holds two records. Load these records into the file in the given order, using the hash function $h(K) = K$ mod 8. Calculate the average number of block accesses for a random retrieval on Part#.

7. Consider this query: Retrieve the SSNs of employees who work on at least those projects on which the employee with SSN = 123456789 works. This may be stated as (FORALL x) (IF P THEN Q), where
   - x is a tuple variable that ranges over the PROJECT relation.
   - P M employee with SSN = 123456789 works on project x.
   - Q M employee e works on project x.

   Express the query in tuple relational calculus, using the rules

   - **(x)(P(x)) M not(x)(not(P(x))).**
   - **(IF P THEN Q) M (not(P) or Q).**

Create

1. Design an EER schema for a database application that you are interested in. Specify all constraints that should hold on the database. Make sure that the schema has at least five entity types, four relationship types, a weak entity type, a superclass/subclass relationship, a category, and an n-ary (n > 2) relationship type.

2. Suppose that we have the following requirements for a university database that is used to keep track of students' transcripts:

   a. The university keeps track of each student's name (SNAME); student number (SNUM); social security number (SSN); current address (SCADDR) and phone (SCPHONE); permanent address (SPADDR) and phone (SPPHONE); birth date (BDATE); sex (SEX); class (CLASS) (freshman, sophomore, ..., graduate); major department (MAJORCODE); minor department (MINORCODE) (if any); and degree program (PROG) (B.A., B.S., ..., PH.D.). Both SSSN and student number have unique values for each student.

   b. Each department is described by a name (DNAME), department code (DCODE), office number (DOFFICE), office phone (DPHONE), and college (DCOLLEGE). Both name and code have unique values for each department.

   c. Each course has a course name (CNAME), description (CDESC), course number (CNUM), number of semester hours (CREDIT), level (LEVEL), and offering department (CDEPT). The course number is unique for each course.

   d. Each section has an instructor (INAME), semester (SEMESTER), year (YEAR), course (SECCOURSE), and section number (SECNUM). The section

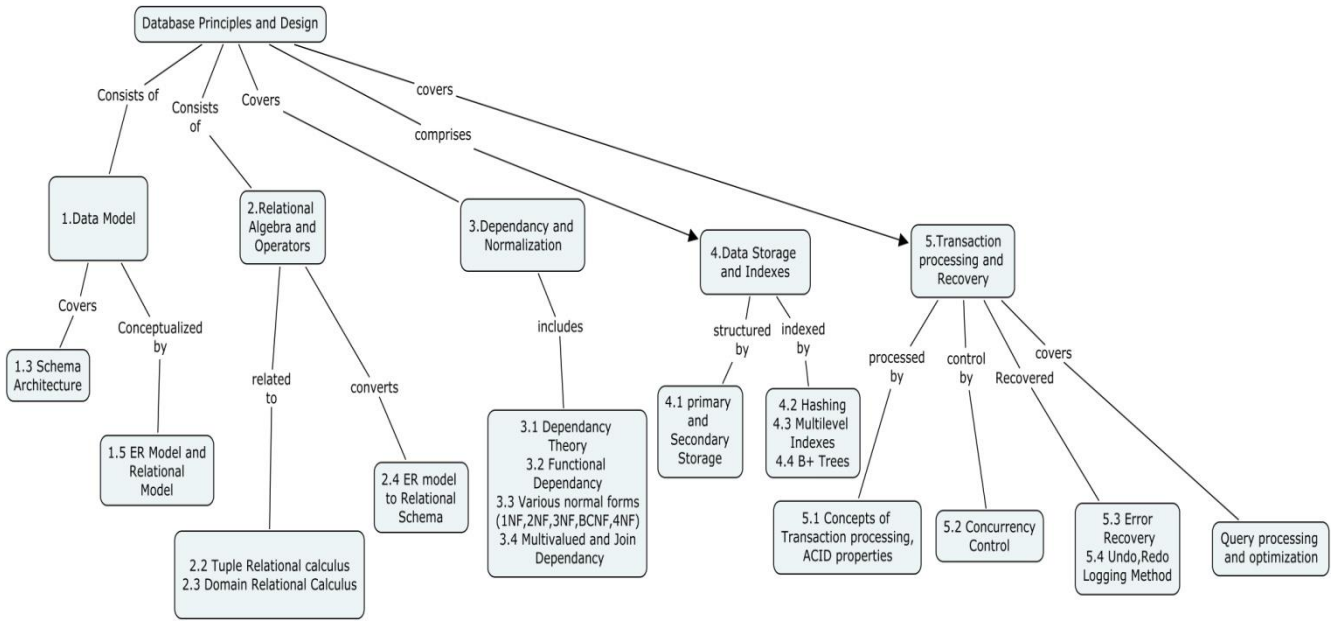
number distinguishes different sections of the same course that are taught during the same semester/year; its values are 1, 2, 3, ..., up to the total number of sections taught during each semester.

e. A grade record refers to a student (SSN), a particular section, and a grade (GRADE).

Design a relational database schema for this database application. First show all the functional dependencies that should hold among the attributes. Then design relation schemas for the database that are each in 3NF or BCNF. Specify the key attributes of each relation. Note any unspecified requirements, and make appropriate assumptions to render the specification complete.

3. Composite and multivalued attributes can be nested to any number of levels. Suppose we want to design an attribute for a STUDENT entity type to keep track of previous college education. Such an attribute will have one entry for each college previously attended, and each such entry will be composed of college name, start and end dates, degree entries (degrees awarded at that college, if any), and transcript entries (courses completed at that college, if any). Each degree entry contains the degree name and the month and year the degree was awarded, and each transcript entry contains a course name, semester, year, and grade. Design an attribute to hold this information and normalization with query optimization with your own algorithm of deferred update.

Concept Map



Syllabus

Introduction-General Introduction to database systems ,Database-DBMS distinction, Approaches to building a database, Data models, Database management system, Three-schema architecture of a database, Challenges in building a DBMS, Various components of a DBMS,ER Model, Relational Data Model **Relational algebra operators**-Relational algebra operators, Tuple relation calculus, Domain relational calculus, Converting the database specification in E/R notation to the relational schema **Dependencies and Normal forms**-Importance of a good schema design, Problems encountered with bad schema designs, Motivation for normal forms, dependency theory - functional dependencies, Armstrong's axioms for FD's,Closure of a set of FD's, Minimal covers, Definitions of 1NF, 2NF, 3NF and BCNF, Decompositions, Algorithms for 3NF and BCNF normalization of Control Structures,Multi-valued dependencies and 4NF, Join dependencies and definition of 5NF **Data Storage and Indexes**- File organizations, Primary and Secondary index structures, Various index structures - hash-based, Dynamic hashing techniques , Multi-level indexes, B+ trees **Transaction processing and Error recovery**-Concepts of transaction processing, ACID properties, Query Processing and Optimization, Concurrency control, Locking based protocols for Concurrency Control ,Error recovery and logging, Undo, Redo, Undo-redo logging and recovery methods.

Text Book:

1. RamezElmasri and ShamkantB.Navathe, "Fundamentals of Database Systems", 4th edition, Pearson Education, 2004.

Reference Book:

1.Abraham Silberschatz, Henry F.Korth and Sudarshan, "Database System Concepts", 5th edition, Mcgraw-Hill, 2006.

Course Schedule

No.	Topic	No. of Lectures
1	**Introduction**	
1.1	General Introduction to database systems	1
1.2	Database-DBMS distinction, Approaches to building a database	1
1.3	Data models, Database management system, Three-schema architecture of a database	1
1.4	Challenges in building a DBMS, Various components of a DBMS	1
1.5	ER Model, Relational Data Model	1
2.	**Relational algebra operators**	
2.1	Relational algebra operators, Tuple relation calculus	1
2.2	Domain relational calculus	1
2.3	Converting the database specification in E/R notation to the relational schema	1
3	**Dependencies and Normal forms**	
3.1	Importance of a good schema design, Problems encountered with bad schema designs,	2
3.2	Motivation for normal forms, dependency theory - functional dependencies,Armstrong's axioms for FD's,Closure of a set of FD's, Minimal covers	1
3.3	Definitions of 1NF, 2NF, 3NF and BCNF	1
3.4	Decompositions, Algorithms for 3NF and BCNF	2

No.	Topic	No. of Lectures
	normalization of Control Structures	
3.5	Multi-valued dependencies and 4NF, Join dependencies and definition of 5NF	2
4	**Data Storage and Indexes**	
4.1	File organizations, Primary and Secondary index structures	1
4.2	Various index structures - hash-based, Dynamic hashing techniques	1
4.3	Multi-level indexes.	1
4.4	B+ trees	1
5	**Transaction processing and Error recovery**	
5.1	Concepts of transaction processing, ACID properties	2
5.2	Query Processing and Optimization	2
5.3	Concurrency control, Locking based protocols for Concurrency Control	1
5.4	Error recovery and logging	1
5.5	Undo, Redo, Undo-redo logging and recovery methods	2

Course Designers:

1.G.Prakash prakashguru@tce.edu

2. S. Senthamarai Kannan sskit@tce.edu

3. S. Appavu alias Balamurugan sbit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T 46	3	0	--	3

T46 System and Data center Administration 3:0

Preamble: System administration involves the management of various resources as well as the user accounts of a network. In addition to the management of user accounts, file systems and devices,it offers backup- and security- services. This course also emphasizes the characteristics of a strong data center and explores primitive solutions for monitoring and managing the data center. It also touches upon the remote data center managementin case of disasters and hardware failures.

Program outcomes addressed:

a. Graduates will demonstrate an ability to use techniques, skills, and modern engineering tools to implement and organize engineering works under given constraints.

b. Graduates will demonstrate an ability to apply knowledge of engineering, information technology, mathematics and science.

c. Graduates will demonstrate an ability to design and conduct experiments, as well as to analyze and interpret data.

Competencies:

1. Understand the fundamental operations pertaining to user group, file system, kernel, and backup administration.
2. Design and configure the system network in compliance with the protocols like IPv6, DHCP etc.
3. Realize the security administration through various security strategies like Kerberos, firewalls etc.
4. Enumerate the characteristics of the DataCenter like Security, compliance, uptime, performance etc.
5. Understand the data center operations and maintenance best practices for critical facilities.

Assessment Pattern:

	Bloom's Category	Test 1	Test 2	End-semester examination
1	Remember	20	10	10
2	Understand	50	50	30
3	Apply	30	30	40
4	Analyze	0	0	10
5	Evaluate	0	0	0
6	Create	0	10	10

Course Level Learning Objectives

Remember

1. What is an OS? Explain features of Linux OS.
2. List the general-purpose utilities available in Linux and explain any three of them.
3. What are the tasks of a system administrator?
4. How will you set permissions for Owner, Group and Public?
5. What is a System Call? Explain any four system calls inLinux in detail.
6. List two IP routing protocols which can be used in a medium size network under a single administration.
7. What is the effect of the "cd" command executed without any argument?
8. What is the output of "netstat -ta" command?
9. List the important Linux directories and briefly describe them.
10. List any five TCP/IP-related protocols and describe them in brief.
11. What are the general methods of implementing network security by firewalls? Elaborate.
12. What are the core elements of a datacenter?
13. What are the management tasks in data center administration?
14. Which command will display all processes including system and user?
15. What is meant by system variable? Which command is used to display it on screen? Give description of all variables.

Understand

1. Explain Command line Arguments.
2. How can the File system mounting and Unmounting be accomplished?
3. Enumerate the necessities for backup and restoration?
4. Explain common runlevel values on RHEL and how to check current runlevel.
5. Specify the benefits of shadow passwords with example.
6. Write the difference between low level I/O and standard library I/O functions. Give suitable examples.
7. Distinguish the functionalities of the three main shells available in Linux.
8. Describe the character and block devices.
9. List down the factors on which the Linux Operating System proves to be superior to other OS.
10. Differentiate the operation of hard-link and soft-link withhelp of block diagram.

11. Describe the complete fork-execute life cycle of a process with a block diagram.
12. Explain the following terms:
    i) Sticky bit mode ii) SUID iii) SGID
13. How do you monitor a data center that will scale for 10 years of growth?

Apply

1. Write a script to kill the last background process without knowing its PID and also to kill the current shell
2. Write a shell script to sort 6th, 7th column and 4th filed for file1.txt and output copy to file2.txt.
3. Write a shell script to cut 3rd filed from emp1.lst file and sort that filed first and list all records without duplicated
4. Write a shell script to get first 20 records from emp1.lst file and print the records with the following format: add double space, number the records, add header "Listof Employees", and increase left page margin by 10 spaces.
5. Write a shell script to display particular event number and add trigger to execute it.
6. Write a shell script to convert the contents of file f1.txt to uppercase and to change the permission of hidden files in recursive manner.
7. Write a shell script that can be used to list all the available free space of all the mounted file systems.
8. Write a shell script to create a directory under /tmp for each user listed in the password file.
9. Write a shell script to show the list of users who are working in the particular time interval.
10. Write a shell script to shut down some of the user accounts who are visiting unwanted/ restricted sites in the internet.

Analyze

1. As a system administrator you are asked to install two operating systems, Windows XP and Fedora Core 6 with GRUB as the boot loader on a host. Which operating system should you install first? Why?
2. You want to shutdown a machine running Linux at 5.00 pm. on a given day.What is the command that you should issue to achieve this task?
3. Due to an attack, one of the systems in the network goes down, how will you rectify it based on the topologies like ring, bus, star etc.? Analyze the factors.

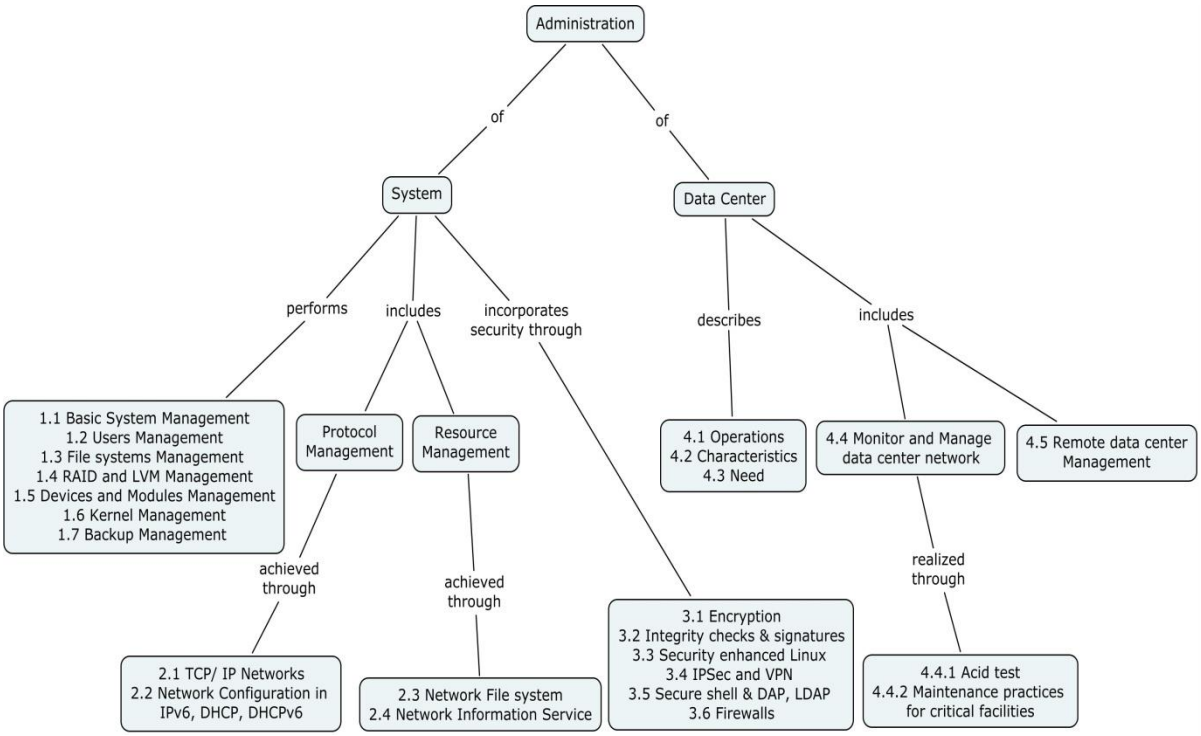
4. Identify the various situations under which "Email" and "file transfer", may become unsecured communication methods. Why?
5. A system is able to ping within the network but not outside. Why?
6. You installed a new AD domain, and the new DC has not registered its SRV records in DNS. What may be the possible causes?
7. Looking at IPSec-encrypted traffic with a sniffer, what packet types do you see?
8. Analyze the factors for forcibly remove AD from a server, and what do you do later? Can you get user passwords from the AD database?
9. Elaborate how to connect Active Directory to other 3rd-party Directory Services?
10. How can you determine whether GPO was applied or not for a user?

Create

1. Create a directory structure of university management and perform the following:
    a. List all directories in /university/cse
    b. Set the shortcut of university directory as univ.
    c. Create the files and append it with some text using only cat command
    d. Show the difference between cp and mv
    e. Set the permissions as u=rw, g=rx and o=r
    f. Change permission of the user and groups from same console of the user
    g. Show the exit status for Directory.
2. Create a user "student" in such a way that he changes his password in 30 days and receives a notice 3 days in advance. The home directory should be created automatically.
3. Configure a system in such a way that
    a. after a wrong password has been entered, a user must wait four seconds before he can retry login.
    b. the user passwords (for new accounts) are valid for a maximum of 50 days
    c. the permissions of newly created user directories are set to rwx ------
4. Create a user group for accessing the server in your department laboratory with special permissions depending on the user category like head, staffs, students etc.

5. Design and configure a network with 5 machines in a lab with different access permissions depending on the special role of the machines in the network.
6. Create a user group based on their SIG and provide permissions to access the services and resources based on that.
7. Create a set of users and send an alert regarding their interest, to all the users daily in a particular time. Allocate the memory to all of them based on their designation and set access permissions to all.
8. Configure a network with a set of systems (Computers and Peripherals) and incorporate restrictions on their use, by the faculty members of the department.

Concept Map:



Syllabus:

System Administration: Basic System management, Users management, File Systems management, RAID and Logical Volume Manager, Devices and Modules management, Kernel management, Backup Management. **Protocol Management:** TCP/ IP Networks, Network Auto configuration with IPv6, DHCPv6

and DHCP. **Resource Management:** Network File System, Network Information Service. **Security Administration:** Encryption, Integrity Checks and Signatures, Security Enhanced Linux, IPSec and Virtual Private Networks, Secure Shell and DAP, LDAP, Firewalls. **DataCenter Administration:**Characteristics, Operations, Need, Monitor and Manage Data Center Network, Acid Test criteria, Data center maintenance best practices for critical facilities, Remote data center management.

Text Books:

1. Richard Peterson, "The Complete Reference: Linux", McGraw Hill Publications, Sixth Edition, 2008. (System Administration – Part VII ,Part VIII, Security Administration – Part V)
2. DataCenter Administration related course materials from IBM Systems Journal, Dell, NetSuite, TechTarget Datacenter Media, etc.

Reference Books:

1. Nicholas Wells, "Guide to Linux Installation and Administration", Vikas publishing house, 2000.
2. Aelig, Leen Frisch, "Essential System Administration, Tools and Techniques for Linux and Unix Administration", Third Edition, O'Reilly Media Publishers, 2002.

Course contents and Lecture Schedule:

No.	Topic	No. of Lectures
1	**System Administration**	
1.1	Basic system Management	1
1.2	Users Management	2
1.3	File systems Management	2
1.4	RAID and LVM	2
1.5	Devices and Modules	2
1.6	Kernel Management	1
1.7	Backup Management	2
2.	**Protocol and Resource Management**	
2.1	TCP/ IP Networks	2

2.2	Network Auto configuration with IPv6, DHCPv6 and DHCP	2
2.3	Network File System	2
2.4	Network Information Service	2
3	**Security Administration**	
3.1	Encryption	2
3.2	Integrity Checks and Signatures	1
3.3	Security Enhanced Linux	2
3.4	IPSec and Virtual Private Networks	2
3.5	Secure Shell and DAP,LDAP	2
3.6	Firewalls	1
4	**DataCenter Administration**	
4.1	Data center operations	1
4.2	Characteristics of a strong data center.	1
4.3	Need for data center management	1
4.4	Monitor and manage data center network	1
4.4.1	Acid test criteria for the data center	1
4.4.2	Data center maintenance best practices for critical facilities	1
4.5	Remote data center management	2

Course Designers:

Mr. S.Srinivasan ssnit@tce.edu

Ms. S.Geetha sgeetha@tce.edu

Mr. P.Karthikeyan karthikit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T 47	--	--	1	1

T47 Network Programming Lab **0:1**
(Common to CSE and IT, C47, T47)

Preamble: The global connectivity can be achieved through computer networks. It is important to understand the function of computer networks andthe knowledge about hardware and software requirements of networks is essential. This course emphasizes the various components to make a network operational.

LIST OF EXPERIMENTS

1. Write a program to obtain an IP Address of a given/any host.
2.Write a program to implement a time server.
3.Write a program to illustrate a simple client/server communication.
4. Write a program to implement remote method invocation.
5. Write a program to implement ECHO and PING commands.
6.Write a program to implement a file transfer using TCP/IP.
7. Write a program to find which port is currently used/ scanning the port.
8. Configuration of various networks components - connections, BNC, RJ-45, I/O box, Cables, Co-axial, twisted pair, UTP, NIC (network interface card), Switch, hub.
9. Establishment of a LAN/ Preparing of network cables
10. Use of protocols in establishing LAN
11. Installation of network device drivers
12.Installation of networks (Peer to Peer Networking client server interconnection)

Course Designers:

1. S. Srinivasan ssnit@tce.edu
2. S. Muthuramalingam smrit@tce.edu
3. S. Gilbert Nancy gilbertnancyl@tce.edu
4. P.Subathra pscse@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T 48	0	0	1	1

T48 System Administration Lab 0:1

Preamble: In this laboratory, the students will perform various administering tasks like installing, supporting, and managing servers and other computer systems. Specifically their tasks encompass:

- Analyzing system logs and identifying potential issues with computer systems.
- Performing routine audits of systems and software.
- Performing backups.
- Applying operating system updates, patches, and configuration changes.
- Installing and configuring new hardware and software.
- Adding, removing, or updating user account information, resetting passwords, etc.
- Incorporatingprimitive security services.
- Ensuring that the network infrastructure is up and running.

List of Experiments:

1. Server installation and configuration
2. Client installation and configuration
3. User management
4. File system management
5. Web server management
6. Mail server management
7. Backup Management
8. Configuring Printer
9. DHCP configuration
10. Schedule Tasking and run level operations
11. Implementing any Encryption/ Decryption algorithm

Course Designers:

1. S.Geetha sgeetha@tce.edu
2. P.Karthikeyan karthikit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T49	1	0	1	2

T49 PROFESSIONAL COMMUNICATION 1:1

(Common to ALL branches of B.E)

Subject codes: B49, C49, D49, E49, G49, T49

Preamble: Professional communication aims to develop Listening, Speaking, Reading and Writing skills in Engineering students' professional development contexts such as projects, competitive exams, organizational communication and soft skills.

Competencies: At the end of the course the students should be able to

Listening:

1. Listen and understand the project presentations, competitive exam exercises, organizational communication activities
2. Listen to the lectures on soft skills for practice.

Speaking:

1. Present project reports, self introduction
2. Participate in GD and interview in work context.

Reading:

1. Read and collect information for project report writing.
2. Read and understand the comprehension passages given in competitivexaminations.
3. Read and understand the company profile

Writing:

1. Write a project report adhering to proper format
2. Create a paragraph and essay using their own ideas
3. Write circulars, minutes of the meetings, and curriculum vitae

Assessment Pattern:

	Internal (50)	**External (100)**
1. Recall	10	10
2. Understanding	10	20
3. Application	10	20
4. Analysis	10	30
5. Evaluation	5	10
6. Creation	510	

Course Content:

1. Listening:

1.1Attention, understanding and responding

1.2 Project report writing, competitive exam exercises, organizational communication and soft skills practice

2. Speaking:

2.1 Planning, preparation and presentation

2.2 Project report, self introduction, GD and interview

3. Reading:

3.1 Rapid reading and reference skills

3.2 Project reports, competitive exam exercises and company profiles

4. Writing:

4.1. Structure

4. 1.1 Sentence structure

4.1.2 Abstract writing

4.1.3. CV writing

4.1.4. Project report writing

4.2 Organizational Communication

4.2.1 Circulars

4.2.2 Minutes of the meeting

Syllabus:

Listening: Listening to Project presentation: Asking Questions, Listening test as conducted in TOEFL and BEC, Listening in the context of Organizational communication and Soft skill practice.

Speaking: Project presentation skill, Speaking in the context of Group Discussion, Interview, TOEFL and BEC Exam Spoken Test, Speaking in the work Contexts : Self introduction, Mini Presentation

Reading: Reference Skills for Project Report Writing: Topic selection, Data Collection. Rapid Reading, Reading comprehension tests conducted in CAT, TOEFL, GRE and BEC, Reading skills in work situation: Company Profile

Writing: Project Report Writing : Format, Abstract, Bibliography, Structure : Sentence structure, CV Writing, Writing in Work context : Circulars, Minutes of the meeting

References

1. Tony Lynch: Study Listening. Cambridge,CambridgeUniversity Press, 2007
2. Sangeeta Sharma and Binod Mishra: Communication Skills for Engineers and Scientists. New Delhi, PHI Learning Pvt. Ltd. 2009.

3. Hari Mohan Prasad and Uma Rani Sinha: Objective English for Competitive Examination. New Delhi, Tata McGraw – Hill, 2005
4. Bob Dignen, Steeve Flinders et. al.: Work and Life: English 365. Students Book 1,2 & 3. New Delhi, Cambridge, 2004.

List of Lecture sessions:

Listening:

1. Effective listening skills
2. Nature of listening tests in competitive examinations
3. Introduction of soft skills

Speaking:

1. Introduction of Presentation skills
2. Suggestions for speaking tests in competitive exams
3. How to participate in GD
4. Interview techniques

Reading:

1. Rapid reading techniques
2. Reference skills
3. Suggestions for reading tests in competitive exams

Writing:

1. Format of project report
2. Abstract of the project
3. Sentence structure
4. Organizational communication like sending circulars, writing minutes of the meetings
5. CV writing

List of Practice Sessions:

Listening:

1 Messages, descriptions, conversations and lectures

Speaking:

1. Self Introduction
2. Mini Presentation
3. GD
4. Interview

Reading:

1. Rapid reading practices
2. Comprehension exercises
3. Topic selection and data collection for project report

Writing:

1. Sentence structure
2. Abstract writing
3. Project Report Writing
4. Circulars
5. Minutes of the meeting
6. Model test

Course Designers

1. T. Sadasivansadasivan@tce.edu
2. S. Rajaramsreng@tce.edu
3. A. Tamil Selvitamilselvi@tce.edu

REVISED CURRICULUM AND DETAILED SYLLABI

FOR

B.Tech DEGREE (Information Technology) PROGRAM

FIFTH SEMESTER

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2012-2013

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Approved in BOS Meeting on 13.10.2012 Approved in 45 th Academic Council Meeting on 24.11.2012

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015

B.Tech. Degree (Information Technology) Program

SUBJECTS OF STUDY

(For the candidates admitted from 2012-2013 onwards)

FIFTH SEMESTER

Subject code	Name of the subject	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
T51	Discrete Mathematics	BS	4	-	-	4
T52	Software Engineering Methodologies	DC	3	1	-	4
T53	Information Storage and Management	DC	3	-	-	3
T54	Information Security and Assurance	DC	4	-	-	4
T55	Graphics and Visualization	DC	3	1	-	4
T56	Platform Technologies	DC	4	-	-	4
PRACTICAL						
T57	Platform Technologies Lab	DC	-	-	3	1
T58	Network Security Lab	DC	-	-	3	1
Total			**21**	**2**	**6**	**25**

BS : Basic Science
HSS : Humanities and Social Science
ES : Engineering Science
DC : Department Core

L : Lecture
T : Tutorial
P : Practical

Note:

1 Hour Lecture/Tutorial is equivalent to 1 credit
2/3 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015

B.Tech. Degree (Information Technology) Program

SCHEME OF EXAMINATIONS

(For the candidates admitted from 2012-2013 onwards)

FIFTH SEMESTER

S.No	Sub. code	Name of the subject	Duration of Terminal Exam in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	T51	Discrete Mathematics	3	50	50	100	25	50
2	T52	Software Engineering Methodologies	3	50	50	100	25	50
3	T53	Information Storage and Management	3	50	50	100	25	50
4	T54	Information Security and Assurance	3	50	50	100	25	50
5	T55	Graphics and Visualization	3	50	50	100	25	50
6	T56	Platform Technologies	3	50	50	100	25	50
PRACTICAL								
7	T57	Platform Technologies Lab	3	50	50	100	25	50
8	T58	Network Security Lab	3	50	50	100	25	50

* Continuous Assessment evaluation pattern will differ from subject to subject and for different tests. This will have to be declared in advance to students. The department will put a process in place to ensure that the actual test paper follow the declared pattern.

** Terminal Examination will be conducted for maximum marks of 100 and subsequently be reduced to 50 marks for the award of terminal examination marks.

Sub code	Lectures	Tutorial	Practical	Credit
T51	4	0	-	4

T51 Discrete Mathematics **4:0**

Preamble:

An engineering student needs to have some basic mathematical tools and techniques. This emphasizes the development of rigorous logical thinking and analytical skills of the student and appraises him the complete procedure for solving different kinds of problems that occur in engineering. Based on this the course aims at giving adequate exposure in the theory and applications of Set theory, logic and Automata theory.

Program Outcomes addressed

a. Graduates will demonstrate knowledge of Mathematics, Science and Engineering.

b. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

c. Graduates will develop confidence for self education and ability for life-long learning.

Competencies

At the end of the course the student should be able to

1. Check the validity of the arguments.
2. Understand how to construct correct mathematical arguments.
3. Design of computers and electrical circuit.
4. Check whether a particular combination of words is a valid sentence or not
5. Specify a well defined set of rule of syntax by which certain formal Languages such as programming languages can be constructed.

Assessment Pattern

	Bloom's category	Test 1	Test 2	Test 3 / End Semester Examinations
1	Remember	10	10	0
2	Understand	30	30	30
3	Apply	60	60	70
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	0	0	0

Course level learning objectives
Remember

1. Define a biconditional statement and draw its truth table
2. Show that $(P \wedge Q) \rightarrow (P \vee Q)$ is a tautology without constructing truth table.
3. Define term NAND
4. Show that following implication $[P \rightarrow (Q \rightarrow R)] \Rightarrow [(P \rightarrow Q) \rightarrow (P \rightarrow R)]$
5. Define minterm and maxterm.
6. Show that $A - B = A \cap B'$.
7. Show that by using mathematical induction $2+2^2+2^3+--------+2^n=2^{n+1}-2$
8. Define NFA with ε-moves.
9. Define a Moore machine.
10. List the steps in conversion a CFG to Chomsky normal form.

Understand

1. Estimate the PCNF and PDNF of the formula given by $(7P \rightarrow R) \wedge (Q \rightarrow P)$
2. Discuss this $(7Q), P \rightarrow Q, P \vee R \Rightarrow R$ by indirect method
3. Show that $S \vee R$ is tautologically implied by $(P \vee Q) \wedge (P \rightarrow R) \wedge (Q \rightarrow S)$
   using automatic theorem proving
4. Let R denote a relation on the set of ordered pairs of integers such that
   $< x, y > R < u, v >$ iff xv=yu. Show that R is an equivalence relation
5. Let A,B,C are any three sets Prove that $(A \cup B)XC = (AXC) \cup (BXC)$.
6. Let f(x)=x+2, g(x)=x-2,h(x)=3x, for $x \in R$, where R is the set of real numbers. Find
   $f \circ g, g \circ f, f \circ f, g \circ g$ and $f \circ g \circ h$. And also write the corresponding matrix.
7. Discuss NFA for regular expression $(01+10)^+$
8. Discuss the language generated by $S \rightarrow aB, B \rightarrow AB, aA \rightarrow b, A \rightarrow b, B \rightarrow Aa.$
9. Construct Pushdown automata which accept the language with equal number of a's and b's with empty store.

Apply

1.Show that $7(p \wedge (7q \wedge r) \vee (q \wedge r) \vee (p \wedge r) \Leftrightarrow r$

2. Define NAND and NOR. Prove that the connectives NAND and NOR are commutative.

3. Show that $S \vee R$ is tautologically implied by $(P \vee Q) \wedge (P \rightarrow R) \wedge (Q \rightarrow S)$

4. Show that $(x)(P(x) \vee Q(x)) \Rightarrow (x)P(x) \vee (\exists(x))Q(x)$,Using indirect method.

5**.** Prov**e** that L= $\{0^{i^2}$ /i is an integer,i>1$\}$ is not regular

6.Discuss the Pushdown automata .

Syllabus

Logic

Introduction – Statements and Notations , Negation – Conjuction – Disjunction – Truth table , Conditional – Biconditional – Tautological Statements – Equivalence of Formulas, Duality Law – Tautological Implications-Functionally Complete set of Connectives –Other Connectives, Two State Devices & Statement Logic, Disjunctive – Conjunctive – Principle Disjunctive – Principle Conjunctive

Predicate calculas

Checking the validity using the truth table, Rules of Inference –Consistency of premises and Indirect Method, Automatic Theorem proving, Predicates-Function, Variables and Quantifiers, Valid Formulas – Equivalences, Theory of Inferences for the predicate Calculus.

Set theory

Basic Definitions – Operations on Sets –Identities, Ordered pairs & n-tuples – Cartesian Products, Definition of Relation –Binary Relation – Properties-Matrix – Graph, Equivalence relation – Compatibility Relation-Composition of relation-Poset, Functions –Composition-Inverse.

Automata Theory

Finite State machines and Basic Definitions, Non-Deterministic Finite Automata, Finite Automata with ε - moves, Regular Expressions, Two way Finite Automata, Finite Automata with output

Grammar

The Pumping Lemma for Regular Sets, Closure Properties of Regular Sets, Context – Free Grammar and its Simplification, Derivation Trees, Chomsky-Normal Form and Greibach Normal Form

Text Books

1. T.P.Tremblay and R.Manohar ,"Discrete Mathematical Structures with application to Computer Science",Tata McGraw Hill,1997

2. John.E.Hopcorff and Jeffery D. Ullman ,"Introduction to Automata Theory, Languages and Computation"Narosa Publishing House,1997.

References

1. Dr. M.K.Venkataraman.,Dr.N.Sridharan and N.Chandrasekaran, Discrete Mathematics,National Publishing Company,Chennai.of India(2004)

Course contents and lecture schedule

Sl.No	Topics	No.of Lectures
	Unit I	
1	Introduction – Statements and Notations	1
2	Negation – Conjuction – Disjunction – Truth table	1
3	Conditional – Biconditional – Tautological Statements – Equivalence of Formulas	2
4	Tutorial – I	1
5	Duality Law – Tautological Implications- Functionally Complete set of Connectives –Other Connectives	1
6	Two State Devices & Statement Logic	1
7	Disjunctive – Conjunctive – Principle Disjunctive – Principle Conjunctive	2
8	Tutorial – II	1
	UNIT II	
11	Checking the validity using the truth table, Rules of Inference –Consistency of premises and Indirect Method	2
12	Automatic Theorem proving	1
13	Predicates-Function, Variables and Quantifiers	1
14	Tutorial – I	1
15	Predicate formulas – Free and Bound Variables	1
16	Valid Formulas – Equivalences	1
17	Theory of Inferences for the predicate Calculus	2
18	Tutorial – II	1
	UNIT III	
19	Basic Definitions – Operations on Sets –Identities	1
20	Ordered pairs & n-tuples –Cartesian Products	1
21	Definition of Relation –Binary Relation – Properties- Matrix - Graph	2
22	Tutorial - I	1
23	Equivalence relation – Compatibility Relation- Composition of relation- Poset	2

24	Functions –Composition-Inverse	2
25	Tutorial - II	1
	UNIT IV	
26	Finite State machines and Basic Definitions	2
27	Non-Deterministic Finite Automata	1
28	Finite Automata with ε - moves	1
29	Tutorial – I	1
30	Regular Expressions	2
31	Two way Finite Automata	1
32	Finite Automata with output	1
33	Tutorial – II	1
	UNIT V	
34	The Pumping Lemma for Regular Sets	1
40	Closure Properties of Regular Sets	1
41	Context – Free Grammar and its Simplification	1
42	Tutorial – I	1
43	Derivation Trees	1
44	Chomsky-Normal Form and Greibach Normal Form	2
45	Tutorial – II	1
46	Demonstration using Mathematica Player	2

Course Designers

1.V.Mohan vmohan@tce.edu

2.G.Jothilakshmi gjlmat@tce.edu

3.A.P.Pushpalatha appmat@tce.edu

4.K.Angaleeswari kangaleeswari@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T52	3	1	--	4

T52 Software Engineering Methodologies 3:1

Preamble: This course presents methods, tools and procedures that enable to control the process of software development and provide the student with a foundation for building quality software in a productive manner. The course highlights the application of systematic, disciplined, quantifiable approach to the development, operation and maintenance of software.

Program Outcomes addressed

Graduates will demonstrate

c. An ability to design a system, component or process as per needs and specifications.

d. An ability to identify, formulate and solve engineering problems.

e. An ability to use techniques, skills, and modern engineering tools to implement and organize engineering works under given constraints.

Competencies

The Students will be able to

1. Perceive the idea on different process models and select suitable model for their project requirements
2. Annotate the software engineering practices and methodologies
3. Analyze requirements and design techniques to provide solutions
4. Apply design concepts to the real world problems
5. Use testing tactics and tools on software applications
6. Manage and provide solutions for software projects
7. Reuse and Reengineer the existing solutions to for real life problems.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3	End-semester examination
1	Remember	20	10	10	0
2	Understand	30	20	20	30
3	Apply	40	40	30	30
4	Analyze	10	30	20	20
5	Evaluate	0	0	0	0
6	Create	0	0	20	20

Course Level Learning Objectives:

Remember

1. List the components of S/W engineering.
2. Write down the various metrics for S/W Quality.
3. Define Formal Technical Review.
4. Recall the important elements of CSR in SCM.
5. State earned value analysis.
6. Mention the different approaches to the sizing problem?
7. How are test cases generated during loop testing?
8. Exhaustive testing is not possible. Justify.
9. List out the software risks.

Understand

1. Write down the differences between Incremental process model and evolutionary process model.
2. Explain the application where agile process model is implemented.
3. Explain the need for measurement. List down the classifications of measurement.
4. Explain how statistical S/W Quality Assurance is done in organization. Give suitable examples.
5. Discuss how testing is performed in various levels of the software. Give examples.
6. Explain how Risk estimation is done during project development.
7. Explain the various risk strategies.
8. Discuss about Software reviews.
9. Explain Empirical estimation model with suitable example.
10. Explain project scheduling and tracking.
11. There is a subtle difference between restructuring and forward engineering. What is it?

Apply

1. Using the Cocomo models, estimate the effort required to build a S/W for a Simple ATM that produces 24 screens, 15 reports and will require approximately 110 S/W components. Assume average complexity and average developer / environment maturity. Use the application composition model with object points.
2. Select which software life cycle model is applicable for weather monitoring system.
3. Prepare an SRS document for Online shopping S/W.

4. Implement the Empirical estimation model for any real time application.
5. Assume that you are the project manager for a company that builds software for consumer Products. You have been contracted to build the software for an ATM system. Estimate each function in FP. Assuming that your organization produces 67 FP/pm with a burdened labour rate of $8000 /person-month, estimate the Effort and cost required to build the software using the FP–based estimation technique. Have your own assumption for weighted complexity.
6. Compute the 3D function point value for a project with the following information domain characteristics {45, 55, 20, 25, 6, 27, 5}. The code designed for the project is reusable, all master files are updated on-line and the performance is critical. The other complexity adjustment values are considered to be average.
7. Assume a simple ATM that produces 12 screens, 10 reports, and will require approximately 80 software components. Use the COCOMO II model to estimate the effort required to build the software. Assume average complexity and average developer/environment maturity. Use the application composition model with object points.
8. Draw the CFG and Calculate the cyclomatic Complexity for the following program.

```
int compute_gcd(x, y)
int x, y;
{
while (x! = y){
if (x>y) then
x= x – y;
else y= y – x;
}
return x;
}
```

Analyze

1. Differentiate the various processing models highlighting their advantages and disadvantages.
2. Illustrate the cost impact of early defect detection during the software process, assuming the following error detection percentage. Preliminary design-65%, detailed desing-45%, code/unit test-70% and others-50%. Assume your own amplification factors and error generated in each phase.
3. Analyze the tasks that the software Engineer needs to perform to ensure that he has specified a system that properly meets the customer needs and satisfy the customer's expectations.

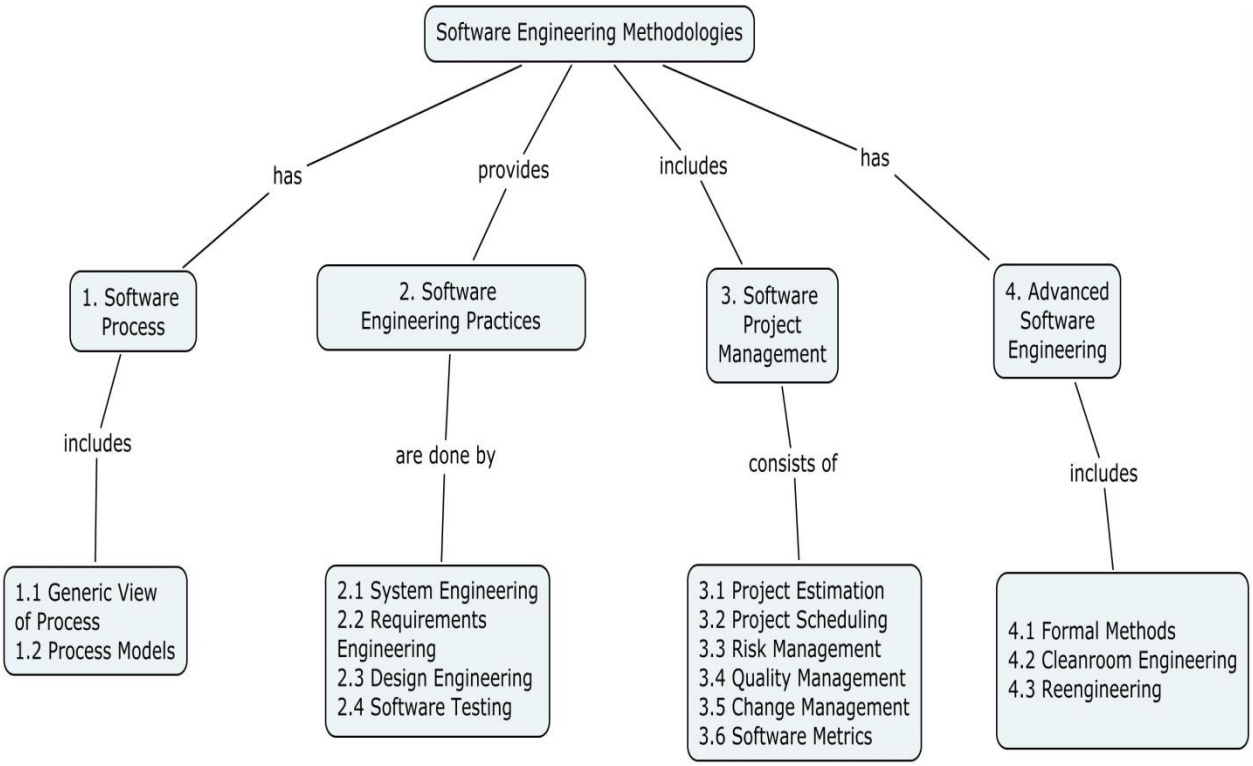
4. Some software metrics are kept "private" and some are kept "public". Analyze the reasons for the same with suitable examples.
5. Validation test is performed by_____ (software developer or the software user). Justify your answer.
6. Compare and contrast various black box testing techniques.

Create

Prepare an XP user story that describes the "favourite places" or"favourites" features available on most web browsers also explain the Extreme programming process with a flow diagram.

1. Develop a design model to represent the interrelationships among Software Configuration Items using UML aggregation and composites
2. Prepare an earned value to assess progress for a software project.
3. Assume you have been contracted to build the software for a Banking application system for a company that builds software for consumer products. Write a statement of scope that describes the software. Develop an estimation model using the functional decomposition.
4. Assume that you are the project manager for a company that builds software for consumer products. You have been contracted to build the software for a home security system. Write a statement of scope that describes the software.
   (i) Do a functional decomposition of the home security system software you described. Estimate the size of each function in LOC. Assuming that your organization produces 450 LOC/pm with a burdened labor rate of $7000 per person-month, estimate the effort and cost required to build the software using the LOC-based estimation technique. Using the 3D function point measure compute the number of FP for the home security system software and derive effort and cost estimates using the FP-based estimation technique.
   (ii) Use the software equation to estimate the home security system software.
   (iii) Develop a single estimate for the project using a three-point estimate.
5. Consider a Vending machine that takes quarters and when it has received two quarters, gives a can of soda. Develop a state model of the system and generate sets of test cases for the various criteria.

Concept Map



Syllabus:

Software Process - Generic view of Process - Layered Technology, Process Framework, Personal and Team Process Models (PSP, TSP) – Process Models – Waterfall Model, Incremental Model, Evolutionary Model, Specialized Process Models, Agile Model, Case Study **Software Engineering Practices - System Engineering** – System Engineering Hierarchy, Business Process Engineering **Requirements Engineering** – Requirement Engineering Tasks – Requirements Analysis – Analysis Modelling Approaches, Data Modelling **Design Engineering** – Design Overview, Design Process and Quality, Design Concepts, Design Model, Pattern based Software Design, Architecture Design, Component modelling. **Software Testing** - Testing Strategies – White Box Testing – Control Flow Graph, Cyclomatic Complexity, Program Dependence Graph– Black Box Testing – Equivalence Partitioning, Boundary Value Analysis, Cause Effect Graph, Predicate Testing **Software Project Management** – Project Estimation, Project Scheduling, Risk Management, Quality Management, Change Management, Software Metrics **Advanced Software Engineering** Formal Methods, Cleanroom Software Engineering, Reengineering – Business Process Reengineering, Software Reengineering, Reverse Engineering, Restructuring.

Textbook:

1. Roger S. Pressman, Software Engineering A Practitioner's Approach, Sixth Edition, McGraw Hill International Edition, 2007

2. Pankaj Jalote, Software Engineering A Precise Approach, 1st edition, Wiley India, 2011.

3. Aditya P. Mathur, Foundations of Software Testing, Pearson Education, 2008.

Reference Books:

1. Ian Somerville-Software Engineering, Pearson Education India, New Delhi, 2006.

2. S. L. Pfleeger, Software Engineering, MacMillan Publishing Company, 1987.

3.Carlo Ghezzi, Mehdi Jazayari and Dino Mandrioli, "Fundamentals of Software Engineering", Prentice Hall of India, New Delhi, 1991.

4. Stephen R.Schach, Classical and Object Oriented Software Engineering, Irwin McGraw Hill - 1996.

5. Philip B. Crosby, Quality is free: the art of making quality certain, McGraw-Hill, New York, October 1979.

Some useful links:

1. http://elearning.tvm.tcs.co.in
2. http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/

Course Contents and Lectures schedule

S. No	Topic	No of Lectures
1	**Software Process**	
1.1	Generic View of Process	
1.1.1	Layered Technology	1
1.1.2	Process Framework	1
1.1.3	Personal and Team Process Models	1
1.2	Process Models	
1.2.1	Waterfall Process Model	0.5
1.2.2	Incremental Process Model	0.5
1.2.3	Evolutionary Process Model	1
1.2.4	Specialized Process Models	1
1.2.5	Agile Process Model	1
1.2.6	Case Study	1
2	**Software Engineering Practices**	
2.1	**System Engineering**	
2.1.1	System Engineering Hierarchy	0.5

2.1.2	Business Process Engineering	0.5
2.2	**Requirements Engineering**	
2.2.1	Requirements Engineering Tasks	1
2.2.2	Requirements Analysis	1
2.2.3	Analysis Modeling Approaches	1
2.2.4	Data Modelling	1
2.3	**Design Engineering**	
2.3.1	Design Overview	0.5
2.3.2	Design Process and Quality	1
2.3.3	Design Concepts	1
2.3.4	Design Model	1
2.3.5	Pattern based Software Design	0.5
2.3.6	Architecture Design	4
2.3.7	Component Modelling	2
2.4	**Software Testing**	
2.4.1	Testing Strategies	3
2.4.2	White Box Testing	
2.4.2.1	Control Flow Graph	1
2.4.2.2	Program Dependence Graph	1
2.4.3	Black Box Testing	
2.4.3.1	Equivalence Partitioning	1
2.4.3.2	Boundary Value Analysis	1
2.4.3.3	Cause Effect Graph	1
2.4.3.4	Predicate Testing	1
3	**Software Project Management**	
3.1	Project Estimation	2
3.2	Project Scheduling	1
3.3	Risk Management	1
3.4	Quality Management	2
3.5	Change Management	2
3.6	Software Metrics	1
4	**Advanced Software Engineering**	
4.1	Formal Methods	1
4.2	Cleanroom Software Engineering	1
4.3	Reengineering	
4.3.1	Business Process Reengineering	0.5
4.3.2	Software Reengineering	0.5

4.3.3	Reverse Engineering	0.5
4.3.4	Restructuring	0.5
	Total Lectures	**45**

Course Designers:

1. S.Selvakumar ssit@tce.edu
2. A.M.Abirami abiramiam@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T 53	3	0	--	3

T53 Information Storage and Management 3:0

Preamble: The course on Information Storage and Management aims to emphasize the need for information storage, provide an in depth coverage of various technologies in the various phases of designing and building an Information Storage System and to provide an overview of various management techniques.

Programme Outcomes addressed

a. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

b. Graduates will demonstrate an ability to design a system, component or process as per needs and specifications.

c. Graduate will demonstrate skills to use modern engineering tools, software and equipment to analyze problems.

Competencies

1. Explain the components and functions of information storage systems.
2. Evaluate the existing information storage systems.
3. Create storage system requirement specification document based on interaction with storage user.
4. Investigate the common threats in Storage security domains.
5. Design and manage the storage infrastructure effectively.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3/End-semester examination
1	Remember	30	20	10
2	Understand	40	40	30
3	Apply	30	20	30
4	Analyze	0	0	0
5	Evaluate	0	0	10
6	Create	0	20	20

Course Level Learning Objectives

Remember

1. What are the different categories of data?
2. What do you mean by downtime?

3. List the demerits of centralized data storage.
4. Describe the key requirements of storage systems with their functionalities.
5. What are the characteristics of Platter?
6. What is TAG RAM?
7. What are the challenges of NAS?
8. Explain DAS performance considerations and challenges.
9. What do you mean by fixed content?
10. What are the access control components?
11. What are the storage regulations?
12. Define: Strip and Stripe.
13. Define Single-mode and Multi-mode fiber.
14. What is the need for Storage Security?
15. What is BURA?
16. What are the benefits of NAS?

Understand:

1. Describe how you can control Application access, User access and Host access.
2. Explain how CAS stores and retrieves data objects
3. Write down the benefits of CAS.
4. Describe the disk drive components and its operation in detail
5. Describe the evolution of date storage systems.
6. Explain the structure of Cache
7. Describe the functionalities of FC Switch and Hub with necessary diagrams.
8. Describe how you can control Management access and protect Administrative access
9. Write down the benefits and forms of Virtualization
10. Describe the disk drive components and its operation in detail
11. Compare and contrast NFS and CIFS.
12. Differentiate between Internal and External DAS management
13. Compare and contrast between Integrated and Gateway NAS
14. Explain the IP SAN protocol
15. Compare the different storage media with their merits and demerits
16. Explain how remote replication technology is helpful in disaster recovery.
17. Distinguish Strip from Stripe.
18. Compare Single-mode and Multi-mode fiber.

Apply

1. An application specifies a requirement of 200GB to host a database and other files. It also specifies that the storage environment should support 5000 IOPS

during its peak processing cycle. The disks available for configuration provide 66GB of usable capacity and the manufacturer specifies that they can support a maximum of 140 IOPS. The application is response time sensitive and the disk utilization beyond 60% will not meet the response time requirements of the application. Compute the minimum number of disks that should be configured to meet the requirements of the application.

2. Consider a disk I/O system in which an I/O request arrives at the rate of 80 IOPS. The disk service time is 6ms.
    a. Compute the following
        i. Utilization of I/O controller
        ii. Total response time
        iii. Average queue size
        iv. Total time spent by a request in a queue
    b. Compute the preceding parameter if the service time is halved.
3. A 10k RPM drive is rated to perform 130 IOPS and a 15k RPM drive is rated to perform 180 IOPS for an application. The read/write ratio is 3:1. Compute the RAID-adjusted IOPS for the 10k and 15k drives for RAID1, RAID5 and RAID6.
4. An application has 1000 heavy users at a peak of 2 IOPS each and 2000 typical users at a peak of 1 IOPS each, with a read/write ratio of 2:1. It is estimated that the application also experiences an overhead of 20% for other workloads. Calculate the IOPS requirement for RAID 1, RAID 3, RAID 5 and RAID 6.
5. A UNIX host has a path to a storage device that shows as c0 t1 d3. Draw a diagram to show the path.
6. Seventeen switches with 16 ports each are connected in a mesh topology. Determine the ports available for host and storage connectivity for a high-availability solution.
7. SAN is configured for a backup to disk environment and the storage configuration has additional capacity available. Determine whether a NAS gateway can be configured to utilize this SAN.
8. ABC Corporation is trying to decide between an integrated or a gateway NAS solution. The existing SAN at ABC will provide capacity and scalability. The IT department is considering a NAS solution for the training department at ABC for training videos. The videos will only be used by the training department for evaluation of instructors. Determine a NAS solution.
9. A company is considering storage implementation. They do not have a current storage infrastructure to use, but they have a network that gives them good performance. Determine whether native or bridged iSCSI should be used.

10. The IP bandwidth provided for FCIP connectivity seems to be constrained. Estimate the implications if the SANs that are merged are fairly large with 500 ports on each side and the SANs at both the ends are constantly reconfigured.
11. The IT department of a bank promises customer access to the currency conversion rate table between 9am and 4pm from Monday to Friday. It updates the table everyday at 8am with a feed from the Mainframe system. The update process takes 35 minutes to complete. On Thursday, due to a database corruption, the rate table could not be updated. At 9.05 am, it was established that the table had errors. A rerun of the update was done and the table was recreated at 9.45 am. Verification was run for 15 minutes and the rate table became available to the bank branches. Compute the availability of the rate table for the week in which this incident took place assuming that there were no other issues.

Create:

1. AirTel Telecom is involved in mobile wireless services across the India and has about 5000 employees worldwide. This company is Chennai based and has 7 regional offices across the country. Although AirTel is doing well financially, they continue to feel competitive pressure. As a result, the company needs to ensure that the IT infrastructure takes advantage of fault tolerant features.

   Current Situation/Issues:

   i. The company uses a number of different applications for communication, accounting, and management. All the applications are hosted on individual servers with disks configured as RAID 0.
   ii. All financial activity is managed and tracked by a single accounting application. It is very important for the accounting data to be highly available.
   iii. The application performs around 15% write operations, and the remaining 85 % are reads.
   iv. The accounting data is currently stored on a 5-disk RAID 0 set. Each disk has an advertised formatted capacity of 200 GB, and the total size of their files is 730 GB.
   v. The company performs nightly backups and removes old information—so the amount of data is unlikely to change much over the next 6 months. The company is approaching the end of the financial year and the IT budget is depleted. Buying even one new disk drive will not be possible. Design an infrastructure for the company to suit the new requirements. Justify your design based on cost, performance, and availability.

2. A mid-sized publishing house has a centralized IT department located in Mumbai, connected to 3 branch offices in the South. Recent analysis of the storage environment suggests that they are getting poor ROI in their storage infrastructure.

Current Situation/Issue:

The company's current infrastructure consists of several storage arrays direct-attached to a heterogeneous mix of 90 servers. All servers are dual-attached to the arrays for reliability and redundancy. Since each storage device has 32 connectivity ports, each could support a maximum of 16 servers. The company sees the 32-ports as a limitation. Each storage device has the disk capacity to support more than 16 servers. However, there was no way to add a 17^{th} server to make use of that capacity and support future growth. There are limited backup devices in a file sharing NAS environment.

Design a suitable backup strategy that will minimize the network traffic, avoid any congestion and at the same time, does not impact the production operations. Justify your design.

3. A manufacturing corporation uses tape as their primary backup storage media throughout the entire organization.

Current Situation/Issue:

Full backups are run every Sunday. Incremental backups are run from Monday through Saturday. There are many backup servers in the environment, backing up different groups of servers. Their e-mail and database applications have to be shut down during the backup process. The main concerns facing the corporation are:

i. Due to the de-centralized backup environment, recoverability of the backup servers is compromised.

ii. Key applications have to be shut down during the backup process.

iii. Too many tapes need to be mounted in order to perform a full recover, in case of a complete failure. The company would like to:

   a. Deploy an easy-to-manage backup environment.

   b. Reduce the amount of time the email and database applications need to be shutdown.

   c. Reduce the number of tapes required to fully recover a server in case of failure.

Create a network based on IP SAN topology.

4. A Manufacturing Corporation maintains the storage of their mission critical applications on high-end Storage Arrays on RAID 1 volumes. The corporation has two data centers which are 50 miles apart.

   Current Situation/Issue:

   The corporation's mission critical Database application takes up 1 TB of storage on a high end Storage Array. In the past year, top management has become extremely concerned because they do not have DR plans which will allow for zero RPO recovery if there is a site failure. The primary DR Site is the 2nd Data Center 50 miles away. The company would like explore remote replication scenarios which will allow for near zero RPO and a minimal RTO. The company is aware of the large costs associated with network bandwidth.

   Develop a better solution based on other remote replication technologies and the zero RPO solution.

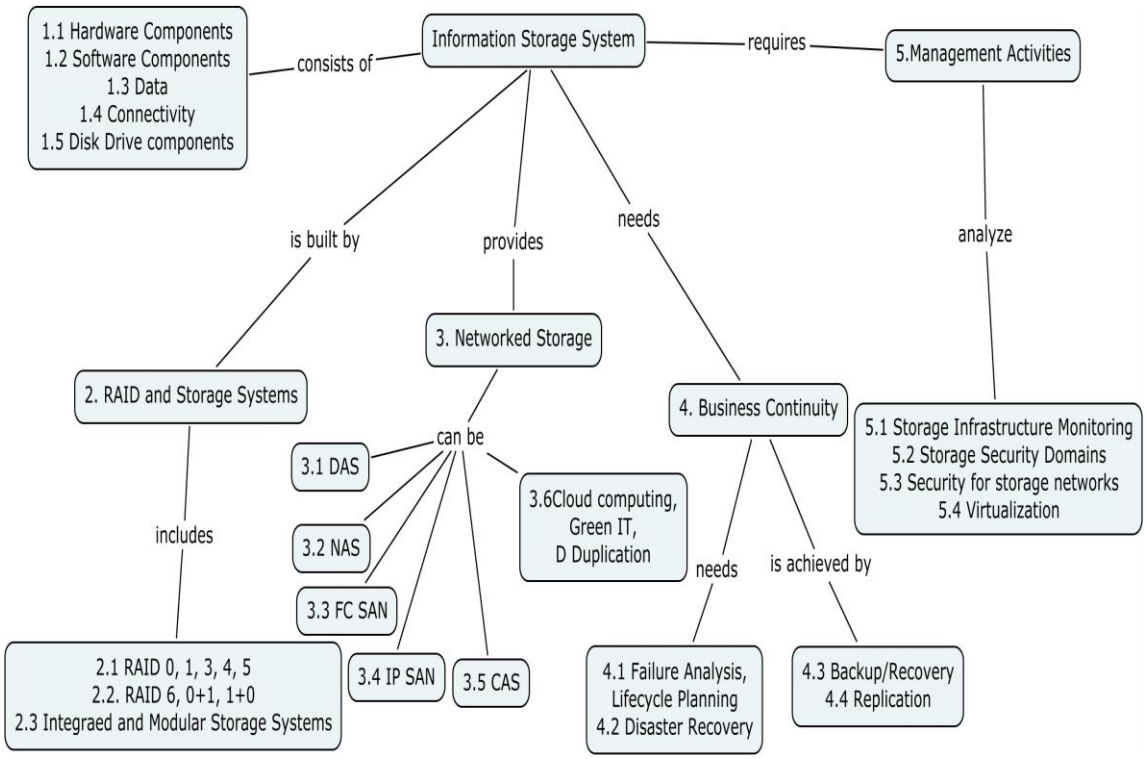
5. A performance problem has been reported on a database. Monitoring confirms that at 12am, a problem surfaced and access to the database is severely affected until 3pm everyday. This timeslot is critical for business operations and an investigation has been launched. A reporting process that starts at 12pm contends for database resources and constrains the environment. Create monitoring and management procedures, tools and alerts to ensure accessibility, capacity, performance and security.

6. The IT Department of a departmental store uses tape to archive data. The data once created may be accessed within 30 days and when it crosses that period, the frequency of access is less than 1%. Design a CAS solution.

7. An oracle database uses a block size of 4kb for its I/O operation. The application that uses this database primarily performs a sequential read operation. Design a Cache memory with the following parameters: Cache page size, cache allocation, pre-fetch type and write aside cache.

8. The IT Department of TCE has 200 workstations in 2 laboratories. All the machines are connected in IT workgroup, connected to a server with 1 TB capacity. The average utilization of the workstations is 300 hours/week and usage is restricted to 9am to 5pm segment. Develop a backup/recovery topology for the department.

Concept Map



Syllabus

Introduction

Hardware and software components of the host environment - Key protocols and concepts used by each component - Data creation - Physical and logical components of connectivity environment - Physical Components of a disk drive - Logical constructs of a physical disk.

RAID and Intelligent Storage Systems

RAID levels: RAID 0, RAID 1, RAID 3, RAID 4, RAID 5 - RAID 6 - RAID 0+1, RAID 1+0 - Integrated and Modular Storage Systems - Intelligent Storage System- Architecture and working.

Networked Storage

Direct Attached Storage - Architecture and components - Network Attached Storage – Architecture and components and connectivity - Fibre Channel Storage Area Network – Evolution, Architecture, components and connectivity - Internet Protocol Storage Area Network – Architecture, components and topologies - Need for long-term archiving solution: CAS Introduction- Cloud Computing, Green IT,D Duplication

Business Continuity

Business Continuity (BC) – Information Availability, BC Planning Lifecycle - Failure Analysis: Single points of failure, Fault Tolerance – Disaster Recovery (DR),

Recovery Time Objective (RTO) and Recovery Point Objective (RPO) – Architecture of backup/recovery - Backup topologies: Direct Attached, LAN-based, SAN-based, Mixed – Replication - Remote replication: Host-based, Storage Array-based

Storage Management Activities

Storage Infrastructure Monitoring: Parameters, Components, Alerts - Storage Security Domains: Threats, Vulnerabilities - Security Implementations in Storage Networks - Virtualization technologies: Block-level and file-level virtualization technologies and processes

Text Book

1. Information Storage and Management, EMC Corporation, Wiley 2009, ISBN: 04702942134

Reference Books

1. "Storage Networks: The Complete Reference" - Robert Spalding - Tata McGraw Hill- Osborne- 2003.
2. "Building Storage Networks" - Marc Farley - Tata McGraw Hill-Osborne- 2001.
3. "Disaster Recovery and Business Continuity" - Thejendra BS- Shroff Publishers and Distributors-2006.
4. Storage Area Network Fundamentals- Meeta Gupta- Pearson Education Limited, 2002.

Course contents and Lecture Schedule:

No.	Topic	No. of Lectures
1	**Introduction**	
1.1	Hardware components of the host environment	1
1.2	Software components	2
1.3	Data Creation	2
1.4	Physical and logical components of Connectivity environment	2
1.5	Physical Components of a disk drive - Logical constructs of a physical disk	2
2	**RAID and Intelligent Storage Systems**	
2.1	RAID levels- RAID 0, RAID 1, RAID 3, RAID 4, RAID 5	2
2.2	RAID 0+1, RAID 1+0, RAID 6	2
2.3	Integrated and Modular Storage Systems - Intelligent Storage System- Architecture and working	2

No.	Topic	No. of Lectures
3	**Networked Storage**	
3.1	Direct Attached Storage – Architecture and components	2
3.2	Network Attached Storage – Architecture and components and connectivity	2
3.3	Fibre Channel Storage Area Network – Evolution, Architecture, components and connectivity	2
3.4	Internet Protocol Storage Area Network –Architecture, components and topologies	2
3.5	Need for long-term archiving solutions - CAS	1
3.6	Introduction- Cloud Computing, Green IT,D Duplication	1
4	**Business Continuity**	
4.1	Business Continuity (BC) – Information Availability, BC Planning Lifecycle- Failure Analysis: Single points of failure, Fault Tolerance	2
4.2	Disaster Recovery (DR), Recovery Objective (RPO) Time Objective (RTO) and Recovery Point	2
4.3	Architecture of backup/recovery – Backup topologies: Direct Attached, LAN-based, SAN-based, Mixed	2
4.4	Replication – Remote replication: Host-based, Storage Array-based	2
5	**Management Activities**	
5.1	Storage Infrastructure Monitoring: Parameters, Components, Alerts	2
5.2	Storage Security Domains: Threats, Vulnerabilities	2
5.3	Security Implementations in Storage Networks	2
5.4	Virtualization technologies - Block-level and file-level virtualization technologies and processes	2

COURSE DESIGNERS:

1. G.S.R. Emil Selvan emil@tce.edu
2. R. Muthu Selvi rmsit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T54	4	0	--	4

T54 INFORMATION SECURITY AND ASSURANCE 4:0

Preamble: Information Security and Assurance is the discipline which studies the theoretical, practical and managerial aspects of securing Information against threats. It utilises techniques from discrete applied mathematics, including algebra, number theory and probability, as well as from computer science, communications engineering and business. The course will enable the students to understand, develop, and deploy countermeasures to mitigate the risks inherent in the transmission, storage and retrieval of sensitive information.

Program Outcomes addressed

a. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

b. Graduates will demonstrate an ability to design a system, component or process as per needs and specifications.

c. Graduate will demonstrate skills to use modern engineering tools, softwares and equipment to analyze problems.

Competencies

1. Analyze threats and vulnerabilities of information systems including databases, networks, applications, internet-based communication, web services, and mobile technologies.
2. Deploy measures that protect and defend information and information systems by ensuring their availability, integrity, authentication, confidentiality, and non-repudiation.
3. Explain Information Assurance that includes corporate governance issues such as privacy, compliance, audits, business continuity, and disaster recovery.
4. Determine the strength of a given algorithm used for security service.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3/End-semester examination
1	Remember	20	10	10
2	Understand	20	20	20
3	Apply	40	50	50
4	Analyze	20	20	20
5	Evaluate	0	0	0
6	Create	0	0	0

Course Level Learning Objectives

Remember:

1. Define Confidentiality, Integrity and Non repudiation.
2. What is Crypt analysis?
3. Explain Brute force attack with an example.
4. Develop the model for network security.
5. Depict by a table the relationship between security and mechanisms.
6. Define Euler's totient function.
7. Enlist some of the methods for primality testing.
8. Define Discrete logarithm problem.
9. What is a finite field?
10. What is a Challenge Response scheme?
11. What is availability?
12. What is Markov model?

Understand:

1. Distinguish between diffusion and confusion.
2. Enlist the differences between active and passive attacks.
3. Use sieve of Eratosthenes to generate prime numbers less than 200.
4. Differentiate MAC and Hash functions.
5. How does public key cryptosystem provide authentication?
6. Enumerate the differences between strong and week collision resistance of hash functions.
7. Which three independent dimensions characterize the cryptographic system?
8. What are the various types of crypt analytic attacks on encrypted systems?
9. What do 'unconditionally secure' and 'computationally secure' imply?
10. Tabulate the average time required for various exhaustive key searches.
11. Explain disaster Recovery Management in detail.
12. How is availability of Information service achieved?
13. Draw the E/D scheme of Blowfish by schematic diagrams for 16 rounds.
14. What is the F function? Illustrate by means of a layout diagram the computations involving the 4 S-Box values in blow fish algorithm.
15. Compute the P_w and Q_w values of RC5 algorithm for w=16, 32, 64.

Apply

1. Apply Hill cipher to the message "TCE" with key

   17 7 5 ; 21 18 21 ; 12 2 19
2. Apply Vignere cipher to message DONKEY with key "DIVINE".
3. Apply Miller Rabin primality test to 561, to verify if it is a prime.
4. Apply trial division algorithm to factorize 1523357784.

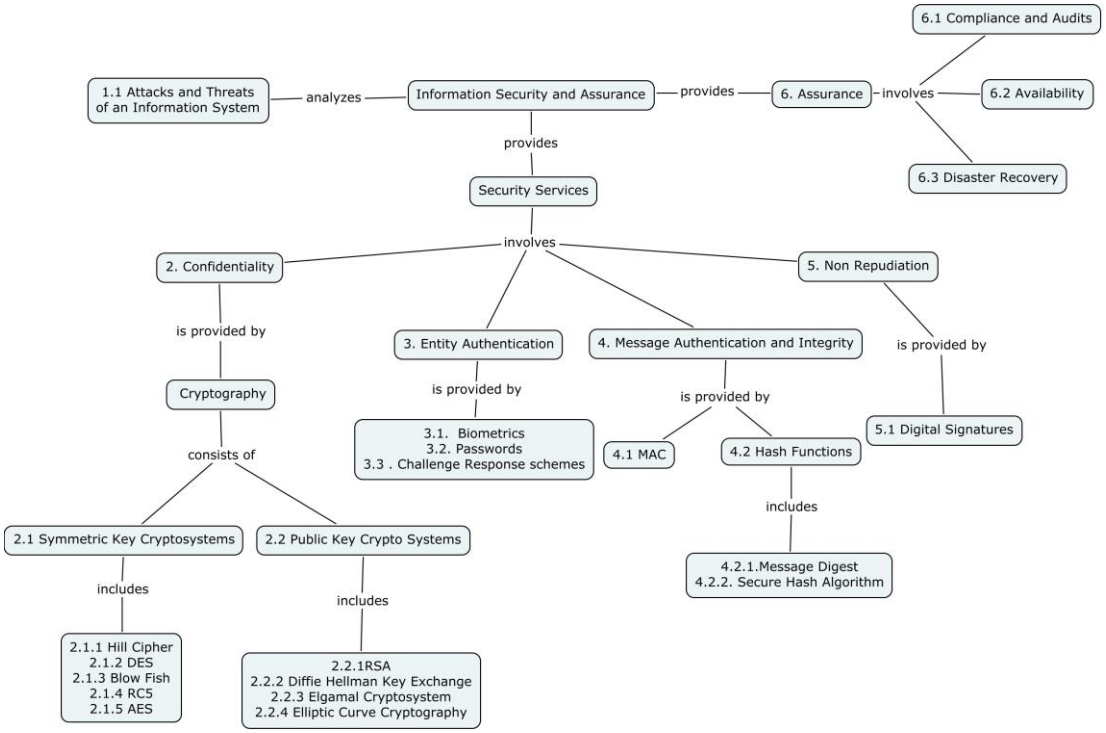
5. Apply Fast Exponentiation algorithm to evaluate 7560 mod 561.
6. Perform encryption and decryption using RSA algorithm for the following:
   a. n=33, M=5 (b) n=55, M=9 (c) n=77, M=8 (d) n=143, M=7 (e) n=527,M=2
7. Determine the primitive roots of prime 19.
8. Given a message TO and a key K, apply the Feistel cipher for 3 rounds. Use ASCII Hex representation for the alphabets.
9. Given a 64 bit key THEQUICK determine the first and second round sub keys for use in DES.
10. Given a 64 bit message CLIMATEC, perform the initial permutation IP and then apply the IP-1, to recover the original message.
11. Given a right side 32 bit message SOME, apply the E/P technique to convert into a 48 bit message. Given a 48-bit key WXYZRE, calculate the F(R,K) function.
12. Apply the key expansion pseudo code and generate the first 4 words of the key using the key byte values in AES.
13. Apply the key expansion pseudo code and generate w[4], w[5],w[6] and w[7].
14. Given that the round key for the 4th round is e2f467893153f560292f8d7fec2d3712, determine the first 4 bytes of round 5 in AES.
15. Given a message DONKEYSVERSATILE, apply one round of AES and determine the cipher text using the initial key K=MADURAIDESTINATI.
16. The message "DONE" is concatenated with its Frequency Check Sum. This combination is encrypted with the symmetric key K="RUSTE" and transmitted to host B. At the receiving end host B decrypts it using the symmetric key and segregates M||F(M). He then re-computes F(M) and compares it with the received F(M), thereby authenticating the message. Illustrate the above steps. Use ASCII Hex for alphabets.
17. Given E67(2,3), e1=(2,22), d=4, determine e2=d.e1, by first multiplying and then adding twice. Applying ElGamal principle evaluate C1=r.e1 and C2=P+r.e2, Where P is the message (24,26) At the receiving end using C1 and C2, recover the plain text.
18. For E13(1,1), What is 4P, given P=(1,4)?

Analyze:

1. Analyze the key management issues involved in symmetric key cryptosystems with respect to public key Cryptosystems.
2. Can the following matrix be used as key in Hill cipher? Justify your answer. {1,2,3; 4,5,6; 7,8,9}

3. You intercept the message 'FBRTLWUGATEPHBNXSW' which was encoded using a Hill Cipher System with a 3 X 3 key matrix in a 26 letter alphabetic system. The last nine letters are the sender's signature 'JAMESBOND'. Find the enciphering matrix, deciphering matrix and read the message.
4. If an adversary is using 2 X 2 enciphering matrix with a 29 letter alphabet, where AZ have the usual numerical equivalents, underscore=26, ?=27, !=28,and the cipher text received is "GFPYJP_X?UYXSTLADPLW" , Crypt analyze the message. (Hint: Last five letters of plain text corresponds to the adversary signature "KARLA").
5. Comment on the strength of mono alphabetic cipher if the language used is Tamil with respect to the number of mappings (Keys).
6. Analyze the threats and vulnerabilities involved in an online examination system.

Concept Map



Syllabus:

Attacks and Threats: Active and passive attacks, Threats and Vulnerabilities, Services and Mechanisms, Model for network security. **Security services: Confidentiality:** Cryptography **Mathematics behind Cryptography**: Number Theory, Modular Arithmetic, Greatest Common Divisor, Multiplicative Inverses, Groups, Rings, Fields, Finite Fields of the form GF(p), Finite fields of the form GF(2^n), Primes, Cardinality of Primes, Euler's totient function , Fermat's and

Euler's Theorem, Primality Testing and Factorization, primitive roots, Discrete logarithm. **Symmetric Key Crypto systems**: Hill Cipher, Data Encryption standard,Blowfish, RC5 Advanced Encryption Standard. **Public key cryptosystems:** RSA, Diffie Hellman Key Exchange, ElGamal Cryptosystem, Elliptic Curve Cryptography **Entity Authentication:** Biometrics, Passwords, Challenge Response protocols, **Message Authentication and Integrity:** Message Authentication code, Hash functions – Message Digest, Secure Hash Algorithm **NonRepudiation:** Digital Signatures **Assurance:** Availability, Disaster Recovery, Compliance and audits.

Textbook

1. Behrouz A.Foruzan, Cryptography and Network Security, Tata McGraw Hill 2007.

Reference Books

1. William Stallings, Cryptography and Network Security: Principles and Practice", PHI 3rd Edition, 2006.
2. Nina Gobbole: Information Systems Security: Security management, Metrics, Framework and best practices, 1st Edition, Wiley, 2009.
3. Mark Stamp: Information Security Principles and Practice, Wiley-Inter science, 2001.
4. Steve Burnett, Stephen Paine RSA Security's official guide to cryptography, Tata Mcgraw Hill Publishing Co Ltd,2001

Course Contents and Lectures schedule

S.No	**Topic**	No of Lectures
1	**Introduction To Information Security**	1
1.1	Attacks and Threats	1
	Security Services	1
2	**Confidentiality – Cryptography**	1
2.1	**Symmetric Key Cryptosystems**	1
2.1.1	Modular Arithmetic, Greatest common Divisor, Euclid Algorithm, Multiplicative Inverses	1
2.1.1	Matrices and Determinants, Inverses, Residue Matrices	1

2.1.1	Hill Cipher	1
2.1.2	Data Encryption Standard	1
2.1.3	Blow Fish	1
2.1.4	RC5	1
2.1.5	Groups, Rings, Fields	1
2.1.5	Finite fields of the form GF(p), Finite fields of the form GF(2^n),	1
2.1.5	Multiplicative inverse of a polynomial	1
2.1.5	Advanced Encryption Standard	2
2.2	**Public Key Cryptosystems**	
2.2.1	Primes, Cardinality of Primes, Eulers totient function, Fermat's and Euler's Theorem	1
2.2.1	Primality testing and Factorization	1
2.2.1	RSA	1
2.2.2	Primitive roots, Discrete Logarithm, Diffie Hellman Key Exchange	1
2.2.3	Elgamal Cryptosystem	1
2.2.4	Elliptic Curve Arithmetic	2
2.2.4	Elliptic Curve Cryptography	1
3	**Entity Authentication**	
3.1	Biometrics – Finger prints, Iris recognition, Voice recognition, Face recognition, Biometric Fusion.	1
3.2	Passwords	1
3.3	Challenge Response Schemes	1
4	**Message Authentication and Integrity**	
4.1	MAC	1
4.2	Hash Functions	1
4.2.1	Message Digest	2
4.2.2	Secure Hash Algorithms	1

5	**Non Repudiation**	
5.1	Digital Signatures	2
6	**Assurance**	
6.1	Compliance and Audits	2
6.2	Availability	2
6.3	Disaster Recovery	2

Course Designers:

1. C. Jeyamala jeyamala@tce.edu

2. J. Jennifer Ranjani jenni@tce.edu

3. M. Amutha Prabhakar ap_sse@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T55	3	1	-	3

T55 GRAPHICS AND VISUALIZATION 3:1

(Common to CSE/IT: C32)

Preamble: The goal of this subject is to motivate the students to acquire knowledge and skills in graphics and visualization. The topics like 2D, 3D objects and visualization help the students to implement their own ideas with more creativity.

Programme Outcomes addressed

a. Graduates will demonstrate knowledge of mathematics, science and engineering.

b. Graduate will demonstrate skills to use modern engineering tools, softwares and equipment to analyze problems.

Competencies

1. Explain the role of graphics and images in human computer interface.
2. Explain the functioning and characteristics of input and output devices.
3. Explain and compare algorithms for line and circle drawings, clipping and transformations.
4. Explain the differences between 2D and 3D graphics.
5. Select 2D and 3D tools for different applications.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	End-semester examination
1	Remember	30	10	10
2	Understand	20	30	20
3	Apply	20	30	40
4	Analyze	20	20	20
5	Evaluate	10	10	10
6	Create	0	0	0

Course Level Learning Objectives

Remember

1. What is clipping? List out its types.
2. Name the basic two dimensional transformations.
3. Give the matrix representation for shearing.
4. What is composite transformation?
5. What is viewing and windowing?
6. Define projection.

7. Give the properties of B-Spline curve.
8. Define animation.
9. Mention some general purpose languages that support animation.
10. What is scientific visualization?
11. List out the visualization techniques.
12. What are the display methods available for representing 3D objects?
13. List out 2D and 3D tools.
14. What are the steps involved in designing animation sequence.
15. Give a procedure to design an object surface using Bezier curves?
16. Prove that the rotation matrix R is equal to the composite matrix R_y (ß). R_x(α).

Understand

1. Difference between raster and vector graphics.
2. What is the difference between 2D and 3D objects?
3. Illustrate the logic of the Sutherland –Hodgeman algorithm.
4. Explain the reflection and shear transformation in three dimensional graphics applications by providing the matrix representation and example figures.
5. Difference between window and view port.
6. How curves are used in modeling and function graphing?
7. Explain how straight lines can be used to draw Bezier cubic curves.
8. Describe the procedure for drawing Bezier curves.
9. How will you perform morphing? Mention its applications.
10. Explain how to create running tiger from moving automobile using morphing effect.
11. Write a procedure to create bouncing ball using motion specifications.
12. Differentiate parallel projection from perspective projection.

Apply

1. Perform a 45 deg. Rotation of triangle a (0, 0), b (1, 1), c (5, 2) about the origin and about p (1,-1).
2. Write a function to draw a rectangle.
3. Write a function to draw a polygon.
4. Formulate transformation matrices for the following: Rotation about any pivot point (xp, yp) not just origin, Scaling relative to any fixed point (xf, yf) not just origin.
5. Illustrate the mid point circle generation algorithm for drawing a circle centered at (0, 0) and having radius of 5 units.
6. Write a program to demonstrate the Reflection and Shear transformations.
7. Write the algorithm for point clipping.
8. Implement the Cohen-Sutherland Line Clipping Algorithm.

9. Write an efficient routine to display 2D cubic Bezier curve given a set of four control points in the xy plane.
10. Explain how visualization is used in different applications?.
11. Write a program to implement scan-line algorithm for displaying the visible-surfaces of a given polyhedron.
12. Write a program to perform text clipping.
13. Determine the sequence of basic transformations that are equivalent to the x-direction shearing matrix.
14. Determine the Bezier blending functions for five control points. Plot each function and label the maximum and minimum values.
15. Determine the blending functions for uniform, periodic B-Spline curves for d=5.

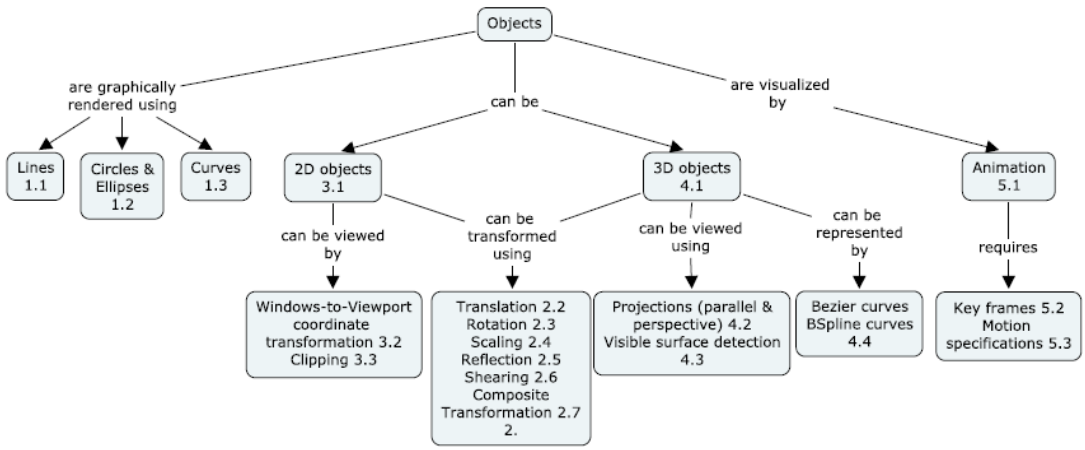
Analyze

1. Analyze the space and time complexity of two line drawing algorithms.
2. Analyze the computational efficiency in composite transformations.
3. Analyze the number of arithmetic operations performed in Cohen Sutherland and the Liang-Barsky line clipping algorithm for several different line orientations relative to clipping window.
4. Compare and contrast three methods of visible surface detection algorithm and hence analyze which algorithm is efficient? .
5. What are the advantages of B-Spline over Bezier Curves?

Evaluate

1. Evaluate that the multiplication of 2D transformation matrices for each of the following sequence of operation is commutative.
    a. Two successive Rotations.
    b. Two successive translations.
    c. Two successive scaling.

Concept Map



Syllabus

Introduction to Computer Graphics : Graphics types and its applications, Points and Lines, Line Drawing Algorithms- DDA, Bresenham and Parallel Line Algorithm, Midpoint Circle drawing Algorithm, Ellipse generating Algorithm, Parallel Curve Algorithm. **2. 2D and 3D Geometric Transformations:** Basic Transformations- Translation, Rotation, Scaling, Reflection, Shearing, Composite Transformations **3. 2D Viewing:** Viewing Pipeline, Window to view-port coordinate Transformation, Clipping Operations- Point Clipping, Cohen Sutherland Line Clipping, Liang Barsky Line Clipping, Sutherland Hodgeman Polygon Clipping, Curve and Text Clipping. **4. 3D Viewing and Object Representation:** Viewing Pipeline, Projections – Parallel and Perspective Projection, View-port Clipping, Bezier Curves and Surfaces, B-Spline Curves and Surfaces, Visible Surface Detection Algorithm- Back-Face Detection Algorithm, Depth Buffer Method, Scan line Method. **5**. **Visualization and Animation:** Visualization and its techniques, Overview of Data Visualization, Applications of Visualization, Principles of animation, File Formats, Keyframes, Motion Specifications.

Text Book

1. Donald Hearn and M. Pauline Baker: Computer Graphics, PHI/Pearson Education, Second Edition, 2004.

References

1. Zhigand Xiang, Roy Plastock: Theory and problems of Computer Graphics, Schaum's outline Series, Tata Mc-Graw hill edition. 2005.
2. Foley, James D Dam, Andries Van: Computer Graphics Principles and Practice, Pearson Education, 2002.
3. Donald Hearn and M. Pauline Baker: Computer Graphics: C Version, Pearson Education, Second Edition, 2006.

Course Contents and Lecture Schedule

No	Topic	No of Lectures
0	**Graphics and Visualization: an Introduction**	**1**
1.	**Rendering of Objects**	
1.1	Rendering of lines ((DDA Algorithm, Bresenham's Line Algorithm, and Parallel Line Algorithm)	3

1.2	Rendering of circles (Mid-point circle algorithm)	2
1.3	Rendering ellipses (Mid-point ellipse algorithm)	1
1.4	Rendering curves (Parallel curve algorithm)	1
2	**2D and 3D Geometric Transformations**	
2.1	Introduction to Basic Transformations	1
2.2	Translation	1
2.3	Rotation	1
2.4	Scaling	1
2.5	Reflection	2
2.6	Shearing	2
2.7	Composite Transformations	2
3.	**2D Viewing**	
3.1	Viewing Pipeline	1
3.2	Window to view-port coordinate Transformation	1
3.3	Clipping	
3.3.1	Point Clipping	1
3.3.2	Line Clipping (Cohen Sutherland Line Clipping, Liang Barsky Line Clipping)	2
3.3.3	Sutherland Hodgeman Polygon Clipping	1
3.3.4	Curve and Text Clipping	1
4	**3D Viewing and Object Representation**	
4.1	Viewing Pipeline	1
4.2	Projections – Parallel and Perspective Projection	2
4.3	Visible surface detection (Back-Face Detection Algorithm, Depth Buffer Method, Scan line Method)	2
4.4	Bezier, B-spline curves and Surfaces	4
5	**Visualization and Animation**	
5.1	Visualization of objects	4

5.2	Key frames	1
5.3	Motion Specifications	1

Course Designers

1. S. Sridevi sridevi@tce.edu
2. R. Suganya rsuganya@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T56	4	0	--	4

T56 Platform Technologies 4:0

Preamble: One of the roles of the IT professional is to design and build systems and integrate them into an organization. The role of the IT professional is to select, deploy, integrate and administer platforms or components to support the organization's IT infrastructure. This course covers the skills in the fundamentals of hardware and software and how they integrate to form essential components of IT systems. It also enables the students to develop components of IT Systems.

Programme Outcomes addressed

1.Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

2.Graduates will demonstrate an ability to design a system, component or process as per needs and specifications.

3.Graduate will demonstrate skills to use modern engineering tools, software and equipment to analyze problems.

Competencies

1. Understand the working principle of Java Virtual Machine and the structure of Class Files
2. Develop applications by using the Java libraries
3. Understand the workings of the fundamental Android architecture, platform framework and emulator.
4. Apply the main components of Android APIs to develop Android Applications.
5. Create a database in SQlite and access it through Android platform

Assessment Pattern

	Bloom's Category	Test 1	Test 2	End-semester examination
1	Remember	20	20	20
2	Understand	40	40	30
3	Apply	30	30	40
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	10	10	10

Course Level Learning Objectives

Remember

1. What is the role of heap in JVM?
2. What is a ClassLoader?
3. What are the characteristics of mobile operating systems?
4. Distinguish Symbian S60 OS and Android
5. List out design goals of Android platform
6. What is Dalvik?
7. Differentiate Dalvik and JVM.

Understand:

1. When the JVM creates a frame?
2. Why is it important to maintain system compatibility for older applications?
3. What is the name of the interface used to represent collections that maintain non-unique elements in order? Select the one correct answer.
   - Collection
   - Set
   - SortedSet
   - List
   - Sequence
4. What will be the output from the following program?

```
import java.util.*;
public class Iterate {
public static void main(String[] args) {
        List l = new ArrayList();
        l.add("A"); l.add("B"); l.add("C"); l.add("D"); l.add("E");
        ListIterator i = l.listIterator();
        i.next(); i.next(); i.next(); i.next();
        i.remove();
        i.previous(); i.previous();
        i.remove();
        System.out.println(l);
};
};
```

5. Count the number of words in a sentence "This is a sample" using map API?
6. Why write a Class Loader?
7. Why we need a mobile operating system?

Apply:

1. Write java program to create a class "student" and derive the corresponding class structure.
2. Implement a simple event driven Java program using the Swing libraries

3. Write a java program to find out the list of students who have scored more than 80 percentage in a subject "Operating Systems" using JDBC.
4. Write an Android application to display the exam results in a table layout.
5. Define an intent object and invoke the components of Android platform.
6. How do you add sound and vibrate in your application using Android platform.

Create

1. Write a Java program to develop a basic shopping cart application that would allow the program user to add items to a shopping cart while browsing through a list of priced items. Once the user stops browsing and filling the cart, the program should produce the list of items ordered along with quantity and extended prices (price * quantity) plus an order total.

   Input

   Item file: A list of a dozen or more authentic priced items for sale (e.g. 401 Levi 501 19.95).

   Item orders from keyboard or screen selection: User item choices including the quantity of each item to be purchased.

   Output:

   i. List of priced items for sale;
   ii. View of the shopping cart;
   iii. Final order.
   iv. Processing:

   Provide an interface that allows the user to browse through a list of items for sale, choose an item and specify quantity to add to the cart, view the cart, allow more items to be added and items to be deleted from the cart, and choose to check out and view the final order with a detailed list of items ordered and the total amount to pay.
2. Design a user interface to demonstrate the various features of a product "car" using java2D.
3. Create a service named "MyService" to provide time service and define its life cycle. Create an Activity to interact with the created Service.
4. There are 20 staffs in the IT department. When a meeting is scheduled, a message "Meeting" is to be sent to all staffs. Implement a Broadcast Receiver in Android platform.
5. Design an Interactive interface to collect the information from a user in the registration form by using Alert Dialog and Progress Dialog components.

   Create an interface to perform the following tasks

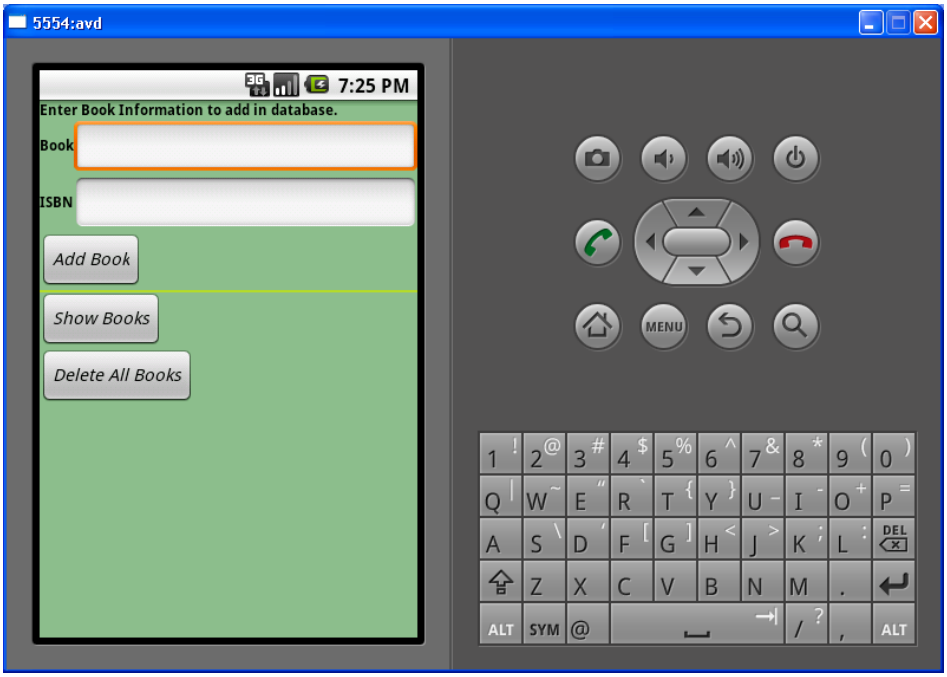
   i. Enter the Name of the book and ISBN number and click on

Add Book.

ii. Data will be added to Sqlite database.

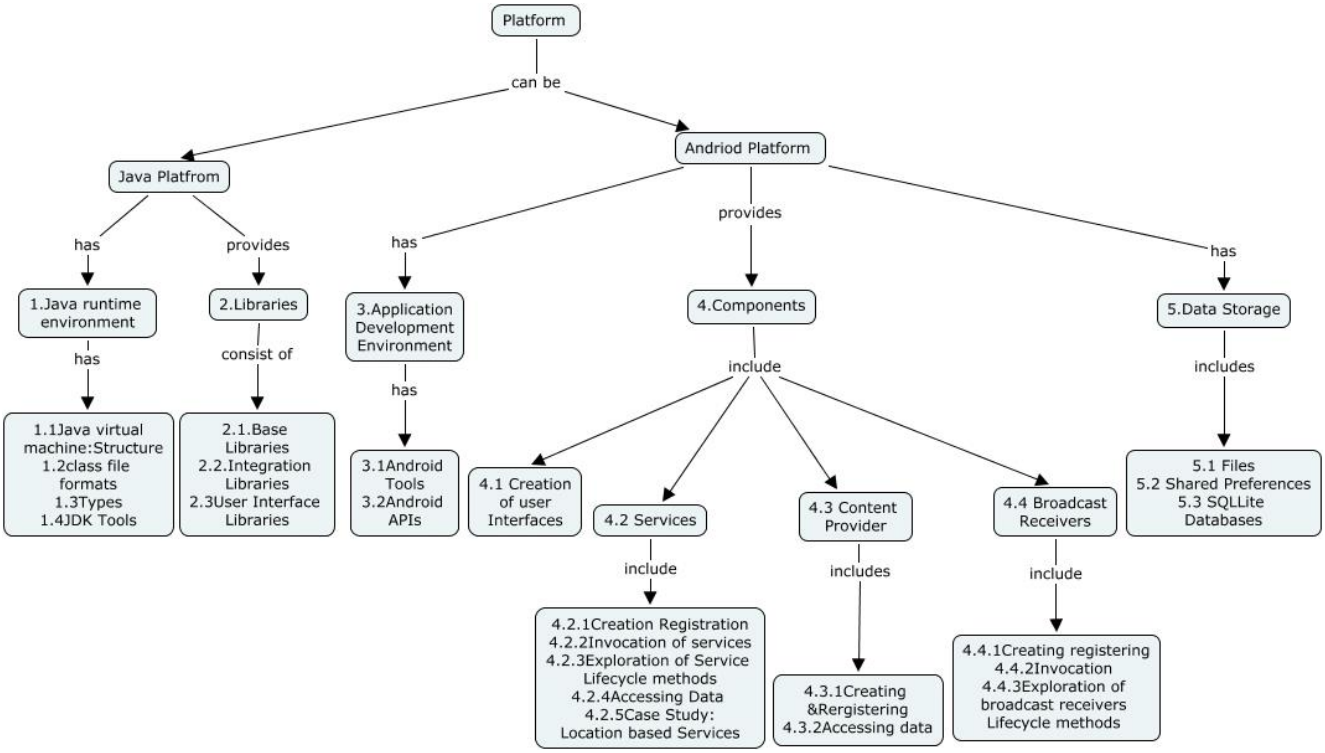
iii. Add multiple entries into database.

iv. Click on Show Books to view the contents added so far.



v. Click on Delete All Books button to delete the contents of database

Concept Map



Course contents and Lecture Schedule:

No.	Topic	No.of Lectures
1	**Java Platform**	
1.1	Java virtual Machine : Structure	2
1.2	Class file formats	2
1.3	Types : Server VM and Client VM	1
1.4	JDK Tools: Basic Tools, Security Tools, Java Deployment Tools, Java Plug-in Tools, Java Profiler, Java ANT	2
2	**Libraries**	
2.1	**Base Libraries**	
2.1.1	Collections Framework, Reflection, Java Archive (JAR) Files	2
2.1.2	Logging, Monitoring and Management	1
2.1.3	Package Version Identification	1
2.2	**Integration Libraries**	
2.2.1	Java Database Connectivity (JDBC) API	1
2.2.2	Java Naming and Directory Interface (JNDI) API	1
2. 3	**User Interface Libraries**	
2.3.1	Springs	1
2.3.2	Java 2D Graphics and Imaging	1
2.3.3	Sound , Accessibility	1
3	**Android Application Development Environment**	
3.1	Architecture	1
3.2	Application Development Environment	1
3.3	Android Tools: DDMS, Emulator, ADB, ADT, AVD	2
3.4	Android APIs	2
4	**Android Components**	
4.1	Creation of User Interfaces	1
4.2	**Services**	
4.2.1	Creation, registration	1
4.2.2	Invocation of services	1
4.2.3	Exploration of Service Lifecycle methods	1
4.2.4	Accessing data	1
4.2.5	Case Study: Location Based Services	

No.	Topic	No.of Lectures
4.3	**Content Providers**	
4.3.1	Creating and registering a content provider	1
4.3.2	Accessing data from content provider	1
4.4	**Broadcast Receivers**	
4.4.1	Creating and registering broadcast receivers	1
4.4.2	Invocation of broadcast receivers	1
4.4.3	Exploration of broadcast receivers Lifecycle methods	1
5	**Data Storage Mechanisms**	
5.1	Files	2
5.2	Shared Preferences	2
5.3	**SQLite database**	
5.3.1	Creating SQLite Database	2
5.3.2	Using databases in Android applications	2
	TOTAL	40

Syllabus

Java Platform - Java runtime environment - **Java virtual Machine** : Structure, Class file formats, Types : Server VM and Client VM , Java Vs .NET, **JDK Tools :** Basic Tools, Security Tools, Java Deployment Tools, Java Plug-in Tools Java Profiler, Java ANT, Java meet **Libraries : Base Libraries**: Collections Framework, Reflection, Java Archive (JAR) Files, Logging, Monitoring and Management, Package Version Identification **Integration Libraries** : Java Database Connectivity (JDBC) API, Java Naming and Directory Interface (JNDI) API **User Interface Libraries** : Springs , Java 2D Graphics and Imaging, Sound, Accessibility **Android Platform**: Architecture, Application Development Environment, Android Tools: DDMS, Emulator, ADB, ADT, AVD, Android APIs – Creation of User Interfaces **Android Services**: Creation, registration, invocation of services, Exploration of Service Lifecycle methods, Definition and use of a service interface, **Content Providers**: Creating and registering a content provider, Accessing data from content provider JDK Case Study:Location based Services **Broadcast Receivers**: Creating and registering broadcast receivers, Invocation of broadcast receivers, Exploration of broadcast receivers Lifecycle methods **Data Storage Mechanisms :** Files, Shared Preferences, SQLite database : Creating SQLite Database, Using databases in Android applications.

Text Books

1. The Java™ Virtual Machine Specification", Tim Lindholm, Frank Yellin, second edition, Sun Microsystems Press 2001
2. "Java: The Complete Reference, Seventh Edition", Herbert Schildt, McGraw-Hill Publications 2004
3. Professional Android Application Development, Reto Meier, Wrox, November 2008

Reference

1. Beginning Android, Mark Murphy, Apress, June 2009
2. Pro Android, Sayed Y Hashimi, Apress, June 2009
3. Android – Application Development, Rick Rogers et.al, O'Reilly, May 2009
4. "Hello, Android", Introducing Google's Mobile Development Platform, Ed Burnette, The Pragmatic Bookshelf, 2008

COURSE DESIGNERS:

1. S.Srinivasan ssnit@tce.edu
2. R.Muthu Selvi rmsit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T 57	0	0	-	1

T57 Platform Technologies Lab 0:1

Preamble: In this laboratory, the students will implement various Classes and interfaces that provide basic features and fundamental functionality for the Java and Android platforms.

List of Experiments:

Java Platform

1. Implementation of Reflection and Memory management
2. Implementation of Collection classes and algorithms
3. Bundling and compressing the files through JAR
4. Generation of Log Reports using Logging APIs
5. Implementation of JDBC API

Android Platform

1. Setup of Android Development Environment.
    - Design and configuration of an AVD
    - Execution of a simple application
2. Implementation of Activity and Layout
3. Implementation of Menus
4. Implementation of Dialogs and Notifications
5. Implementation of SQLite database
6. Mini project

COURSE DESIGNER:

P.Karthikeyan karthikit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T 58	0	0	1	1

T58: Network Security Lab 0:1

Preamble: Network security lab is the course which implements the classical encryption, symmetric encryption, and asymmetric encryption algorithms. This lab will enable the student to develop the security algorithms for real time applications and it gives hands on experience to the students in network security.

List of Experiments

1. Perform cryptanalysis of Caeser cipher using brute force and frequency analysis attack.
2. Perform cryptanalysis of Hill cipher using known Plain text - Cipher text attack.
3. Write a program for password cracking.
4. Develop a Confidential E transaction service using S-DES and DES algorithm
5. Implementation of Blow fish algorithm.
6. Write a program for Performance analysis of AES algorithm for varying key size, block size and number of rounds.
7. Develop an online voting system using RSA algorithm for authentication service.
8. Implementation of SHA algorithm.
9. Write a program to implement LSB Method in Steganography.
10. Implementation of multimedia security (like watermarking of video, audio).
11. Configuration of Firewalls.

Course Designers:

1. M.Amutha Prabakar ap_sse@tce.edu
2. P.Karthikeyan karthikit@tce.edu

REVISED CURRICULUM AND DETAILED SYLLABI

FOR

B.Tech DEGREE (Information Technology) PROGRAM

SIXTH SEMESTER

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2012-2013

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015

B.Tech. Degree (Information Technology) Program

SUBJECTS OF STUDY

(For the candidates admitted from 2012-2013 onwards)

SIXTH SEMESTER

Subject code	Name of the subject	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
T61	Operations Research	DC	3	1	-	4
T62	Accounting and Finance	DC	3		-	3
T63	Data Warehousing and Mining	DC	3	1	-	4
T64	Web Programming	DC	3	-	-	3
TCX	Elective 1	DC	3	-	-	3
XGX	Elective 2	DC	3	-	-	3
PRACTICAL						
T67	Web Programming Lab	DC	-	-	3	1
T68	Software Tools Lab	DC	-	-	3	1
Total			**18**	**2**	**6**	**22**

BS : Basic Science
HSS : Humanities and Social Science
ES : Engineering Science
DC : Department Core

L : Lecture
T : Tutorial
P : Practical

Note:

1 Hour Lecture/Tutorial is equivalent to 1 credit
2/3 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015

B.Tech. Degree (Information Technology) Program

SCHEME OF EXAMINATIONS
(For the candidates admitted from 2012-2013 onwards)

SIXTH SEMESTER

S.No	Sub. code	Name of the subject	Duration of Terminal Exam in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	T61	Operations Research	3	50	50	100	25	50
2	T62	Accounting and Finance	3	50	50	100	25	50
3	T63	Data Warehousing and Mining	3	50	50	100	25	50
4	T64	Web Programming	3	50	50	100	25	50
5	TCX	Elective 1	3	50	50	100	25	50
6	XGX	Elective 2	3	50	50	100	25	50
PRACTICAL								
7	T67	Web Programming Lab	3	50	50	100	25	50
8	T68	Software Tools Lab	3	50	50	100	25	50

* Continuous Assessment evaluation pattern will differ from subject to subject and for different tests. This will have to be declared in advance to students. The department will put a process in place to ensure that the actual test paper follow the declared pattern.

** Terminal Examination will be conducted for maximum marks of 100 and subsequently be reduced to 50 marks for the award of terminal examination marks.

Sub code	Lectures	Tutorial	Practical	Credit
T61	3	1	-	4

T61 OPERATIONS RESEARCH 3:1

(Common to Mech: G61)

Preamble: Operations research is a scientific approach to decision making that seeks to best design and operate a system, usually under conditions requiring the allocation of scarce resources.

The term *operations research* was coined during World War II when British military leaders asked scientists and engineers to analyze several military problems such as the deployment of radar and the management of convoy, bombing, antisubmarine, and mining operations. Because of the complexity of most real-world problems, it has been necessary for researchers and practitioners, when applying mathematical approaches, to reduce the complexity of the problem by either simplifying the problem or constraining it by making numerous assumptions.

A model is an abstraction or mathematical representation of a problem of interest and is an essential part of the process of solving that problem optimally. However, it is difficult, and sometimes impossible, to develop a mathematical model that addresses all aspects of the problem and its planning environment, since most real-world problems are too complex and involved. As the modelling approach provides solutions to the simplified or approximated problem, there may exist a significant discrepancy between those solutions and the subjectively expected realistic solution to the original problem.

Programme Outcomes addressed

b. Graduates will demonstrate knowledge of mathematics, science and engineering.

c. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

c. Graduates will demonstrate skills to use modern engineering tools, softwares and equipment to analyze problems.

d. Graduate can participate and succeed in competitive examinations.

Competencies: At the end of the course the student should be able to

1. Understand the phases of Operations Research study.
2. Formulate allocation problems mathematically in the frameworks of Linear Programming (LP) and Integer Programming (IP).
3. Solve LP problems by graphical, simplex and dual-simplex methods.
4. Solve IP problems using branch and bound and cutting plane method.

5. Formulate transportation and assignment problems as LPP.
6. Solve transportation and assignment problems using special algorithms.
7. Model project scheduling problem as a Network.
8. Solve project scheduling problems using PERT and CPM.
9. Solve inventory problems formulated as purchase, production, and news-paper boy models.
10. Solve single channel and multi-channel queuing problems.
11. Solve replacement problems.
12. Solve sequencing and production scheduling problems.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3 / End-semester examination
1	Remember	10	10	10
2	Understand	10	10	10
3	Apply	80	80	80
4	Analyze	0	0	0
5	Evaluation	0	0	0
6	Create	0	0	0

Course Level Learning Objectives

Remember

1. What are the phases of operations research?
2. What do you mean by degeneracy in transportation problem?
3. What is fathoming?
4. Define Lead time.
5. What do you mean by Kendall's notation.
6. What do you mean by Travelling Salesman Problem?
7. Define Mixed Integer programming problem.
8. What is dummy activity and when is it needed?
9. Define Traffic Intensity.

Understand

1. What is the importance of the slack variables in simplex method?
2. How to find that a LPP has got an alternate optimal solution from the optimal simplex table?
3. Write the mathematical formulation of Assignment problem.
4. Differentiate PERT and CPM.
5. How does a fixed order quantity system (Q-system) and periodic review system (P-system) differ in placing an order?

Apply

1. A company manufacturer two types of products, P1 and P2. Each product uses lathe and milling machine. The processing time per unit of P1 on the lathe is 5 hours and on the milling machine is 4 hours. The processing time per unit of P2 on the lathe is 10 hours and on the milling machine is 4 hours. The maximum number of hours available per week on the lathe and milling machine are 60 hours and 40 hours, respectively. Also, profit per unit of selling P1 and P2 are Rs.6 and Rs.8, respectively. Formulate a linear programming model to determine the production volume of each of products such that the total profit is maximized.
2. Solve the following transportation problem

	I	II	III	IV	Supply
A	4	6	8	13	50
B	13	11	10	8	70
C	14	4	10	13	30
D	9	11	13	8	50
Demand	25	35	105	20	

3. A travelling salesman has to visit five cities. He wishes to start from a particular city, visit each city once and then return to his starting point. Cost of travel from one city to another is shown below. Determine the least cost route.

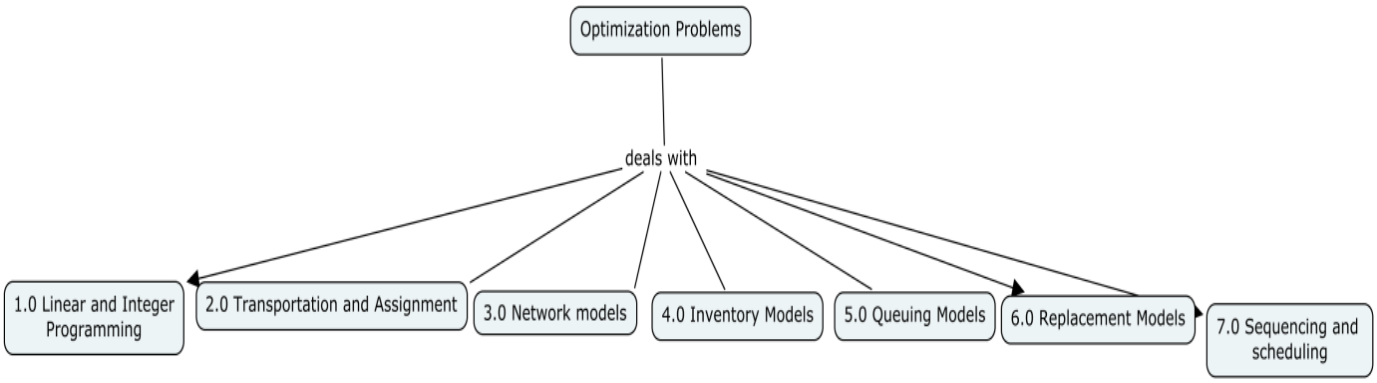
		To				
		A	B	C	D	E
From	A	∞	4	10	14	2
	B	12	∞	6	10	4
	C	16	14	∞	8	14
	D	24	8	12	∞	10
	E	2	6	4	16	∞

4. The following table provide the activities in a construction project and other relevant information:

Activity	Normal		Crash	
	Time (Days)	Cost (Rs.)	Time (Days)	Cost (Rs.)
1-2	8	100	6	200
1-3	4	150	2	350
2-4	2	50	1	90
2-5	10	100	5	400
3-4	5	100	1	200
4-5	3	80	1	100

Assume indirect cost is Rs. 100 per day. Suggest the project duration for least cost by analyzing the effect of reduction in cost by crashing.

Concept Map



Course Contents and Lecture schedule

S.No	Topics	No. of Lectures
	Introduction: Phases of Operations Research	1
1	**Linear Programming**	
1.1	Linear Programming (LP): Formulation	2
1.2	Graphical Method for LP problems	2
1.3	Simplex and Dual Simplex Methods for LP problems	3
2	**Integer Programming**	
2.1	Integer Programming(IP): Formulation	1
2.2	Branch and Bound Method for IP problems	2
2.3	Cutting Plane Method for IP problems	2
3	**Network Models**	
3.1	Transportation Problem(TP): LP Formulation	1
3.2	Initial Basic Feasible solutions for TP using North West Corner Method, Least Cost Method and Vogel's Approximation Method	1
3.3	Optimal solution for TP using Modified Distribution Method (MODI)	2
3.4	Assignment Problem (AP): Formulation	1
3.5	Hungarian Method for AP	1
3.6	Project Management: Network Construction – Terminologies – LP Formulation	2
3.7	Critical Path Method (CPM) and crashing	3
3.8	Programme Evaluation Review Technique (PERT)	2
4	**Inventory Models**	
4.1	Purchase Model (with and without shortages)	2

S.No	Topics	No. of Lectures
4.2	Production Model (with and without shortages)	2
4.3	Newspaper Vendor Problem	1
4.4	P-System and Q-System	1
5	**Queuing Models** (Poisson Arrival and Exponential Service pattern)	
5.1	Queuing Terminologies and applications	1
5.2	Single Channel – Infinite population Queue	1
5.3	Single Channel – Finite population Queue	1
5.4	Multiple Channel – Infinite population Queue	1
6	**Replacement Models**	
6.1	Replacement of suddenly detroiting system	2
6.2	Replacement of gradually detroiting system	2
7	**Sequencing and Scheduling**	
7.1	Scheduling objectives	1
7.2	Setup time dependent single machine scheduling	1
7.3	Johnson's algorithm for 'n' jobs 2-machines flow-shop scheduling	1
7.4	Johnson's algorithm for 'n' jobs 3-machines flow-shop scheduling	1
7.5	Graphical method for '2' jobs 'k'-machines job-shop scheduling	1

Syllabus

Linear Programming

Phases of Operations Research

Linear Programming: Formulation - Graphical Method and Simplex Method (2 variable problems only)

Integer Programming

Integer Programming: Formulation - Branch and Bound Method and Cutting Plane Method (2 variable problems only)

(Note: Problems with more than 2 variables and more number of constraints to be practiced in LINDO/LINGO/Excel/ILog as Programming Assignments)

Network Problems

Transportation Problem: Formulation - Initial solutions using North West Corner Method, Least Cost Method and Vogel's Approximation Method - Modified Distribution Method (MODI)

Assignment Problem: Formulation - Hungarian Method

Project Management: Network Construction – Terminologies - Critical Path Method (CPM) – crashing - Programme Evaluation and Review Technique (PERT)

Inventory Models

Purchase Model (without and with shortages) - Production Model (without and with shortages) - Newspaper Vendor Problem - P-System and Q-System

Queuing Models

(Poisson Arrival and Exponential Service pattern)

Basic Terminologies - Single Channel with infinite population queue - Single Channel with finite population queue - Multiple Channel with infinite population Queue.

Replacement Models

Replacement of suddenly deteriorating system - gradually deteriorating system.

Sequencing and Scheduling

Scheduling objectives – setup dependent single machine – Johnson's algorithm for 'n' jobs 2/3-machines flow-shop – Graphical method for '2' jobs 'k'-machines job-shop.

Text Book:

1. Sharma.J.K., "Operations Research : Theory and applications", Macmillan India Ltd., Fourth Edition, 2009.
2. Hamdy A. Taha, "Operations Research - An Introduction", MacMillan Co., Seventh Edition 2003.

Reference Book:

1. Panneerselvam, R., "Operations Research", Prentice Hall, Second Edition, 2007.
2. Ravindran A., Don. T. Phillips, and James J. Solberg, "Operations Research - Principles and Practice", John Wiley and Sons, Second Edition, 2000.
3. Hiller / Lieberman, "Introduction to Operations Research" Tata McGraw Hill, Seventh Edition, 2001
4. Wayne L. Winston, "Operations Research: Applications and Algorithms", Thomson Brooks/Cole, Fourth Edition, 2003.
5. Ronald L Rardin, "Optimisation in Operations Research" Pearson Education Asia, First Indian reprint, 2002.

Course Designers:

N. Jawahar – jawahartce@tce.edu

S. Saravana Perumaal – sspmech@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T62	3	0	-	3

T62 Accounting and finance 3.0

(Common to Mech: G56)

Preamble: Engineering profession involves lots of decision making. The decisions may range from operation to non-operation. For taking decisions of these kinds an engineer needs among other things data about the organizations routine operations and non-routine operations. Accounting is a science which provides all the data by recording, classifying, summarizing and interpreting the various transactions taking place in an organization and thereby helps an engineer in taking vital decisions in an effective manner. Finance is an allied but a separate field relying on accounting, and enables engineers in taking useful financial and cost related decisions by providing well defined concepts, tools and techniques.

Program outcomes addressed

- Engineering graduates will understand the basic concepts, processes, tools and techniques of accounting and finance.
- Engineering graduates will apply the concepts, processes, tools and techniques of accounting and finance and take effective decisions in organizational settings.

Competencies:

At the end of the course, the students will

1. Develop an understanding about what accounting is and its importance in decision making.
2. Understand the recording function of accounting.
3. Understand the classification function of accounting.
4. Understand the summarizing function of accounting.
5. Understand the analysis and interpretation function of accounting.
6. Perform the various functions of accounting.
7. Prepare trial balance and there from financial statements like trading account, Profit & loss account and balance sheet.
8. Interpret the financial statements of an organization.
9. Understand the meaning of financing and its functions and objectives.
10. Understand some of the basic concepts, tools and techniques of finance and their applications.

Assessment pattern

S.No	Bloom's category	Test 1	Test 2	Test 3/End-Semester examination
1	Remember	20	20	20
2	understand	30	30	30
3	Apply	50	50	50
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	create	0	0	0

Course level learning objectives

Remember

1. The meaning of the term "accounting" and definition of accounting
2. The concepts and conventions of accounting.
3. Its importance in decision making.
4. The functions of accounting
5. The meaning of the term Depreciation and causes of Depreciation
6. The meaning of the term "Cost accounting"
7. The bases of cost classification
8. The relationship between volume of output cost of production and profit
9. The meaning of the terms "Budget and Budgetary control"

Understanding

1. Understand the definition of accounting
2. Explain the various functions of accounting.
3. Discuss the concepts and conventions of accounting.
4. Understand the process of preparing final accounts.
5. Understand the concept of depreciation and methods of providing depreciation
6. Explain the classification of cost.
7. Describe the process of preparing cost sheet.
8. Discuss the importance of budgets and budgetary control
9. Understand the functions of financing
10. Explain the process of preparing working capital budget.

Apply

1. Journalise the following business transactions:

   A) A brings in cash Rs.10, 000 as the capital and purchases land worth Rs.2000.

   b) He purchases goods worth Rs.5, 000.

   C) He returns goods worth Rs.500 as they are defective.

d) He sells goods for Rs.7, 000.

e) He incurs traveling expenses of Rs.200.

2. Record and classify the following transactions in the books of Suresh

Suresh introduces capital of Rs.20, 000 into his business.

He purchases furniture worth Rs.2000.

He purchases goods worth Rs, 8,000.

He incurs Rs.200 as freight expenses.

He sold goods for cash Rs.5, 000 and for credit Rs.2000

He paid salary Rs.3, 000

He paid electricity expenses Rs.800.

3. Prepare Trading and profit and loss account and Balance sheet on 31.12.96 from the following trial balance extracted from the books of Mr. Kumar as on 31.12.96.

Debit Balances	Rs.	Credit Balances	Rs.
Buildings	30,000	Capital	40,000
Machinery	31,400	Purchase returns	2,000
Furniture	2,000	Sales	2,80,000
Motor car	16,000	Sundry creditors	9,600
Purchases	1,88,000	Discounts received	1,000
Sales return	1,000	Provision for bad and doubtful debts	600
Sundry debtors	30,000		
General expenses	1,600		
Cash at bank	9,400		
Rates and taxes	1,200		
Bad debts	400		
Insurance premium	800		
Discount allowed	1,400		
Opening stock	20,000		
Total	3,33,200		3,33,200

4. Senthil purchased machinery for Rs.4, 00,000 on 1st April 2000.On 1st April 2001 additional machinery was purchased for Rs.40, 000.prepare the asset account for three years. Depreciation is to be provided at 10%p.a using straight line method. The firm closes its books on 31st March of every year.

5. A factory is currently working at 50% capacity and the product cost is Rs.180 per unit as below:

Material ---------- Rs.100

Labor ---------- Rs.30

Factory overhead— Rs.30 (40%fixed)

Administration overhead Rs.20 (50% fixed)

The product is sold at Rs.200 per unit and the factory produces 10,000 units at 50%capacity.

Estimate profit if the factory works to 60% capacity. At 60%working raw material increases by 20% and selling price falls by 20%.

6**.** The following particulars are extracted from the books of a company relating to commodity "A" for the half year ending 30th June1993.

Purchase of raw materials ---------- Rs.1, 32,000

Direct wages ---------- Rs.1, 10,000

Rent, rates, insurance and works cost-------- Rs.44, 000

Carriage inward --------- Rs.1584

Stock on 1-1-93

Raw materials---------- Rs.22, 000

Finished product (1600 tones) - Rs.17, 000

Stock on 30-6-93

Raw materials ---------- Rs.24, 464

Finished products (3,200 tones) ---------- Rs.35, 200

Work-in-progress on 1-1-93 ---------- Rs.17, 600

Work-in-progress on 30-6-93 ------ Rs.5280

Factory supervision ---------- Rs.8, 800

Sales-Finished products ---------- Rs.33, 000

Advertising discount allowed and selling cost at Re.0.75per tones sold.25, 600 tones of commodity was sold during the period.

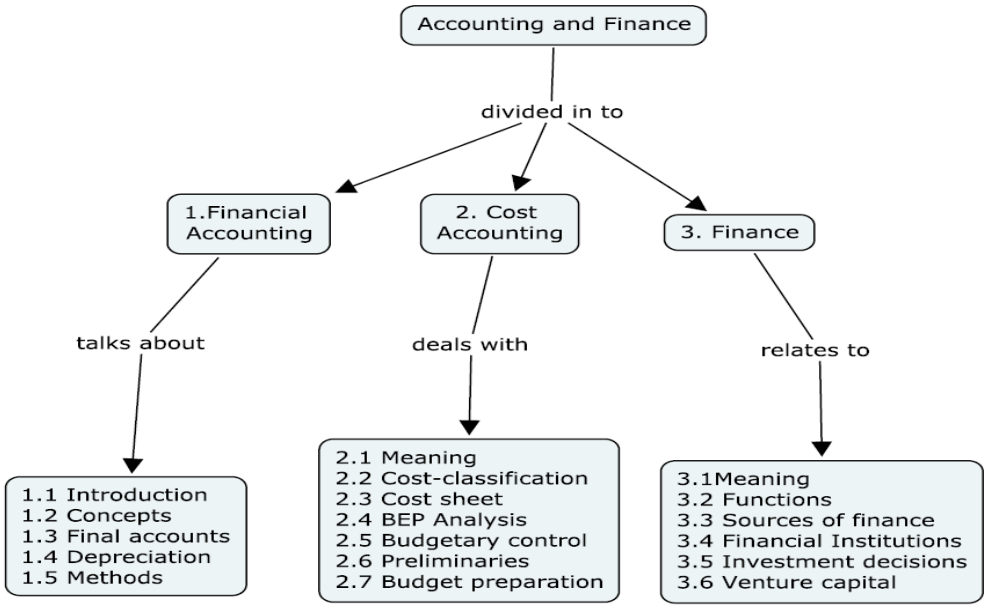
You are required to ascertain:

i) prime cost ii) factory cost iii) cost of sales iv) profit v) No of tones of the commodity sold.

7. From the following information calculate the Break even point in terms of units and Break even point in terms of sales

Sales in Rs.10, 000, Variable costs (direct material and direct labor) Rs.6, 000, Fixed cost Rs.2, 000 profit Rs.2, 000, No. of units produced 1,000 units.

Concept Map



Syllabus

Accounting: Introduction and Definition-Accounting concepts and conventions-final Accounts-Preparation of Trading, Profit and Loss account and Balance sheet.Depriciation-Meaning-Need and objectives-Basic factors-Methods for providing depreciation.

Cost Accounting: Meaning and importance-Cost-Elements of Cost-cost Classification-Preparation of cost sheet-Material costing-Valuation of purchases-pricing of material issues. Break-even analysis-managerial applications. Budgetary control-Introduction-objectives of budgetary control-preliminaries for operation of budgetary control-Budgets-types of budgets and their preparation.

Finance: Meaning-Definition-Objectives-functions of finance-source of finance-short-term, Long-term and medium-term-Role of special financial institution in financing-Investment decisions-short-term Investments and long-term investments-Venture Capital.

Text Books:

1. M.c.Shukla, T.s.Grewal, S.c.Gupta: "Advanced Accouts-volumeI", 2007 Reprint, S.Chand&Company Ltd.2007.
2. S.P.Jain, K.L.Narang: "Advanced Accountancy-Volume I", Thirteenth Revised Edition, Kalyani Publishers.2006.
3. V.K.Saxena, C.D.Vashist: "Advanced cost and Management Accounting", Seventh Enlarged Edition, Sultan Chand and Sons, 2008.

Reference Books:

1. Prasanna Chandra, "Financial Management-Theory and Practice" Sixth Reprint, Tata McGraw-Hill publishing company Limited, 2007.

2. Ramachandra Aryasri, A, Ramana Moorthy, V.V, Engineering Economics and financial Accounting", Tata McGraw hill, 2007.
3. S.N.Maheswari, "Advanced accountancy"Vikas publishing, 2007.

Course content and lecture schedule

S.No	Topics	No. of Lectures
1.	**Financial Accounting**	
1.1	Introduction and Definition	1
1.2	Accounting concepts and conventions	2
1.3	Final Accounts- Preparation of Trading, Profit &Loss account and Balance sheet.	6
1.4	Depreciation –Meaning-Need and objectives	2
1.5	Basic factors-Methods for providing depreciation	3
2.	**Cost Accounting**	
2.1	Meaning and importance	2
2.2	Cost-Elements of cost-Cost classification	2
2.3	Preparation of Cost sheet-Material costing-valuation of purchases-pricing of material issues.	6
2.4	Break-even analysis-managerial applications	2
2.5	Budgetary control-introduction-objectives of budgetary control	1
2.6	Preliminaries for operation of budgetary control	1
2.7	Budget-Types of budgets and their preparation	4
3	**Finance**	
3.1	Meaning-Definition-objectives	2
3.2	Functions of finance	1
3.3	Source of finance-short-term, medium-term, long-term	2
3.4	Role of special financial institutions in financing	2
3.5	Investment decisions-Short-term investments and long-term investments	5
3.6	Venture capital	2
Total		**46**

Course Designers:-

1. S. Dhanasekaran - sdmech@tce.edu
2. P. S. Boopathi Manickam - psbmeco@tce.edu

Subject Code	Lecture	Tutorial	Practical	Credit
T63	3	1	--	4

T63 Data Warehousing and Mining 3:1

Preamble

A data warehouse is a powerful database model that significantly enhances the user's ability to quickly analyze large, multidimensional datasets. Data Mining is a type of database analysis that attempts to discover useful patterns or knowledge from large datasets and to discover previously unknown relationships among the data. The analysis uses advanced statistical methods, such as cluster analysis, and sometimes employs artificial intelligence or neural network techniques.This course provides a practical line for planning, building and managing a successful data warehouse and better knowledge about all Data Mining processes.

Program outcomes addressed

Graduates will demonstrate

a. An ability to apply the knowledge of engineering, information technology, mathematics and science.

b. An ability to design and conduct experiments, as well as analyze and interpret data.

e. An ability to identify, formulate and solve engineering problems.

k. An ability to use techniques, skills, and modern engineering tools to implement and organize engineering works under given constraints

Competencies

The students will be able to

1. Understand the data warehouse project.
2. Apply the data extraction, transformation, and loading (ETL) functions.
3. Analyze the various data preprocessing techniques.
4. Derive Association rules using different algorithms.
5. Apply different types of Classification and Prediction.
6. Analyze the performance of Classifier and Predictor.
7. Explain different Clustering methods.
8. Understand Spatial mining and Temporal mining.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test3	End-semester examination
1	Remember	20	20	20	20
2	Understand	40	30	30	30
3	Apply	40	40	40	40
4	Analyze	0	10	10	10
5	Evaluate	0	0	0	0
6	Create	0	0	0	0

Course Level Learning Objectives

Remember

1. List the measures in multi-dimensional data warehouse.
2. List the types of OLAP Servers.
3. List the strategies for data reduction.
4. What are the techniques involved in attribute subset selection
5. Define Min-Max and Z-Score normalization.
6. Define data transformation.
7. Define the association rule mining view.
8. Define fuzzy set approach.
9. Define EM algorithm.
10. Define temporal mining.

Understand

1. Explain the operations in the multi-dimensional data model.
2. Data mining as a step in the process of knowledge discovery .Discuss
3. Draw the architecture of a typical data mining system.
4. Explain data cleaning.
5. Demonstrate discretization and concept hierarchy generation for numerical data.
6. Describe about mining various kinds of Association rules.
7. Explain the algorithm for inducing a decision tree from training tuples.
8. Explain Bayes Theorem and Bayes Belief networks.
9. Explain the neural network learning for classification using the back propagation algorithm.
10. Demonstrate the types of data in cluster analysis.

Apply

1. Consider the data warehouse for big University with four dimensions of semester, courses, administration and Student and the two measures are count and avg_grade. Apply star and snowflake schema for that database.

2. Consider the following contingency table

Data	Male	Female	Total
Patients	250	200	450
Non-patients	50	1000	1050
Total	300	1200	1500

Apply data integration by correlation analysis.

3. A data base has 5 transactions Let min_sup=60% and min_conf=80%.

Transaction ID	Items purchased
T1	{T,A,K,E}
T2	{M,A,K,E}
T3	{J,A,C,K}
T4	{M,I,C,K,Y}
T5	{C,O,C,K}

(a) Find all the frequent item sets using priori and FP-Growth, respectively. How can we improve the efficiency of Apriori -based mining? Compare the efficiency of the two mining process.

(b) List all the strong association rules.

4. The following table consists of training data from an Employee database. The data have been generalized .For a given row entry, count represents the number of data tuples having the values for department, status, age, and salary in that row

Department	Status	Age	Salary	Count
Sales	Senior	31-35	45,000-50,000	30
Sales	Junior	26-31	25,000-30,000	20
Sales	Junior	31-35	30,000-35,000	25
Marketing	Senior	31-35	45,000-50,000	30
Marketing	Junior	26-31	25,000-30,000	20
Marketing	Junior	31-35	30,000-35,000	25
Manufacturing	Senior	31-35	55,000-	8

			60,000	
Manufacturing	Junior	26-31	45,000-50,000	2
Manufacturing	Junior	31-35	45,000-50,000	5
Manufacturing	Senior	35-40	60,000-65,000	3

Let status be the class label attribute.

(a)How would you modify the basic decision tree algorithm to take into the consideration the count of each generalized tuple (i.e., for each row entry)?

(b) Given a data tuple having the values "Marketing"," 26-31" and "25,000-30,000" for the attributes department ,age and salary respectively ,what would be the naïve Bayesian classification of the status for the tuple be?

5. Consider the following example. Apply DT and NB classification algorithms and report the performance of each algorithm.

MOTOR	WHEELS	DOORS	SIZE	TYPE	CLASS
NO	2	0	Small	Cycle	bicycle
NO	3	0	Small	Cycle	tricycle
YES	2	0	Small	Cycle	motorcycle
YES	4	2	Small	automobile	Sports car
YES	4	3	medium	automobile	Mini van
YES	4	4	medium	automobile	sedan
YES	4	4	large	automobile	sumo

Analyze

1. Compare the processes of OLTP and OLAP.

2. Give three reasons why you think ETL functions are most challenging in a data warehouse environment.

3. Use the NB and J48 learning schemes, with the default settings to analyze the weather data. Use training set as the test option.

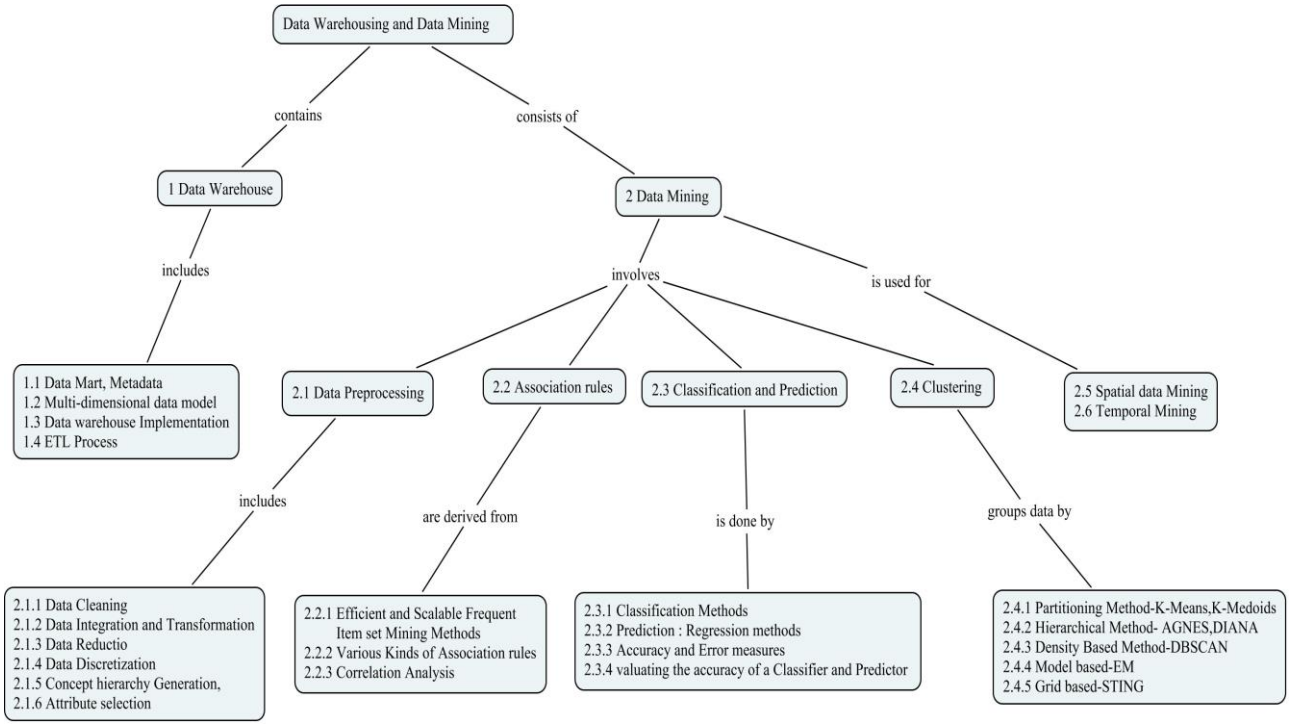
a)Which of these classifiers are you more likely to trust when determining whether to play? Why?

b) Report model percent error rate.

4. Differentiate the strength and weakness of K-means with k-medoids, hierarchical, density, and model based clustering algorithms.

5. Examine the usage of spatial measures in spatial data cube construction. Conclude which is the possible and suitable way under which condition.

Concept Map



Syllabus

Data Warehouse- Data Mart, Metadata, Multi-dimensional data model, Data warehouse Implementation, ETL Process. **Data Mining-**Data Mining Goals, Stages of the data mining Process, Data Mining Functionalities. **Data Preprocessing-**Data Cleaning, Data Integration and Transformation, Data Reduction, Data Discretization and Concept hierarchy Generation, Attribute Selection. **Association rules -**Efficient and Scalable Frequent Item set Mining Methods- Apriori, FP Tree. Mining Various Kinds of Association rules, Correlation Analysis. **Classification and Prediction-** Issues in Classification and Prediction, Classification Methods-Decision Tree, Bayesian Classification, Rule based Classification, Back propagation, Support vector machines, Associative classification, Lazy learners. Prediction - Regression methods. Accuracy and Error measures, Evaluating the accuracy of a Classifier and Predictor. **Clustering-** Issues, Clustering Methods: Partitioning Method-K-Means,K-Medoids, Hierarchical Method-AGNES,DIANA, Density Based Method- DBSCAN, Grid based-STING, Model based-EM .**Spatial data Mining, Temporal Mining, R Tool Practice, Case Study- Healthcare System.**

Textbook

Jiawei Han and Micheline Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, 2nd edition, 2006.

Reference Books

1. Ian H. Witten and Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann Publishers, 3rd edition, 2011.

2. Dunham, M., Data Mining: Introductory and Advanced Topics. New Jersea: Pearson Education, Inc, 2003.

3. Hand, D., Mannila, H. and Smyth, P., Principles of Data Mining. Cambridge, Mass.: MIT Press, 2001.

Some useful links

1. http://R-project.org

Course content and Lecture Schedule

No.	Topic	No. of Lectures
1.	**Data Warehouse**	
1.1	Data Mart, Metadata	2
1.2	Multi-dimensional data model	4
1.3	Data warehouse Implementation	1
1.4	ETL Process	2
2.	**Data Mining** -Goals, Stages of the data mining Process, Data Mining Functionalities	3
2.1	**Data Preprocessing**	
2.1.1	Data Cleaning	1
2.1.2	Data Integration and Transformation	1
2.1.3	Data Reduction	1
2.1.4	Data Discretization	1
2.1.5	Concept hierarchy Generation	1
2.1.6	Attribute Selection	1
2.2	**Association rules**	
2.2.1	Efficient and Scalable Frequent Itemset Mining Methods- Apriori, FP Tree	4
2.2.2	Mining Various Kinds of Association rules	1
2.2.3	Correlation Analysis	1
2.3	**Classification and Prediction**	
2.3.1	Classification Methods Issues-Decision Tree, Bayesian Classification, Rule based Classification, Back propagation, Support vector machines, Associative classification, Lazy learners	6
2.3.2	Prediction-Regression methods	2

2.3.3	Accuracy and Error measures	1
2.3.4	Evaluating the accuracy of a Classifier and Predictor	1
2.4	**Clustering**	
2.4.1	Partitioning Method –K-Means, K-Medoids	2
2.4.2	Hierarchical Method-AGNES,DIANA	1
2.4.3	Density Based Method-DBSCAN	1
2.4.4	Model based-EM	1
2.4.5	Grid based-STING	1
2.5	**Spatial Mining**	**1**
2.6	**Temporal Mining**	**1**
	R Tool Practice	2
	Case Study- Healthcare System	**2**
	Total Lectures	**46**

Course Designers

1. Mrs.K.V.Uma kvuit@tce.edu

2. Mrs.M.Nirmala Devi mnit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T64	3	0	-	3

T64 WEB PROGRAMMING **3:0**

(Common to CSE/IT: C55)

Preamble: Students will learn how to represent structure and how to transport data using XML and XML related technologies and protocols. Students gain understanding of how the Internet application works and develop programming skills for developing Rich Internet Applications. This course will establish a professional, client-based attitude towards web-design. Students will communicate effectively using today's technologies.

Program me Outcomes addressed

a. Graduates will use middleware wherever necessary to integrate data sources.

b. Graduates will able to use vendor support for XML.

c. Graduates will learn how to do client side as well as server-side programming.

d. Graduates will create and develop interactive web pages on the World Wide Web.

Competencies

1. Understand the architectural models of an enterprise system.
2. Understand how the Internet works with HTML, XML, Web services.
3. Understand how Internet application works in general.
4. Construct XML Documents.
5. Use XSL for transformations and formatting.
6. Use XML schemas and DTDs to construct data structure, constraints and type enforcement.
7. Develop server side programming for web application development
8. Develop programming skills in Rich Internet Applications like AJAX, Flex.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	End-semester examination
1	Remember	20	20	10
2	Understand	60	20	10
3	Apply	20	60	60
4	Analyze	0	0	0

5	Evaluate	0	0	0
6	Create	0	0	20

Course Level Learning Objectives:

Remember:

1. Explain the advantages of hybrid 2 tier architecture?
2. What are the String Object XHTML markup methods?
3. What is a SOAP message?
4. What are XML Parsers?
5. List ways that Web 2.0 can be used to make you money.
6. Compare the differences between IIS/Apache and Windows/Linux in regards to web development.
7. List out JSF Components.

Understand:

1. Write a CSS Rule to change color of all elements containing attribute class = "greenmove" to green and shift them down to 25 pixels and right 15 pixels?
2. Write a Java Script to get input from the user in XHTML and convert to lowercase and uppercase?
3. How to include element dimensions in style sheets?
4. Write a DTD for the XML Document formed from the above table?
5. How to display the information about the Web Browser?
6. Write a simple test program using PHP on the local server (http://localhost/)
7. How to create a Full-scale Ajax-Enabled Application?

Apply:

1. Draw the XML Schema for Book Publisher along with List of Books.
2. Write an XML and DTD for Storing Recipes.
3. Construct an XML Doc that marks up the information in the following table (10)

Name	Job	Department	Cubicle
Joe	Programmer	Engineering	5E
Erin	Designer	Marketing	9M
Melisa	Designer	Human Resource	8H
Craig	Admin	Engineering	4E
Eileen	Proj Coordinator	Marketing	3M
Danielle	Programmer	Engineering	12E

Frank	SalesPerson	Marketing	17M
Corinne	Programmer	TechSupport	19T

4. Implement the Remote Interface by using an Server side program and client side program containing the Remote Objects definition.
5. Construct an XML schema for the book's XML document
6. Construct the book's XML document using cascading style sheets.
7. Draw a picture describing the relationship between client/server objects used by PHP and mySQL

Create:

1. Create an XHTML document that marks up your resume? (10)
    a. The first document will establish the **frameset**, which will be split into at least **three** frames. One of these frame windows must contain a document that provides navigation through this set of documents (a table of contents).
    b. The rest of the set will contain, at a minimum, **six** other documents, **four** of which must pass through an active frame. These may include documents that you have created for other assignments, or other pages you have created, but if, you choose to do this, you are responsible for the markup in these pages.
    c. One document in this set must contain a working **form** with at least **three** different types of **input elements**.

    It will also include:

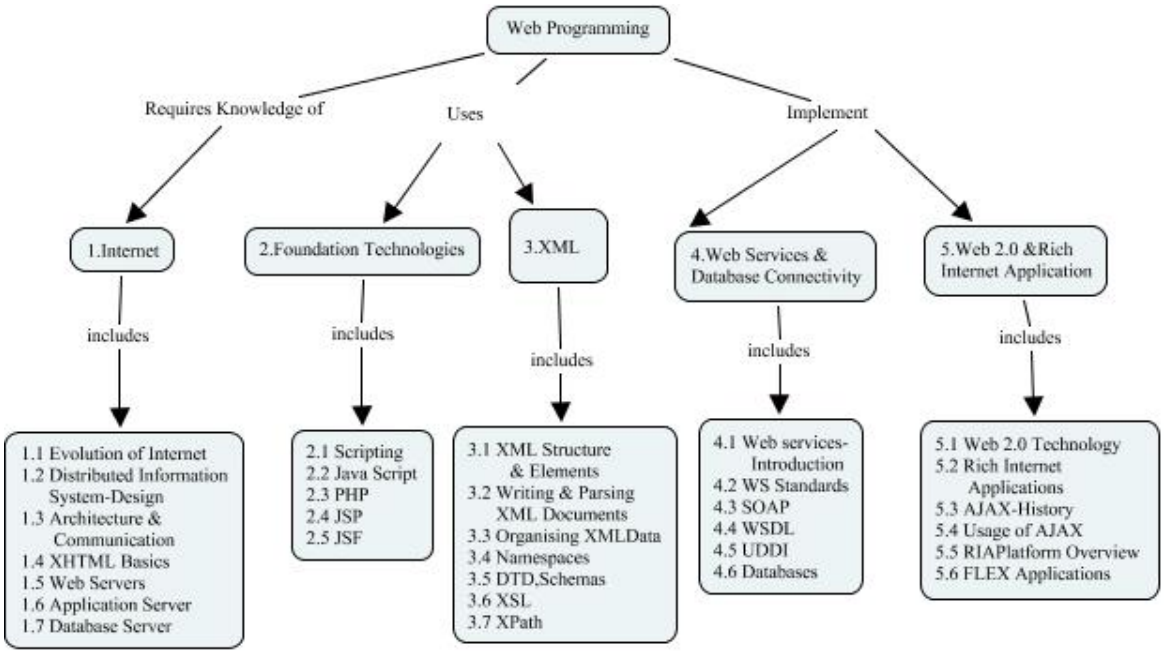
    d. At least **two** external links to documents you do not own;
    e. At least **two** links to other documents you have created;
    f. At least **one** background color or image;
    g. At least **three** images in the body of your resume pages; and
    h. At least **two** targeted links between frames
2. Create an XHTML document titled " How to get Good Grades" Use <meta> tags to include a series of keywords that describe your document?

```
<?xml version = "1.0"?>
<!DOCTYPE html  PUBLIC "/W3XX//DTD XHTML 1.1//EN"
http://www.w3.org/TR/xhtml1/DTD/xhtml1.dtd>
<html xmlns = "http://w3.org/1999/xhtml">
<head>
<title> How to get Good Grades"</title>
```

```
<meta name = "keywords" content = " Way to read, study,
Understanding, concentrate, seek advice, group study"/>
</head>
<body>
<p> concentrate and seek advice to clarify the doubts</p>
</body>
</html>
```

3. Create a Web Page that applies the invert filter to all image if the user moves the mouse over it?
4. Create an XHTML document that marks up your resume?
5. Create an XHTML document titled "How to get Good Grades" Use <meta> tags to include a series of keywords that describe your document.
6. Use PHP sessions, Create persistent variables that can be used by multiple Requests from a client browser session.
7. Create a Working environment for PHP Web page development.

Concept Map



Course Content and Lecture Schedule

No	Topic	No of Lectures
1.	**Architecture models (10)**	
1.1	History-Evolution of Internet, Distributed Information Systems-Design	1
1.2	Architecture of an Information system, Communication in an Information system	2
1.3	XHTML-Basics, Headers, Linking	1
1.4	Images, Image map, meta elements	1
1.4	Frameset, cascading style sheet	2
1.5	HTTP-Limitations of HTTP, Web servers-IIS & Apache	2
1.6	Application servers	1
2.	**Foundation Technologies (12)**	
2.1	Introduction to scripting, control statement, function, objects	2
2.2	Java script	2
2.3	PHP-Basics, String processing and Regular Expressions	2
2.4	Connecting to a Database, Using cookies, Dynamic content	2
2.5	JSP-Java Web Technologies, creating and Running a simple application in Netbeans	2
2.6	JSF Components, Session Tracking	2
3.	**XML Basics (10)**	

3.1	XML Structure, Elements	1
3.2	writing and Parsing XML Document	1
3.3	Organizing XML data, adding attributes	2
3.4	XML Namespaces, Working with DTD, Schemas	3
3.5	Elements,types,groups,defining attributes	1
3.6	XSL,XPATH	2
4	**Web services and Database Connectivity (8)**	
4.1	Need of WS,WS Standards	1
4.2	SOAP-Goals, Structure and contents of SOAP Messages	2
4.3	Binding SOAP Message	1
4.4	WSDL-Goals, Structure	1
4.5	UDDI-Goals, Registry, Data structures	1
4.6	Databases: RDBMS, MySQL , SQL	2
5	**Web 2.0 and Rich Internet Application (10)**	
5.1	Web 2.0-Introduction,Technology overview, Integrating with web services	2
5.2	Rich Internet Application-Basics, Key characteristics	1
5.3	AJAX –History, Examples of usage of AJAX	1
5.4	Creating AJAX enabled application	2
5.5	RIA Platform overview (Flex,Silverlight)	1
5.6	Creating and customizing a simple User interface with Flex	2

5.7	Accessing XML data, Creating charts and graphs	1
	Total	**50**

Syllabus

Internet: History-Evolution of Internet, Distributed Information Systems-Design, Architecture, communication in an Information System, XHTML- headers, Linking, Images, Image map, meta elements, frameset, cascading style sheet. HTTP-Limitations of HTTP, Web servers-IIS & Apache, Application servers

Foundation Technologies: Scripting, Java script, control statement, function, objects. Server side programming, PHP-Basics, String processing and Regular Expressions, Connecting to a Database, Using cookies, Dynamic content, JSP-Java Web Technologies, creating and Running a simple application in Netbeans, JSF Components, Session Tracking.

XML : XML Structure, Elements, writing and Parsing XML Document, Organizing XML data, adding attributes, Namespaces, Working with DTD, Schemas, Elements, types, groups, defining attributes, XSL, XPath.

Web services (WS) and Database Connectivity: Need of WS, WS Standards, SOAP-Goals, Structure and contents of SOAP Message, Binding, WSDL-Goals, Structure, UDDI-Goals, Registry, Data Structures, Databases: RDBMS Model, SQL, MySQL.

Web 2.0 and Rich Internet Application: Web 2.0-Introduction,Technology overview, Integration with web services, Rich Internet Application-Basics, Key characteristics, AJAX- History, Examples of usage of AJAX, Creating AJAX-Enabled Application, RIA Platform overview(Flex,Silverlight), Creating and customizing a simple User interface, Accessing XML Data, creating charts and graphs.

Text Books:

1. Paul J.Deitel and Harvey M.Deitel, "AJAX, Rich Internet Applications, and Web Development for Programmers", Pearson Education, First Edition, 2009.

2. Deitel and Deitel, "Internet and World Wide Web How to Program", Prentice Hall of India, Fourth Edition, 2009

Reference Book:

1. Gustavo Alonso, Fabio Casati, Harumi Kuno and Vijay Machiraju, "Web Services" Springer International Edition. First Edition, 2009.

2. Heather Williamson, " XML:The Complete Reference", Tata McGraw Hill,2001

3. James Talbot, "Total Training for Adobe Flex 3: Rich Internet & AIR applications", Adobe Incorporations.

4. http://oreilly.com/web2/archive/what-is-web-20.html

Course Designers:

1. S.Padmavathi spmcse@tce.edu
2. K.Narasima Mallikarjunan arjunkumbaraj@tce.edu
3. S.Sridevi sridevi@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T 67	0	0	3	1

T67 Web Programming Lab 0:1

Preamble: Students learn how to choose their communication approach by considering platform, dynamically updating the web contents based on the client requirements. They will also learn how to transport data using XML and XML related technologies and protocols and also how to communicate with databases. This course emphasis the working in Rich Internet Application like Ajax, Flash and Flex and also standardization of XML Documents for the purpose of data exchange and integrate the communication mechanism also.

List of Experiments

1. Create a home page for your project using HTTP and HTML.
2. Format and Display the XML file using Java Script.
3. Create a web page to handle Events and Objects using Java Script.
4. Implement Client side and Server side form validation using Java Script.
5. Write a Servlet which communicates with Browser.
6. Present a XML document using Cascading style sheets(CSS).
7. Implement a program for DOM to process XML file.
8. Develop a java program to connect to database using JDBC drivers
   Write the same with PHP programming.
9. Create a relational Database and display the concurred data's from two tables in a web page.
10. Create a Ajax enabled form which gets input for online registration for paper presentation contest(should prompt for missing information in client side).
11. Perform object animation using flash action script.
12. Write a web program to display an flex application.
13. Create a flex chart.
14. Create an interactive application for server side application in flex.

COURSE DESIGNERS:

S.Padmavathi spmcse@tce.edu
K.Narasimha Mallikarjunan arjunkambaraj@tce.edu
S.Sridevi sridevi@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
T68	0	0	3	1

T68 Software Tools Lab 0:1

Preamble: This lab is a simple introduction to some of the concepts inherent to software tools. This includes hands on experience in analysis, design and testing an example software system like Banking, Insurance, healthcare etc., and followed by improving and enhancing open source software tool.

Prerequisite: Proficiency in java and Eclipse/Netbeans.

List of Experiments:

1. Preparation of SRS - Problem Analysis thorough study of the problem, Identify project scope, Objectives, infrastructure, Software Requirement Analysis, Describe the individual Phases/ modules of the project, Identify deliverables, define the problem to be solved providing domain-specific knowledge.
2. Practicing the different types of software tools such as (Rational Rose & other Open Source) used for all the phases of Software development life cycle.
3. Define functional requirement of the system by Usecase, Activity diagram and develop classes necessary for realizing functional requirements by Collaboration Diagram.
4. Define classes to implement the system, attributes and operations for each class and define relationships among the classes by Class Diagram.
5. Define interactions between objects using Sequence Diagram and define behavior of each object using State Chart.
6. Package the related classes into one component using Component Diagram and deployment of components using Deployment Diagram
7. Practicing the different types of testing and Defect Tracking tools such as (Junit, Codecover, Watij, Bugzilla, MuClipse & other Open Source).
8. Perform Requirements-Based Test Generation for the system.
9. Perform Code Coverage, Adequacy Criteria and Test Case Correlation for the system.
10. Perform Functional GUI Testing for the system.
11. Perform Mutation Testing and Analysis for the system.
12. Mini project - Perform Reengineering by improving and enhancing open source software application.

Course Designers:

1. S.Selvakumar ssit@tce.edu
2. P.Karthikeyan karthikit@tce.edu

REVISED CURRICULUM AND DETAILED SYLLABI

FOR

B.Tech DEGREE (Information Technology) PROGRAM

SEVENTH SEMESTER

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2012-2013

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Approved in BOS Meeting on 13.10.2012 Approved in 45th Academic Council Meeting on 24.11.2012

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015

B.Tech. Degree (Information Technology) Program

SUBJECTS OF STUDY

(For the candidates admitted from 2012-2013 onwards)

SEVENTH SEMESTER

Subject code	Name of the subject	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
T71	Management Theory and Practice	HSS	3	-	-	3
T72	Wireless and Mobile Communication	DC	3	1	-	4
TCX	Elective 3	DC	3	-	-	3
TCX	Elective 4	DC	3	-	-	3
XGX	Elective 5	ES	3	-	-	3
PRACTICAL						
T78	Project	DC	-	-	12	6
Total			**15**	**1**	**12**	**22**

BS : Basic Science
HSS : Humanities and Social Science
ES : Engineering Science
DC : Department Core

L : Lecture
T : Tutorial
P : Practical

Note:

1 Hour Lecture/Tutorial is equivalent to 1 credit
2/3 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015

B.Tech. Degree (Information Technology) Program

SCHEME OF EXAMINATIONS
(For the candidates admitted from 2012-2013 onwards)

SEVENTH SEMESTER

S.No	Sub. code	Name of the subject	Duration of Terminal Exam in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	T71	Management Theory and Practice	3	50	50	100	25	50
2	T72	Wireless and Mobile Communication	3	50	50	100	25	50
3	TCX	Elective 3	3	50	50	100	25	50
4	TCX	Elective 4	3	50	50	100	25	50
5	XGX	Elective 5	3	50	50	100	25	50
PRACTICAL								
6	T78	Project	-	150	150	300	75	150

* Continuous Assessment evaluation pattern will differ from subject to subject and for different tests. This will have to be declared in advance to students. The department will put a process in place to ensure that the actual test paper follow the declared pattern.

** Terminal Examination will be conducted for maximum marks of 100 and subsequently be reduced to 50 marks for the award of terminal examination marks.

Sub code	Lectures	Tutorial	Practical	Credit
T71	3	-	-	3

T71 Management Theory and Practice 3:0

Common to D61/ G71/ C61

Preamble

Management is the science of managing operations for an enterprise or organization. It deals with managing men, material, machinery and money. It has become an essential need to analyze the basic concepts of management theory and to understand the ways and means of implementing them in practice. The course work highlights the systematic approach for the management of various departments in an organization.

Program outcomes addressed

a. ability to identify, formulate and solve engineering problems
b. ability to function on multidisciplinary teams
c. ability to communicate effectively in both oral and written forms
d. ability to consider social, environmental, economic and ethical impact of engineering activities in a given context

Competencies

1. Manage the operations in total for an enterprise.
2. Work with team spirit and group coordination.
3. Ability to design Organizational Structure
4. To facilitate an effective communication both within and outside a firm.
5. Formulate the selection and recruitment procedures for a department
6. Evolve proper performance appraisal system
7. Analyze and identify an effective site selection and design a proper layout.
8. Prepare maintenance schedules for an organization.
9. Ability to measure overall productivity and suggest means to improve it
10. Plan the material handling systems for the organization.

Assessment Pattern

SNo	Blooms Category	Tet1	Test2	Test3
1	Remember	10	10	20
2	Understand	10	10	40
3	Apply	30	30	40
4	Analyze	-	-	-
5	Evaluate	-	-	-
6	Create	-	-	-

Course Level Learning Objectives

Remember

1. Define Management
2. What are the various functions of management?
3. Distinguish MBO and MBE.
4. Define Group Cohesiveness
5. What do you mean by semantic barrier of communication?
6. What type of industry requires process type layout?
7. Define Morale.
8. Mention the significance of Market Research?
9. Give an example of centralized layout.
10. Mention the types of maintenance
11. What do you understand by the term Productivity?

Understand

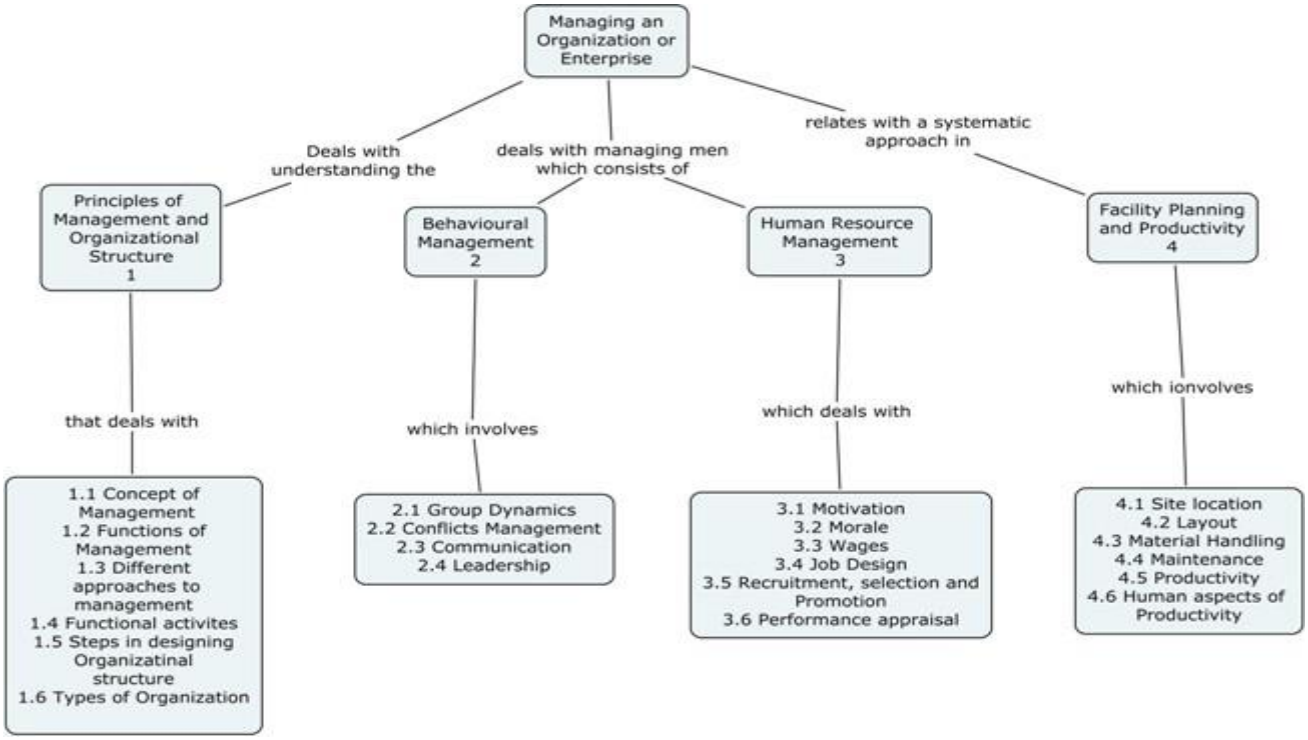
1. Briefly explain all the functions of Management
2. Explain various controlling techniques.
3. What are the merits of Modern Type of Organization?
4. List out all the stages of Group formation and explain
5. What are the barriers of communication?
6. What are the factors associated with morale?
7. Compare job enrichment and job enlargement
8. Compare the merits and demerits of product and process layouts
9. Enumerate all the human factors associated with productivity
10. Differenciate periodical and preventive maintenance
11. Enumerate all the ways of measuring productivity.
12. What are the requirements of an effective material handling system?

Apply

1. Bring out all the steps in the formation of a Quality Circle in an educational institution.
2. Suggest the modalities of selection of a trainee engineer to be recruited for a software firm
3. You as a manager prepare a proposal to locate a site to establish a telecommunication industry / Automobile Industry
4. Suggest all the possible ways to increase the overall productivity of a manufacturing sector
5. Prepare a preventive maintenance schedule for an electronic equipment manufacturing company which operates for three shits in 24 hours for 8

hours per shift by 6.00 AM to 2.00 PM, 2.00 PM to 10.00 PM and 10.00 PM to 6.00 AM

Concept map



Syllabus

Management and Functions of Management

Concept of management, organization, Administration-Management is a Science or an Art- Taylors Scientific Management – Henry Fayol's Principles of management -Functions of management- planning, Organizing, Staffing, Coordinating, Directing and Controlling-different approaches to management-various functional activities of different departments-Strategic planning-Management by Objectives [MBO], Management by Exception [MBE]-Organization Structure- Principles, Steps in designing an Organization, Types of Organization.

Behavioural Management

Group dynamics, types of groups, formation of group, Group cohesiveness, conflicts management- Communication –meaning and types, barriers in communication, communication in Groups- Leadership styles

Human Resources Management

Objectives-employer-employee relations-Motivation-Morale-Ways of achieving high morale-collective bargaining-Wage and wage payments-incentives-job design

,job analysis-job description, job rotation, job evaluation and merit rating-Recruitment, Selection and training of employees-Promotion-Performance appraisal

Facility Planning and Productivity

Site location-Factors to be considered-layout-objectives, types, factors influencing layout, layout procedure-Materials handling-principles, factors affecting the choice of materials handling, Materials handling equipment-Plant maintenance-need functions and types-Productivity-definition and concept, measurement-techniques for productivity measurement-Human aspects of productivity

Text Book

1. Harold Koontz, Heinz weihrich "Essentials of Manangement", Tata McGraw Hill ,6th Edition,2004
2. O.P. Khanna, Industrial Engineering and Management , Dhanpat Rai Publications,2006

References

1. Chase, Jacobs, aquilano, "Production and Operations Managemet " 8th Editin, Tata McGraw Hil Companies Inc 1999
2. Fred Luthans "Organizational Behaviour", Tata McGraw Hill, 2005
3. Edwin Flippo, "Personnel Management", Tata McGraw Hill,2004
4. R.N. Gupta, "Principles of Management",S.Chand and Co Ltd, 2008

Course Contents and Lecture Schedule

No.	Topic	No. of Lectures
1	**Priciples of Managmeent and Organizational Structure**	
1.1	Concept of management, Organization, Administration, Management is science or art, Taylor's Scientific Management, Henry Fayol's Priniciples of management	3
1.2	Functions of management, Planning, organizing, Staffing, Coordinating, Directing and controlling	3
1.3	Different approaches to management	1

No.	Topic	No. of Lectures
1.4	Functional activities, Strategic Planning, MBO, MBE	1
1.5	Principles and Steps Designing Organization structure	2
1.6	Types of Organization	1
2	**Behavioural Management**	
2.1	Group Dynamics, types of group, formation of group, group cohesiveness	3
2.2	Conflicts management	2
2.3	Communication, meaning and types, barriers in communication , communication in groups	3
2.4	Leadership styles	2
3	**Human Resource Management**	
3.1	Employer employee relations, Motivation	3
3.2	Morale, ways of achieving high morale, collective bargaining	1
3.3	Wages, wage and wage payments, incentives	1
3.4	Job Design, job analysis-job description, job rotation, job evaluation and merit rating	3
3.5	Recruitment, Selection and Promotion	3
3.6	Performance appraisal	1
4	**Facility Planning and Productivity**	
5.1	Site Location , factors to be considered	2
5.2	Layout objectives, types , factors influencing layout, layout procedure	2

No.	Topic	No. of Lectures
5.3	Material Handling, principles, factors affecting the choice of materials handling, materials handling equipments	2
5.4	Maintenance, need, functions and types	2
5.5	Productivity, definition and concept, measurement-techniques for productivity measurement	2
5.6	Human aspects of Productivity	2
Total		**45**

Course Designers

1. S.Krishnan deantlp@tce.edu
2. S.Muralidharan murali@tce.edu
3. R.Muruganandham rmmech@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credits
T72	3	1	0	4

T72 Wireless and Mobile Communication 3:1

Preamble:

The course aims at exploring the fundamental concepts of wireless communications in terms of OSI layer. It also explores the latest developments and technologies in the mobile communication systems.

Program Outcomes addressed:

Graduates will demonstrate

b. An ability to design and conduct experiments, as well as to analyze and interpret data

c. An ability to design a system, component or process as per needs and specifications.

d. An ability to identify, formulate and solve engineering problems.

e. An ability to use techniques, skills, and modern engineering tools to implement and organize engineering works under given constraints skills to use modern engineering tools, software and equipment to analyze problems.

Competencies

Students will be able to

1. Get a clear picture on the OSI layers of Wireless Communications.
2. Discern the wireless telecommunication systems including GSM.
3. Perceive the operation of Wireless LAN.
4. Determine the performance of a given wireless network
5. Design a wireless network for a given scenario
6. Describe the components of Cellular systems

Assessment Pattern:

S. No	Bloom's category	Test1	Test2	Test3	End Semester Examination
1	Remember	20	20	20	20
2	Understand	40	40	40	40
3	Apply	20	20	20	20
4	Analyze	20	20	20	20
5	Evaluate	0	0	0	0
6	Create	0	0	0	0

Course Level Learning Objectives

Remember

1. List the out the advantages of TDM over FDM.
2. Define Multiplexing.
3. Specify how Hidden terminal problem is solved by MAC.
4. How SDM is typically realized and combined with FDM?
5. What are WML decks and cards?
6. Which is the TDMA MAC scheme that provides lowest latency?

Understand

1. Describe how the collisions are avoided while transmission.
2. Mention the different types of telecommunication system Architectures.
3. Describe the steps involved Handover.
4. Compare and contrast the different MAC schemes.
5. Draw and explain the GSM System Architecture.
6. Categorize the GSM components based on its function and explain them.

Apply

1. Solve the Hidden and exposed and near far terminal problems by applying DAMA Scheme
2. Apply MACA scheme to solve Hidden and exposed terminal problem and show what will happen?
3. Can the problems using TCP be solved by replacing it with UDP? Where this could be useful?
4. Identify the requirements of Mobile IP and justify them. Does mobile IP fulfill them?
5. Compute the Cluster Head election process in CGSR in such a way the CH has longer life time
6. Construct a Routing table to ensure the loop free table driven operation in MANET

Analyze

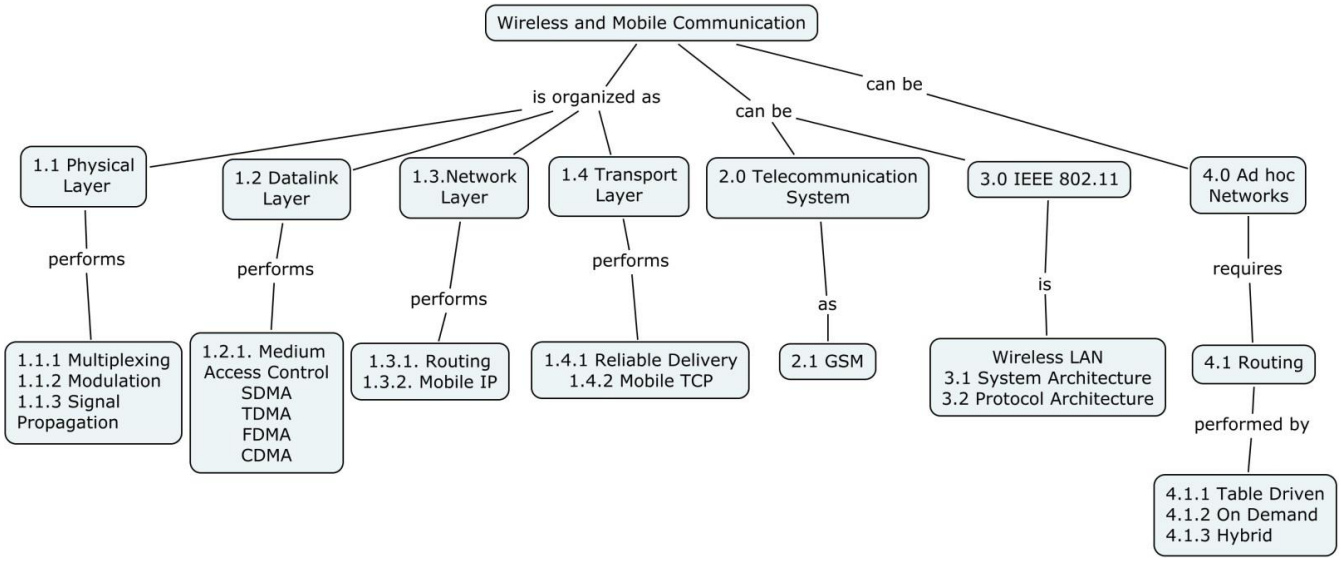
1. Show what will happen if a mobile node traveling under a tunnel for 30 minutes, find a solution to avoid the transmission errors for this situation.
2. What is the reaction of standard TCP in case of packet loss? In what situation does this reaction make sense and why is it quite often problematic in wireless environment
3. Show how the packet will be delivered to the destination node in case of its current address in foreign network

4. For the following situation design a new procedure for IP packet delivery to avoid the unnecessary transmission of data A & B nodes are belongs to same network, after sometime node B moves to foreign network and transfer the data to node A.

5. Show what are the problems encountered if Transport layer functions of wired Communication systems are used in wireless environment? And how this problem is addressed.

6. Examine the Differences in maintenance of topology information in various routing protocols.

Concept Map



Syllabus

Physical Layer: Multiplexing, Modulation, Signal propagation; **Datalink Layer**: MAC, SDMA, TDMA, FDMA, CDMA; **Network Layer**: Routing, Mobile IP; **Transport Layer**: Reliable delivery, Mobile TCP- **Telecommunication Systems** – GSM - **Wireless LAN** – IEEE 802.11 - System Architecture, Protocol Architecture - **Ad hoc Networks –** Introduction - Issues in Ad Hoc Wireless networks - Ad Hoc Wireless Internet - **Routing Protocols for Ad Hoc Networks –** Issues in Designing a routing protocol for Ad Hoc Wireless networks - Classification of routing protocols

Text Books

1. Jochen.H.Schiller, "Mobile Communications", Second edition, Pearson Education, 2003

2. C.Siva Ram Murthy, B.S.Manoj, Ad Hoc Wireless Networks Architectures and Protocols, Pearson Education, 2004

References

1. T.S.Rappaport, Wireless Communications : Principles and Practice, 2nd edition, Prentice Hall, 2002

2. William Stallings, Wireless Communications and Networks, 2nd edition, Prentice

Hall,2002

3. Advanced Wireless Communications and Internet: Future Evolving Technologies, Savo G.Glisic, 3rd edition, Wiley, 2011

Course contents and Lecture Schedule

Sl. No	Topics	No of Periods
1.1	**Physical Layer**	
1.1.1	Multiplexing	2
1.1.2	Modulation	3
1.1.3	Signal propagation	1
1.2	**Datalink Layer**	
1.2.1	MAC, SDMA, TDMA, FDMA, CDMA	5
1.3	**Network Layer**	
1.3.1	Routing	3
1.3.2	Mobile IP	3
1.4	**Transport Layer**	
1.4.1	Reliable delivery	3
1.4.2	Mobile TCP	3
2	**Telecommunication Systems**	
2.1	GSM	4
3	**Wireless LAN -** IEEE 802.11	
3.1	System Architecture	4
3.2	Protocol Architecture	4
4	**Ad hoc Networks**	
4.1	**Routing Protocols for Ad hoc Networks**	2
4.1.1	Table-driven Routing Protocols	3
4.1.2	On-demand Routing Protocols	3
4.1.3	Hybrid Routing Protocols	2
	Total Lectures	**45**

Course Designers:

1.S.Muthuramalingam - smrit@tce.edu

2.G.S.R.Emil Selvan - emil@tce.edu

BOARD OF STUDIES MEETING

B.Tech DEGREE (Information Technology) PROGRAM

Eighth Semester

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Approved in BOS Meeting on 30.04.2011 Approved in 42nd Academic Council Meeting on 11.06.2011

REVISED CURRICULUM AND DETAILED SYLLABI

FOR

B.Tech DEGREE (Information Technology) PROGRAM

EIGHTH SEMESTER

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2012-2013 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Approved in BOS Meeting on 30.04.2011 Approved in 42nd Academic Council Meeting on 11.06.2011

Department of Information Technology

Graduating Students of B.Tech program of IT will be able to

1. Explain and apply appropriate information technologies and employ appropriate methodologies to help an individual or organization achieve its goals and objectives.
2. Manage the information technology resources of an individual or organization.
3. Anticipate the changing direction of information technology and evaluate and communicate the likely utility of new technologies to an individual or organization.
4. Develop IT systems that would perform tasks related to E-governance and/or Health Care Management.
5. Work in a team using common tools and environments to achieve project objectives.

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015

B.Tech. Degree (Information Technology) Program

SUBJECTS OF STUDY
(For the candidates admitted from 2012-2013 onwards)

EIGHTH SEMESTER

Subject code	Name of the subject	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
TCX	Elective 6	DC	3	-	-	3
TCX	Elective 7	DC	3	-	-	3
TCX	Elective 8	DC	3	-	-	3
PRACTICAL						
T84	Project	DC	0	-	24	12
Total			**9**	**0**	**24**	**21**

BS : Basic Science
HSS : Humanities and Social Science
ES : Engineering Science
DC : Department Core
L : Lecture
T : Tutorial
P : Practical

Note:

1 Hour Lecture/Tutorial is equivalent to 1 credit
2/3 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015

B.Tech. Degree (Information Technology) Program

SCHEME OF EXAMINATIONS

(For the candidates admitted from 2012-2013 onwards)

EIGHTH SEMESTER

S. No	Sub. Code	Name of the subject	Duration of Terminal Exam. in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
THEORY								
1	TCX	Elective 6	3	50	50	100	25	50
2	TCX	Elective 7	3	50	50	100	25	50
3	TCX	Elective 8	3	50	50	100	25	50
PRACTICAL								
4	T84	Project	3	150	150	300	75	150

* Continuous Assessment evaluation pattern will differ from subject to subject and for different tests. This will have to be declared in advance to students. The department will put a process in place to ensure that the actual test paper follow the declared pattern.

** Terminal Examination will be conducted for maximum marks of 100 and subsequently be reduced to 50 marks for the award of terminal examination marks.

BOARD OF STUDIES MEETING

B.Tech Degree (Information Technology) Program

Departmental Electives

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

REVISED CURRICULUM AND DETAILED SYLLABI

FOR

B.Tech DEGREE (Information Technology) PROGRAM

DEPARTMENTAL ELECTIVES

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2008-2009 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Department of Information Technology

Graduating Students of B.Tech program of IT will be able to

1. Explain and apply appropriate information technologies and employ appropriate methodologies to help an individual or organization achieve its goals and objectives.
2. Manage the information technology resources of an individual or organization.
3. Anticipate the changing direction of information technology and evaluate and communicate the likely utility of new technologies to an individual or organization.
4. Develop IT systems that would perform tasks related to E-governance and/or Health Care Management.
5. Work in a team using common tools and environments to achieve project objectives.

Subject Code	Lecture	Tutorial	Practical	Credit
TCA	**3**	**--**	**--**	**3**

TCA Data warehousing 3:0

Preamble:

A data warehouse is a powerful database model that significantly enhances the user's ability to quickly analyze large, multidimensional data sets. It cleanses and organizes data to allow users to make business decisions based on facts.Data warehousing has emerged from being a mere passing fad to one of the most talked about technologies today. Enterprises have realized the importance of data warehousing solutions in making strategic business decisions, and have either implemented or are planning to implement data warehouses.This course provides a practical line for planning, building and managing a successful data warehouse.

Program outcomes addressed

Graduates will demonstrate:

a. the ability to design a system or component or process to meet stated specifications.

b. the ability to identify, formulate and solve engineering problems.

c. the ability to use techniques, skills, and modern engineering tools to implement and organize engineering works under given constraints.

Competencies

1. Understand the life cycle approach for a data warehouse project.
2. Choose the architectural components of the data warehouse.
3. Design Logical and physical model of the data warehouse.
4. Perform the data extraction, transformation, and loading (ETL) functions.
5. Deploy and maintain the data warehouse.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	End-semester examination
1	Remember	40	30	20
2	Understand	40	30	30
3	Apply	20	20	20
4	Analyze	0	10	10
5	Evaluate	0	0	0
6	Create	0	10	20

Learning Objectives

Remember

1. Name at least five characteristics or features of a data warehouse.
2. List the various data sources for the data warehouse.
3. What is the STAR schema?
4. What are aggregate fact tables?
5. Name four key issues to be considered while planning for a data warehouse.
6. What is meant by conforming dimensions?
7. Write the three major types of metadata in a data warehouse.
8. What is meant by slice- and- dice? Give an example.
9. What are multidimensional databases?
10. List the major activities during data warehouse deployment?

Understand

1. A data warehouse is an environment, not a product, Discuss.
2. What types of processing take place in a data warehouse? Describe.
3. How are the top-down and bottom-up approaches different for building a data warehouse? Discuss the merits and disadvantages of each approach.
4. How are data warehouse projects different from OLTP system projects?
5. Explain any four development phases in the life cycle of a data warehouse project.
6. What is your understanding of data warehouse architecture? Describe
7. Describe the major components of the physical infrastructure.
8. Why is ER Modelling technique not suitable for the data warehouse? How is dimensional modelling different?
9. Describe the types of activities that are part of the ETL Process. Which of these are time consuming?
10. Describe the various activities in Physical design.

Apply

1. You are the manager for the data warehouse project of a nationwide car rental company. Your data warehouse is expected to start out in the 500 GB range. Select the suitable server hardware and write a justification for that.
2. In a STAR schema to track the shipments for a distributions company, the following dimension tables are found: (1) time, (2) customer ship-to, (3) ship-from, (4) product, (5) type of deal, and (6) model of shipment. Review these dimensions and list the possible attributes for each of the dimension tables.

3. As the data warehouse administrator, performance enhancement is high on your list. Highlight the techniques you plan to adopt. For each technique, indicate tasks necessary to implement the technique.

4. Assume you are in the insurance business. Find two examples of Type 2 slowly changing dimensions in that business. Write the specifications for applying the Type 2 changes to the data warehouse with regard to the two examples.

5. You are the data design specialist on the data warehouse project team for a retail company. Apply STAR schema to track the sales units and sales dollars with three dimension tables. Explain how you will decide to select and build four two-way aggregates.

6. Apply a simple STAR schema with a factless fact table to track patients in a hospital by diagnostic procedures and time.

Analyze

1."In a data warehouse, business requirements of the users form the single and most powerful driving force". Do you agree? If you do, state four reasons. If not, is there any other such driving force?

2. Give three reasons why you think ETL functions are most challenging in a data warehouse environment.

3. Compare and Contrast MOLAP and ROLAP for an insurance company's data warehouse.

4. Analyze the reasons for backing up a data warehouse. How is this different from backing up in an OLTP system?

5. Do you think a multitier user support structure is suitable for the data warehouse environment? Analyze the alternates.

6. Analyze how physical model components are related to components of the logical model.

Create

1. You are the data design specialist on the data warehouse project team for a manufacturing company. Design a STAR schema to track the production quantities. Production quantities are normally analyzed along the business dimensions of product, time, parts used, production facility, and production run.

2. In your organization, assume that customer names and addresses are maintained in three customer files supporting three different source operational systems. Describe the possible entity identification problem you are likely to face when you consolidate the customer records from the three files. Write a procedure outlining how you propose to resolve the problem.

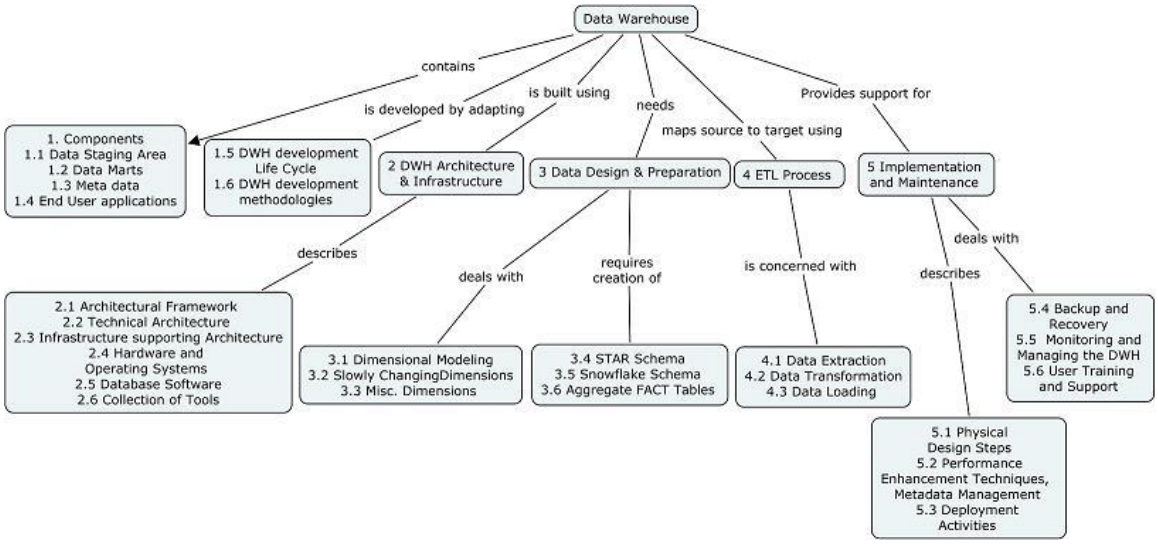
3. As the data warehouse administrator, prepare a backup and recovery plan. Indicate the backup methods and schedules. Explore the recovery options. Describe the scope of the backup function. How will you ensure the readiness to recover from disasters?

4. As the manager for the data warehouse project, write a project completion report to your CIO and the executive project sponsor. List the major activities completed. Mention the plan for staged deployment of future releases. Indicate the plans for ongoing maintenance. Briefly highlight each topic on growth and maintenance.

5. Your project team has decided to use the system logs for capturing the updates from the source operational systems. You have to extract data for the incremental loads from four operational systems all running on relational databases. These are four types of sales applications. You need data to update the sales data in the data warehouse. Make assumptions and describe the data extraction process.

6. As the data designer for an international bank, consider the possible types of snapshot and transaction tables. Complete the design with one set of snapshot and transaction tables.

Concept Map



Syllabus

Components of data warehouse: Data staging area, Data Marts, Metadata, End user applications, DW development life cycle, DW development

Methodologies **Data Warehouse Architecture and Infrastructure:** Understanding DWH Architecture, Architectural Framework, Technical Architecture, Infrastructure Supporting Architecture, Hardware and Operating Systems, Database Software, Collection of Tools **Data Design and Preparation:** Dimensional Modeling, Slowly Changing Dimensions, Miscellaneous Dimensions, STAR Schema, Snowflake Schema, Aggregate Fact Tables **ETL process:** Data Extraction, Data Transformation, Data Loading **Implementation and Maintenance of DWH:** Physical Design Steps, Performance Enhancement Techniques, Meta Data Management, Deployment Activities,. Backup and Recovery, Monitoring and managing the Data warehouse, User Training and Support

Text Books

1. Data warehousing fundamentals Ponnaiah, Paulraj. Wiley Publishers 2003.

2. The Data warehouse Life cycle tool kit, 2nd edition Kimball, Ralph. Wiley Publishers 2008.

References

1.Data Warehousing: Design, Development and best practices Mohanty, Soumendra. Tata McGraw Hill 2006.

2.Data warehousing in the real world Anahory, Sam / Murray, Dennis. Addison Wesley publishers 2003.

Websites

1.http://www.inmoncif.com

2.http://www.ralphkimball.com

3.http://www.datawarehousing.com

Course Contents and Lecture Schedule

No.	Topic	No. of Lectures
1	**Components of data warehouse**	
1.1	Data Staging Area	1
1.2	Data Marts	1
1.3	Metadata	1

1.4	End User Applications	1
1.5	DWH development life cycle	2
1.6	DWH development Methodologies	2
2	**Data Warehouse Architecture and Infrastructure**	
2.1	Architectural Framework	2
2.2	Technical Architecture	2
2.3	Infrastructure Supporting Architecture	1
2.4	Hardware and Operating Systems	1
2.5	Database Software	1
2.6	Collection of Tools	1
3	**Data Design and Preparation**	
3.1	Dimensional Modeling	2
3.2	Slowly Changing Dimensions	1
3.3	Miscellaneous Dimensions	1
3.4	STAR Schema	2
3.5	Snowflake Schema	1
3.6	Aggregate Fact Tables	1
4	**ETL Process**	
4.1	Data Extraction	3
4.2	Data Transformation	3
4.3	Data Loading	2

5	**Implementation and Maintenance of DWH**	
5.1	Physical Design Steps	2
5.2	Performance Enhancement Techniques, Meta Data Management	2
5.3	Deployment Activities	1
5.4	Backup and Recovery	1
5.5	Monitoring and managing the Data warehouse	1
5.6	User Training and Support	1

Course Designers:

1.M.Indra Devi midit@tce.edu

2.S.Appavu alias Balamurugan sbit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
TCB	3	0	-	3

TCB Design and Analysis of Algorithms 3:0

(Common to CSE/IT:C35)

Preamble: This subject will enable students to identify, formulate and solve real world engineering problems that require usage of algorithms.

Program Outcomes addressed

a. Graduates will demonstrate knowledge of mathematics, science and engineering.

b. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

k. Graduate who can participate and succeed in competitive examinations.

Competencies

1. Determine the time and space complexity of algorithms for sorting (insertion, selection, merge, quick sort and heap sort), dynamic programming, searching (binary search tree and red black tree) and graphs (directed and undirected) applied to average, worst and best cases.
2. Apply algorithms for shortest path, knapsack, divide and conquer, minimum spanning tree, and traveling salesman problems
3. Apply algorithms for NP hard and NP complete graph coloring problem.
4. Evaluate the appropriateness of algorithms for given problems.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	End-semester examination
1	Remember	20	10	0
2	Understand	30	20	20
3	Apply	40	40	50
4	Analyze	10	10	10
5	Evaluate	0	20	20
6	Create	0	0	0

Course Level Learning Objectives

Remember

1. Prove T(m) = 27 ($\lfloor n/2 \rfloor$ + [7])+ n is O(n lg n)
2. Prove the recurrence relationship T (n) = 2T (√n) +1 by making a change of variables. Your solution should be asymptotically tight. Do not worry about whether values are integral.

Understand

1. Describe the potential advantage of Notation Ω, θ, O
2. Establish the worst-case running time of heap sort is Ω (n log n)?
3. Verify the truth of the following statements
   a) $n^2 \in O(n^3)$
   b) $n^3 \in O(n^2)$
   c) $2^{n+1} \in O(2^n)$
4. Verify $t_n = \sum t_{n-1} + n + 2^n$ $n \geq 1$; $t_0=0$

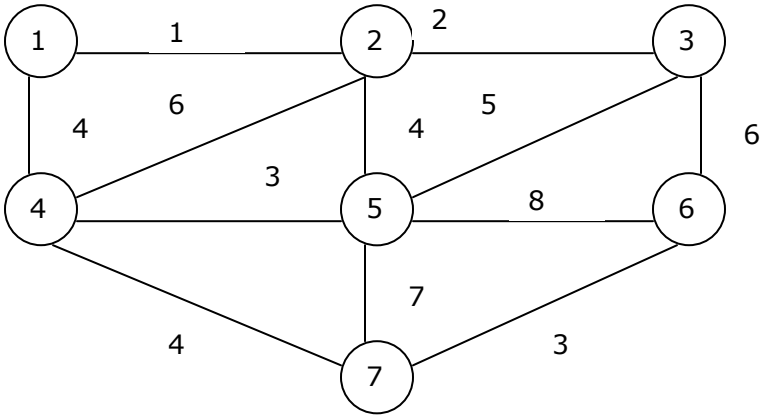
Apply

1. Determine the tight Asymptotic bounds for the following recurrence
   i. $T(n)= 4T(n/2)+n$
   ii. $T(n)=4T(n/2)+n^2$
   iii. $T(n)=4T(n/2)+n^3$
2. Apply Heap sort on the array A= {5, 13, 2, 23, 7, 17, 20, 8, 11}?
3. Determine the optimal parenthesization of a matrix-chain produce whose sequence of dimensions is { 5, 10, 3, 12, 5, 50, 6}
4. Apply the rules of recurrence to the following function
5. t: N+ -> R+

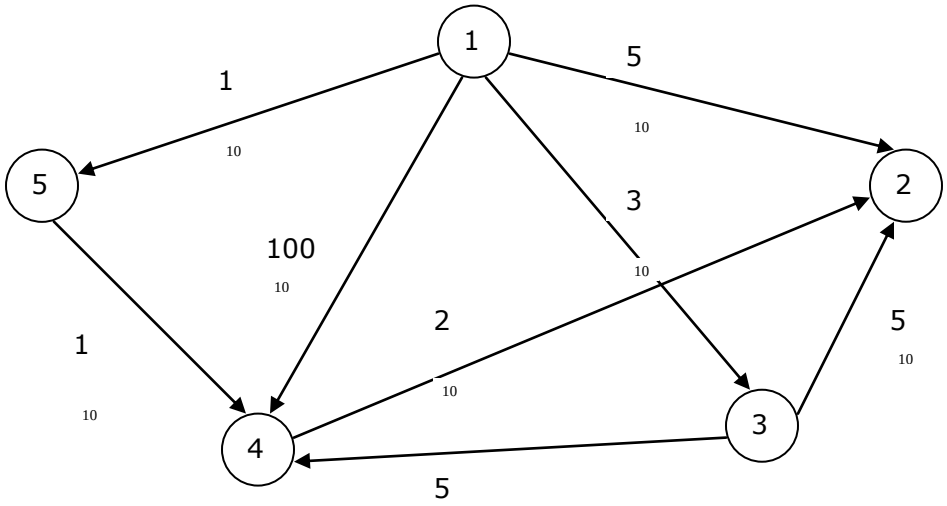
$$t(n) = \begin{cases} a & if n = 1 \\ bn^2 + nt(n-1) & otherwise \end{cases}$$

when a, b are arbitrary +ve constant

6. Apply the Kruskal's algorithm for following minimum spanning tree.



7. Apply the **djikstra** algorithm for the graph



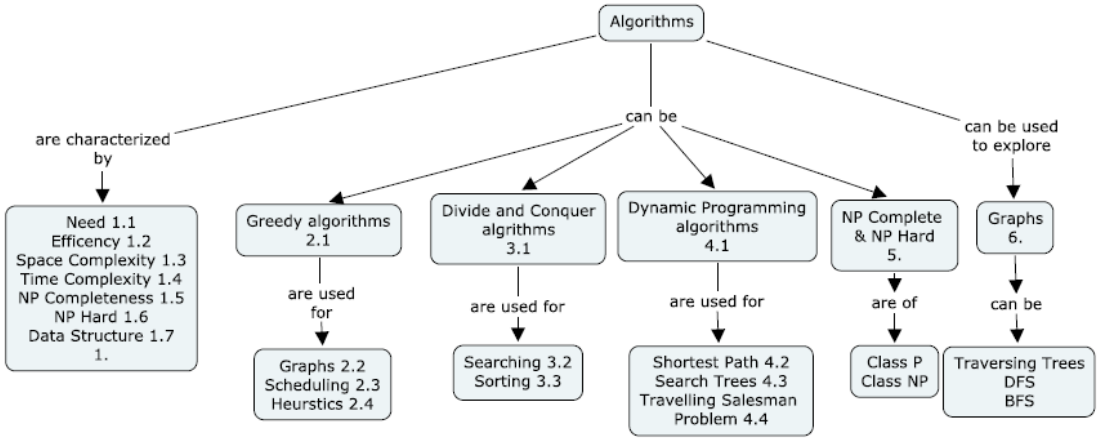
8. Calculate the product ABCD of four matrixes applying chained matrix multiplication

   A is 13 * 5 C is 89 * 3

   B is 5 * 89 D is 3 * 34

9. 15. Apply the reduction theorem **MT ≤' MQ**

Concept Map



Syllabus

Algorithms: Need for an Algorithm, efficiency of algorithms, space complexity, time complexity, NP Complete, NP Hard, data structures: solving recurrences using the characteristic equation. **Greedy Algorithms**: Greedy algorithms for scheduling – greedy heuristics **Divide and Conquer**: Determining the threshold, binary

searching, sorting by merging, quick sort, selection and the median, arithmetic with large integers, matrix multiplication **Dynamic Programming**: chained matrix multiplication, shortest paths, optimal search path, Traveling Salesperson problem, memory functions. **NP and NP Hard**: class P and NP, NP complete problems. **Graphs**: Traversing trees, depth-first search: undirected graphs and directed graphs, Breadth-First Search

Text book

1. Algorithmics Theory and Practice, Gilles Brassard and Paul Bratley, Printice hall international,Inc., 2000.

References

1. Introduction to Algorithms –T.H.Cormen , C.E.Leiseerson, R.L.Rivest second edition PHI .2002
2. Computer algorithms :Introduction to Design and Analysis –Sara Baase Addison wesley publication. 1998.

Course Contents and Lecture Schedule

Sl.no	Topic	Hours
1.	**Algorithms**	
1.1	Need for an Algorithm	1
1.2	Efficiency of Algorithm	1
1.3	Space Complexity	1
1.4	Time complexity (asymptotic notations)	1
1.5	NP completeness	1
1.6	NP Hard	1
1.7	Data structures: solving Recurrences	2
2.	**Greedy Algorithms**	
2.1	Greedy Algorithms and Graphs –Introduction	1
2.2	Graphs	1
2.2.1	Greedy Algorithms – MST	1

2.2.2	Shortest path	1
2.3	Scheduling – Scheduling with deadlines	2
2.4	Heuristics- Coloring a Graph – TSP	2
3.	**Divide and Conquer Approach**	
3.1	Introduction to divide and conquer approach	1
3.2	Binary searching Sorting by Merging	1
3.3	Quick sort	1
3.3.1	Heap sort	1
3.3.2	Merge sort	1
3.3.3	Selection sort	1
3.3.4	Selection and the median	1
3.3.5	Arithmetic with large integers -Matrix Multiplication	1
4.	**Dynamic programming**	
4.1	Chained Matrix Multiplication	1
4.2	Shortest Paths(optimal search path)	3
4.3	Traversing Trees	2
4.4	Traveling salesman problem and memory functions	2
5.	**NP and NP Hard**	
5.1	The class P and NP,NP complete problems	3
6.	Graphs	1
6.1	Graphs: Traversing Trees, Depth-First Search: Undirected Graphs, Directed Graphs, Breadth-First Search (Undirected and Directed)	4
	Total	40

Course Designers

1. C. Deisy cdcse@tce.edu
2. T. Manikandan tmcse@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
TCC	3	-	-	3

TCC Principles of Compiler Design 3:0

(Common to CSE/IT:C43)

Preamble: This course gives an overview of the various phases of compiler and explains how a high level program is converted to a machine language program.

Program Outcomes addressed

b. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

d. Graduates will demonstrate an ability to design a system, component or process as per needs and specifications.

e. Graduates will demonstrate an ability to visualize and work on laboratory and multidisciplinary task.

Competencies

1. Explain the role of each phases of a compiler with its construction tools.
2. Implement various parsers like top down, bottom up, operator precedence parsers.
3. Understand the importance of intermediate code generation and code optimization.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	End-semester examination
1	Remember	20	10	10
2	Understand	50	20	20
3	Apply	30	70	70
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	0	0	0

Course level Learning Objectives

Remember

1. List the criteria to the selection of optimizing transformations.
2. Define induction variable. Give examples.
3. Explain the format of activation record.

4. Define Compiler. What are the phases of the Compiler? Explain with a neat diagram.
5. What are Compiler Construction Tools? Explain its specifications in detail.
6. What are the error recovery actions in a lexical analyzer?
7. What are the algebraic properties of regular expressions?

Understand

1. Define the role of input buffer in lexical analysis.
2. Write the function of syntax directed translation.
3. Write three address code to the statement A=-B*(C+D).
4. What do you mean by code motion? Explain.
5. What is basic block? Explain the role of flow graph in basis blocks
6. How would you calculate the cost of an instruction?
7. How would you map names to values?
8. What does operator-precedence parser do to the input id + id * id.

Apply

1. Write regular expression to generate identifiers give examples.
2. Construct of a parse tree to the string-(id+id) using left most derivation.
3. Draw transition diagrams to floating point numbers.
4. Write procedure for the construction of NFA from a regular expression
5. Discuss neatly the language for specifying lexical analyzers.
6. Construct NFA, DFA for the expression aa* ¦ bb*
7. What is SLR parsing .construct SLR parsing table for the grammar.

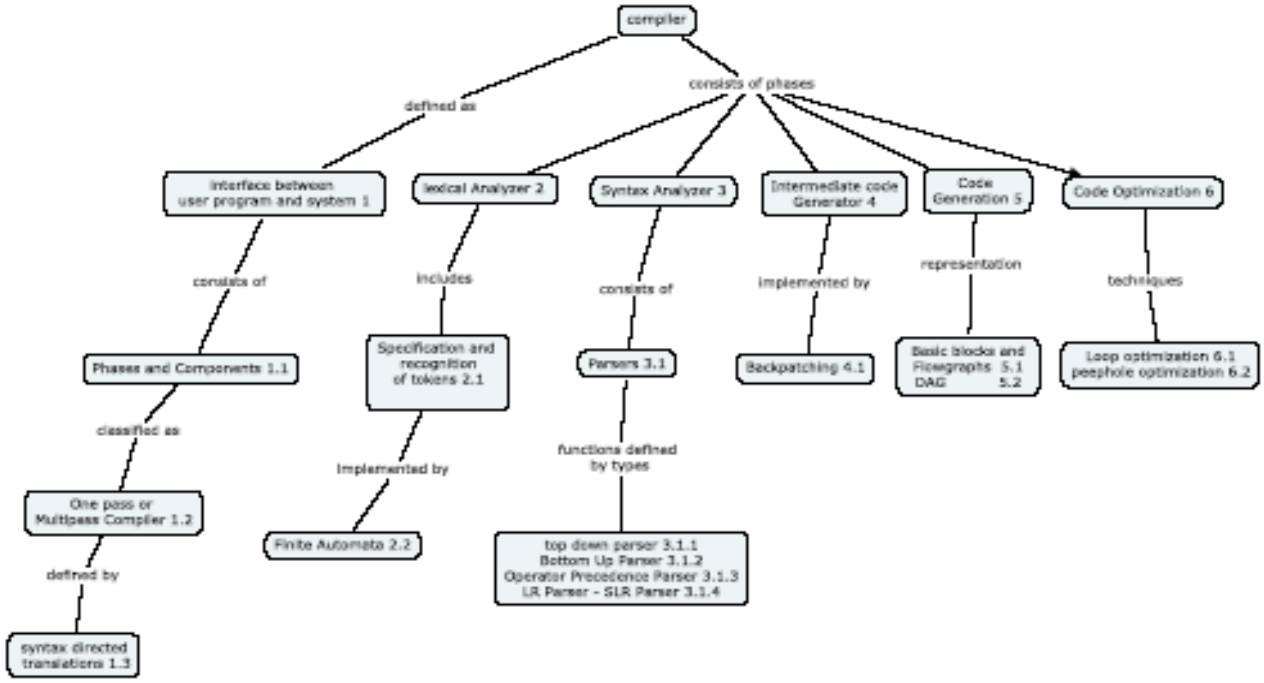
   E->E+T/T

   T->T*F/F

   F->(E)/id
8. Construct the predictive parser for the following grammar.
   S->a|↑|(T)
   T->T,S|S
   Write down the necessary algorithms and define FIRST and FOLLOW.
   Show the behaviour of the parser in the sentences:
   (i)(a,(a,a))
   (ii)(((a,a), ↑,(a),a).

9. Check whether the following grammar is a LL(1) grammar
   S->iEtS| iEtSeS' |a
   E->b

Concept Map



Syllabus

Introduction to compiling: Compilers, Analysis of the source program, Phases of a compiler, Cousins of a compiler, Grouping of Phases, Compiler – construction tools. **A Simple one pass compiler:** Overview, syntax definition, syntax-directed translation, Parsing, A translator for simple expressions. **Lexical Analysis:** The role of the lexical analyzer, Input buffering, specification of tokens , recognition of tokens ,A language for specifying lexical analyzers, Finite Automata. **Syntax Analysis :** The role of a parser, Context-free grammars, Writing a grammar, Top-down Parsing, Bottom-up parsing, Operator-precedence parsing , SLR parser. **Intermediate Code Generation:** Backpatching **Code Generation:** Issues in the design of a code generator, basic blocks and flow graphs, The DAG representation of basic blocks, Code generation from DAG's. **Introduction to Code Optimization:** The principal sources of optimization, Peephole Optimization, Loops in flow graphs.

Text Book:

1. Alfred V. Aho, Ravi Sethi, Jeffrey D Ullman – Compiler Principles, Techniques and Tools, Pearson Education, 2007.

References:

1. Steven S. Muchnick: Advanced Compiler Design & Implementation – Harcourt Asia, Morgan Kaufmann,2001.
2. J. P. Bennet: Introduction to Compiling Techniques, Tata McGraw-Hill Publishing-2002.

Course contents and Lecture schedule

S.No	Topics	No.of periods
1.	**Introduction to Compiling**	
1.1	Phases and components of a compiler	3
1.2	One pass compiler	3
1.3	Syntax-directed translation	2
2.	**Lexical Analysis**	
2.1	Specification and recognition of tokens	4
2.2	Finite Automata	4
3.	**Syntax Analysis**	
3.1	Parsers	1
3.1.1	Top-down Parsing	3
3.1.2	Bottom-up parsing	1
3.1.3	Operator-precedence parsing	2
3.1.4	LR - SLR parser	2
4.	**Intermediate Code Generation**	
4.1	Backpatching	6
5.	**Code Generation**	
5.1	Basic blocks and flow graphs	2

5.2	The DAG representation	3
6	**Introduction to Code Optimisation**	
6.1	Loop optimization	3
6.2	Peephole Optimisation	1
	Total	40

Course Designers

1. B. Thanga Parvathi btpcse@tce.edu
2. M.P. Ramkumar mpr@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
TCD	3	-	-	3

TCD: Basics of Digital Systems **3:0**

(Common to ECE/IT:D23)

Preamble: In designing this course the following assumptions are made.

- The course aims at design of combinational and sequential functions at gate level and simulate and verify their functionality using the Hardware description Language (Verilog)
- No reference will be made to any technology or logic family.
- No hardware aspects (voltages, currents, noise margins, transients etc.) other than the delay time would be considered in designing logic functions.
- As SSIs and MSIs are not in use any more, no reference needs to be made to these ICs.

Program Outcomes addressed:

a. An ability to apply knowledge of engineering, information technology, mathematics, and science.
b. An ability to design a system or component, or process to meet stated specifications.
c. An ability to use techniques, skills, and modern engineering tools to implement and organize engineering works under given constraints.

Competencies:

1. Determine the functional behavior and timing performance of a given combinational circuit.
2. Determine the functional behavior and timing performance of a given sequential circuit using state diagrams, timing diagrams, and PS-NS-O tables.
3. Determine the behavior of a given digital circuit with regard to hazards, asynchronous inputs, and output races.
4. Design digital circuits to perform specified combinational and sequential functions.
5. Describe, simulate and test combinational and sequential logic, and finite state machines in Verilog through behavioral, data flow and structural models.

Methods:

1. Truth-tables
2. Minimization of logic expressions through algebraic and algorithmic methods
3. Schematic diagrams for digital circuits
4. Timing diagrams
5. State diagrams
6. Present-state next-state output tables
7. Verilog description of digital circuits

Tools

1. Timing Tool
2. Verilog Simulator

Assessment Pattern

	Bloom's Category	Test 1	Test 2	End-semester examination
1	Remember	10	10	0
2	Understand	20	20	10
3	Apply	40	20	20
4	Analyze	30	20	20
5	Evaluate	0	10	10
6	Create	0	20	40

Course Level Learning Objectives

Understand

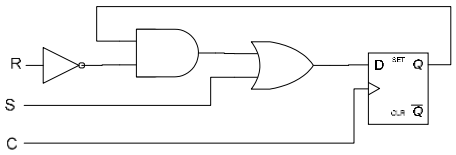
1. Differentiate between a truth table and an excitation table
2. Differentiate between a combinational circuit and a sequential circuit

Apply

1. Design a logic circuit for a children's math toy to teach number theory. A child enters two numbers, each from 0 to 20. The indicator light should be on if the two numbers are equal or if the first number is greater.
2. Design an elevator controller for a building that has four floors. There are four buttons in the elevator, one for each floor. As people enter the elevator, the buttons are being pushed. Design a circuit that determines which floor it should go next when the doors are closed.

Analyze

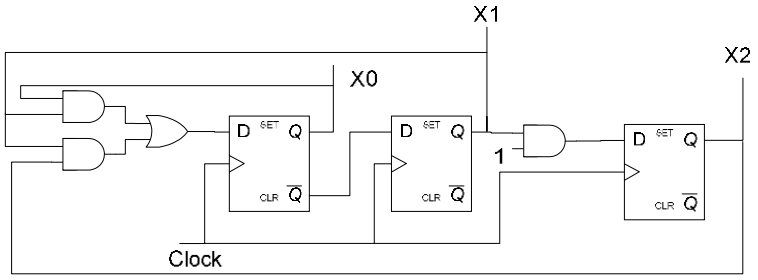
1. Analyze the circuit shown in the figure by constructing a timing diagram.



2. Explain why unused states generate don't-care terms when translating a state table to a transition table? Illustrate your response with a sample state table.

3. Explain why J-K flip-flops produce more don't-care terms than the other flip-flop types, even when all of the states are specified.

4. Analyze the circuit shown in the figure by obtaining its timing diagram. Start the timing diagram at state 1.



Create

1. Develop two different solutions for an ALU to detect whether the input word is a palindrome or not. A palindrome has bits reading from left to right being the same as the bits reading from right to left. Hence 1001 is a palindrome but 1011 is not. Another example is ABCCBA. To test for a palindrome for ABCCBA, you compare the most significant bit with the least significant bit, e.g. A with A. Then you compare the next most significant bit with the next least significant bit, e.g. (B with B), etc. Implement the design using Verilog.

2. An arithmetic logic unit, or *ALU*, is an essential component of computing systems, providing the combinational logic needed to implement commonly used arithmetic functions. In this exercise you will design a 32-bit (that is, the design will cater for numbers represented with 32-bits) ALU. The ALU is to provide options (i.e. *instructions*) for *ADD*, *SUBTRACT* and also bitwise operations for *AND* and *OR*. The number formats used are to be 2's complement.

   The ALU is to input:

• Two 32-bit numbers A and B.

• Two single bit lines used to indicate which operation is required.

It is to output:

• A single 32-bit result Y.

• Flags (i.e. single bits) for overflow, zero, and sign of Y. The overflow flag should indicate whether the last ADD or SUBTRACT produced a 2's complement overflow. The zero flag is true if Y is all zeros, and the sign flag indicates whether Y is positive (denoted by a zero) or negative (denoted by a one) in the 2's complement sense.

The input/output lines of the design *MUST* be named as follows:

• A<0:31> and B<0:31> where bit 31 is the least significant.

• Y<0:31> where bit 31 is again the least significant.

• OV, ZR and SN for the overflow, zero and sign bits.

• S0 and S1 for the operation select lines. These lines are coded as follows:

o Add A to B : S0 = 0, S1 = 0

o Subtract A from B : S0 = 0, S1 = 1

o Bitwise AND A and B : S0 = 1, S1 = 0

o Bitwise OR A and B : S0 = 1, S1 = 1

Determine the functional correctness of your design, by determining the maximum propagation delay of the circuit.

2. Design a Mealy sequential machine that will detect the following input sequences x = 01101 or 01111. If input sequence x=01101 is met, cause z1 = 1. If x = 01111, cause z2 = 1. Each input sequence may overlap with itself or other sequence. Document the whole procedure.

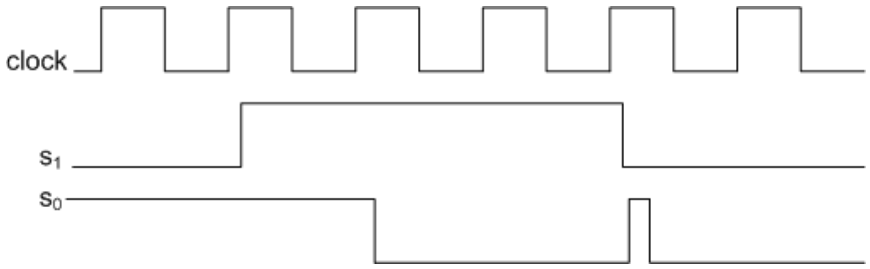
3. Design two different finite state machines that act as a 3 bit parity generator. For every 3 bits that are observed on the input 'w' during 3 consecutive cycles, the FSM generates the parity bit p=1 if and only if the number of 1's in the 3 bit sequence is odd. Verify using timing diagrams. (Use suitable assumptions)

4. A two bit counter has two control lines s1 s0. Its function is described in the table below. Two positive edge triggered flip flops A and B are used to implement this counter. A is the most significant bit of a count.

s1s0	Function
00	Unchanged
01	Count up
10	Count down
00	Clear (Synchronous)

a. Sketch the state diagram. Label each state with A(t)B(t) = 00,01,10,11 respectively.

b. Carefully draw the waveforms for A and B for the given initial values



c. Flip-flop A is a JK flip flop; B is a D flip flop. Derive input equations for the two flip flops.

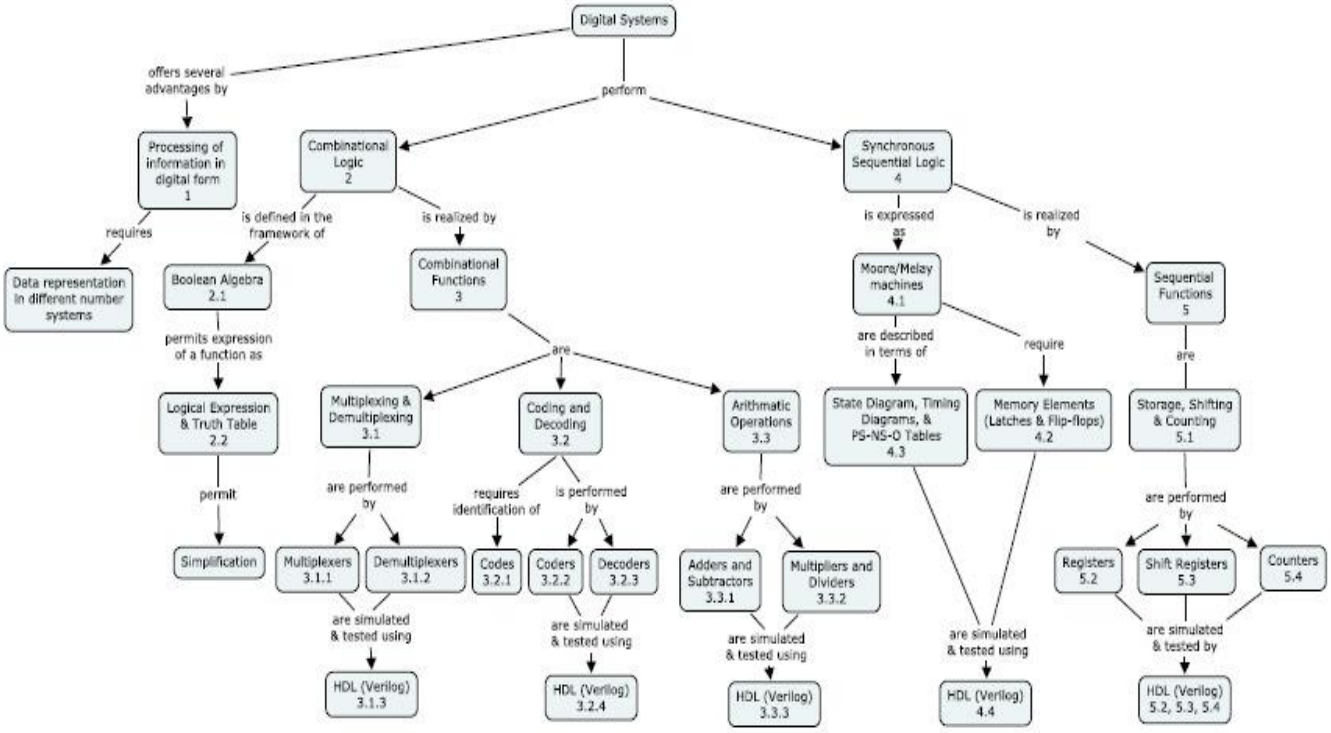
5. Design a special hardware comparator that would keep track of the maximum and minimum of a series of numbers. The numbers are presented to the hardware one at a time beginning with the count of numbers followed by the numbers themselves.

Evaluate

1. Design a synchronous circuit that has a single input variable and single output variable. The input data are received serially. The output Z is to change only when three consecutive input bits have the same value. Decide whether a Moore or a Mealy performs better.

2. Design a serial bit pattern detector that will detect the input sequence 01010 in a longer bit string. If the pattern is detected, then cause output Q to be active high. If a 011 bit pattern occurs within the same serial data string, cause output P to be active high. If the 011 pattern occurs, cause the state machine to initialize and start over looking for the 01010 pattern. Overlapping sequences can occur. Compare the performance of the circuit for a Moore and Mealy configurations.

Concept Map



Syllabus

Digital systems process information digital form offering many advantages. Combination Logic: Boolean Algebra, Logic Expressions and Truth Tables, Logic Minimization. Combinational Functions: Multiplexing and Demultiplexing, Verilog description of Multiplexers and Demultiplexers; Encoding and Decoding: Codes and Verilog description of Encoders and Decoders; Arithmetic Operations: Adders and Subtractors, Multipliers and Dividers, and Verilog Description of Arithmetic Operators. Synchronous Sequential Logic: Moore and Melay Machines, Latches and flip-Flops, State Diagrams, Timing Diagrams and PSNSO Tables, Verilogdescription of Synchronous Sequential Logic. Sequential Functions: Storing, Shifting and Counting.

Text Book:

1. Morris Mano: Digital Design, Third Edition, Prentice Hall, 2001

Course contents and Lecture schedule

No.	Topic	No. of Lectures
1	**Digital Information Processing**	
1.1	Advantages of processing information in digital form	1
1.2	Number systems	1
2.	**Combinational Logic**	
2.1	Boolean Algebra	2
2.2	Logic Expressions and Truth Tables	2
2.3	Logic Minimization	2
3.	**Combinational Functions**	
3.1	**Multiplexing and Demultiplexing**	
3.1.1	Multiplexers	0.5
3.1.2	Demultiplexers	0.5
3.1.3	Introduction to Verilog and Verilog description of Multiplexers and Demultiplexers	3
3.2	**Encoding and Decoding**	
3.2.1	Codes	1
3.2.2	Encoders	1.5
3.2.3	Decoders	1.5
3.2.4	Verilog description of Encoders and Decoders	1
3.3	Arithmetic Operations	
3.3.1	Adders and Subtractors	2
3.3.2	Multipliers and Dividers	2
3.3.3	Verilog Description of Arithmetic Operators	2
4.	Synchronous Sequential Logic	

4.1	Moore and Melay Machines	1
4.2	Latches and Flip-Flops	2
4.3	State Diagrams, Timing Diagrams and PSNSO Tables	3
4.4	Verilog description of Synchronous Sequential Logic	2
5.	Sequential Function	
5.1	Storing, Shifting and Counting	1
5.2	Registers and their Verilog description	2
5.3	Shift Registers and their Verilog description	3
5.4	Counters and their Verilog description	3

Course Designers

1. R. Helen rheee@tce.edu
2. P.G.S. Velmurugan pgsvels@tce.edu
3. V.R. Venkata Subramani venthiru@tce.edu
4. K. Hariharan harii74@tce.edu
5. M. Saravanan mseee@tce.edu

BOARD OF STUDIES MEETING

B.TECH DEGREE (INFORMATION TECHNOLOGY) PROGRAM

DEPARTMENTAL ELECTIVES

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

REVISED CURRICULUM AND DETAILED SYLLABI

FOR

B.Tech DEGREE (Information Technology) PROGRAM

DEPARTMENTAL ELECTIVES

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2008-2009 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Department of Information Technology

Graduating Students of B.Tech program of IT will be able to

1. Explain and apply appropriate information technologies and employ appropriate methodologies to help an individual or organization achieve its goals and objectives.
2. Manage the information technology resources of an individual or organization.
3. Anticipate the changing direction of information technology and evaluate and communicate the likely utility of new technologies to an individual or organization.
4. Develop IT systems that would perform tasks related to E-governance and/or Health Care Management.
5. Work in a team using common tools and environments to achieve project objectives.

Sub Code	Lectures	Tutorial	Practical	Credit
TCE	3		-	3

TCE MULTICORE ARCHITECTURES

(common to CSE/IT C53) **3:0**

Preamble: The era of Moore's law induced performance gains of sequential programs is over. In the future, the primary method of increasing program performance will require the utilization of multicore parallelism. The processing platforms of the future will have hundreds or even thousands of processor cores that are tightly integrated for parallel processing. It is necessary to train the programmers of the future with the right concepts and tools so that they can harness the computing power of multicores and massively parallel systems. Hence, a course on Multicore architectures is an essential part of any computer-science education. The course will facilitate the student to understand the design trade-offs or performance of multiprocessors by knowing the interaction of applications and architecture; defining and evaluating the traditional processors and emerging multicore processors; developing parallel programs using the programming models such as shared address space and message passing; exploiting the potential processing power of multicore processors and writing efficient parallel programs; understanding the issues in the organization of shared memory multiprocessors from the system architect's perspective .

Program Outcomes addressed

1. An ability to engage in life-long learning.
2. An ability to identify, formulate and solve engineering problems.

Competencies

1. Explain what are multi-core architectures
2. Describe the issues involved in writing code for multi-core architectures
3. Convert a given sequential program into a parallel one.
4. Determine the state transitions for a given sequence of memory 'reads' and 'writes' using MSI and MESI protocol.
5. Write parallel programs using OpenMP and message passing libraries.
6. Determine the performance of a given parallel program using the performance metrics

Assessment Pattern

	Bloom's Category	Test 1	Test 2	End-semester examination
1	Remember	30	20	20
2	Understand	30	30	20
3	Apply	40	50	60
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	0	0	0

Course Level Learning Objectives:

Remember

1. What is a Multicore processor?
2. What is a shared memory multiprocessor?
3. What is cache coherence?
4. List out the different parallel programming models.
5. What are the work sharing constructs in OpenMP
6. Define the performance analysis metrics for a parallel architectures
7. Define sequential consistency.

Understand

1. What are the naming issues in the shared address and message-passing model.
2. Explain the five components of parallel execution times.
3. Compare bit level, instruction level and thread level parallelism
4. List three examples which will justify the need for memory consistency models.
5. In what way, the usage of reduction clause in openMP differs from the usage of Critical in OpenMP
6. Give the pros and cons of both message passing and shared memory programming approaches from the point of view of ease of parallel program development and performance

Apply

1. Use OpenMP directives for parallelization; write a program in C that modifies the data elements of a N X N matrix.
2. SMPs keep the private caches of the processors coherent. However, each processor in an SMP has its own private set of registers. Are they kept coherent?

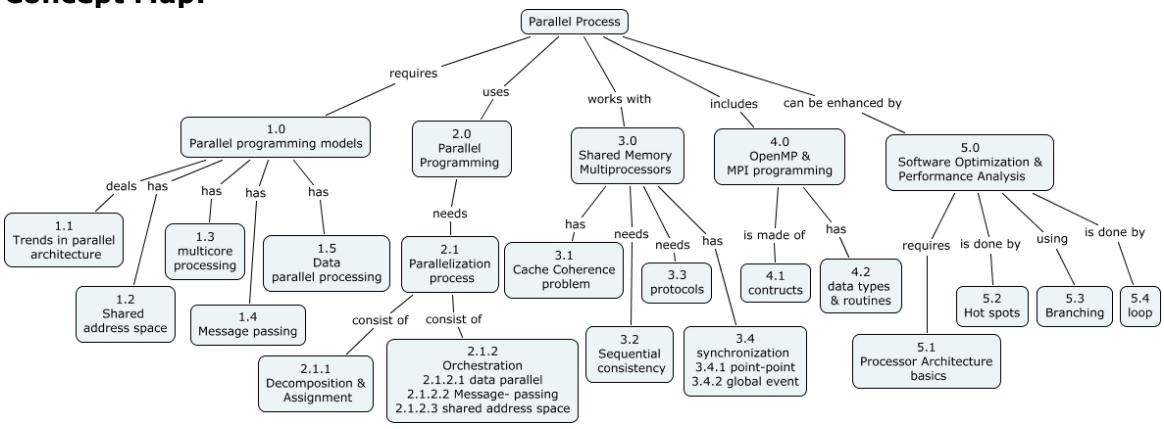
If not, why not? If they are not kept coherent, how is the desired semantics of a multithreaded program that uses a threading package such as pthreads guaranteed to work correctly in an SMP?

3. Consider the performance of a program that exploits the "blocked" approach to multithreading to hide memory latency, where the average miss latency is 100 cycles, the average run length is 15 cycles, and the average context switch latency is 5 cycles. What is the expected processor efficiency with two threads per processor?
4. Give a solution to the cache coherence problem after explaining its nature?
5. Write a parallel program for odd-even sort and find out the performance of the program using the different performance metrics.
6. Consider the following implementation of BARRIER synchronization primitive,used at the end of each phase of the computation.Assume that bar.releasing and bar.count are initially zero and bar.lock is initially unlocked.

```
Struct bar_struct
{LOCKDEC(lock);
Int count,releasing}bar;
…….
BARRIER(N)
{LOCK(bar.lock);bar.count++;
If(bar.count ==N){
Bar.releasing=1;
Bar.count--;
{else{
UNLOCK(bar.lock);
While(!bar.releasing);
LOCK(bar.lock);bar.count--;
if (bar.count==0){
bar.releasing=0;
}
{UNLOCK(bar.lock);}
```

This code fails to provide correct barrier.Find out the problem with this implementation and correct it.

Concept Map:



Course contents and Lecture Schedule:

	Lecture Topics	**Lecture Periods**
1.0	**Parallel programming models**	
1.1	Introduction to trends in parallel architecture	2
1.2	Shared address space	2
1.3	Introduction to multicore processing	1
1.4	Message passing	2
1.5	Data parallel processing	1
2.0	**Parallel Programming**	
2.1	The parallelization process, Parallelizing computation versus data	3
2.1.1	Decomposition and Assignment	1
2.1.2	Orchestration	2
2.1.2.1	Orchestration under data parallel model	2
2.1.2.2	Orchestration under shared address space model	2
2.1.2.3	Orchestration under Message-passing model	2
3.0	**Shared Memory Multiprocessors**	
3.1	Cache Coherence problem, bus snooping	2
3.2	Sequential consistency	1
3.3	Protocols- MSI and MESI	4
3.4	synchronization event	1

3.4.1	Point-to-point synchronization	1
3.4.2	Global event synchronization	1
4.0	**OpenMP and MPI programming**	
4.1	Constructs –Parallel, worksharing and Synchronization constructs	4
4.2	MPI data types and tags , environment management routines	4
4.3	Programming exercise	2
5.0	**Software Optimization and Performance Analysis**	
5.1	Processor Architecture basics and need for optimization	4
5.2	Hot spots	2
5.3	Branching	2
5.4	Loops	2
	TOTAL	50

Syllabus

Parallel programming models- Introduction to trends in parallel architecture, Shared address space, Introduction to multicore processing, Message passing, Data parallel processing. **Parallel Programming -**The parallelization process, Parallelizing computation versus data**,** Decomposition and Assignment, Orchestration- data parallel model, shared address space model, Message-passing model**. Shared Memory Multiprocessors -** Cache Coherence problem, bus snooping**.** Sequential consistency. Protocols- MSI and MESI. synchronization event- Point-to-point synchronization, Global event synchronization. **OpenMP and MPI programming -** Constructs –Parallel, worksharing and Synchronization constructs. MPI data types and tags, environment management routines. **Software Optimization and Performance Analysis-** Processor Architecture basics and need for optimization**,** Hotspots**,** Branching, Loops

Text Book:

1. David E. Culler,Jaswinder Pal Singh, Anoop Gupta : Parallel Computer Architecture: Hardware /software approach, Elsevier Science (reprint Technical Science & Engineering) , 2008
2. Richard Gerber, Aart J.C. Bik, Kevin B. Smith, et Xinmin Tian : Software Optimization Cookbook, Intel Press, 2006

Reference:

1. Maurice Herlihy, Nir Shavit :The Art of Multiprocessor Programming, Morgan Kauffman publishers, 2008
2. Rohit Chandra, Ramesh Menon , Leo Dagum, David Kohr , Dror Maydan , Jeff McDonald: Parallel Programming in OpenMP, Morgan Kauffman publishers, 2001

Course Designers:

1. P.Chitra pccse@tce.edu
2. S.Sudha ssj@tce.edu

Subject Code	Lecture	Tutorial	Practical	Credit
TCF	3	--	--	3

TCF Data Mining 3:0

Preamble:

Data Mining algorithms and computational paradigms allow computers to find patterns and regularities in databases, perform prediction and forecasting, and generally improve their performance through interaction with data. It is currently regarded as the key element of a more general process called Knowledge Discovery that deals with extracting useful knowledge from raw data. The knowledge discovery process includes data selection, cleaning, coding, using different statistical and machine learning techniques, and visualization of the generated structures. This course will cover all these issues and will illustrate the whole process by examples. Special emphasis will be given to the Machine Learning methods as they provide the real knowledge discovery tools.

Program outcomes addressed

Graduates will demonstrate the ability

a. to apply the knowledge of engineering, information technology, mathematics and science

b. to design a system or component or process to meet stated specifications.

c. to identify, formulate and solve engineering problems.

d. to use techniques, skills, and modern engineering tools to implement and organize engineering works under given constraints

Competencies

1. Understand the various steps under knowledge discovery process.
2. Perform data preprocessing on datasets.
3. Perform attribute selection on datasets.
4. Perform association rule mining on datasets.
5. Perform classification on datasets.
6. Perform clustering on datasets.
7. Evaluate the performance of different data mining algorithms.
8. Understand Web mining and Text mining.
9. Familiarize with data mining tools for solving practical problems.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	End-semester examination
1	Remember	40	20	20
2	Understand	40	30	20
3	Apply	20	40	50
4	Analyze	0	10	10
5	Evaluation	0	0	0
6	Create	0	0	0

Course Level Learning Objectives

Remember

1. What is data mining?
2. List the various data mining techniques.
3. What is the need for data cleaning?
4. What are the techniques used for data transformation?
5. What is correlation analysis?
6. What are the advantages of decision trees over other classification algorithms?
7. What are lazy learning methods?
8. What are the applications of clustering methods?
9. What is web mining?
10. What is text mining?

Understand

1. Differentiate classification from prediction.
2. Differentiate clustering from classification.
3. Distinguish between mining of structured data and unstructured data.
4. Discuss the working principle of various clustering algorithms.
5. Describe the ID3 algorithm of the decision tree construction.
6. Describe different pruning strategies.
7. Discuss the usefulness of data mining in e-commerce.
8. How is sequence mining different from mining of data in DBMS?
9. How SVM is different from MLP based classification?
10. How is web usage mining different from web structure mining and web content mining?

Apply

1. Apply association rule mining to the dataset of your own.

2. Apply the decision tree algorithms to the following example.

MOTOR	WHEELS	DOORS	SIZE	TYPE	CLASS
NO	2	0	Small	Cycle	bicycle
NO	3	0	Small	Cycle	tricycle
YES	2	0	Small	Cycle	motorcycle
YES	4	2	Small	automobile	Sports car
YES	4	3	medium	automobile	Mini van
YES	4	4	medium	automobile	sedan
YES	4	4	large	automobile	sumo

3. Apply NB algorithm on weather dataset. Weather dataset represents the training database that gives information of whether or not to play golf, given a set of climatic conditions.

4. Six observations on two variables are available, as shown in the following table.

Obs	X1	X2
a	3	2
b	4	1
c	2	5
d	5	2
e	1	6
f	4	2

Apply the k-means method, assuming that the observations belong to two groups and that one of these groups consists of a and e.

5. The 'database' below has four transactions. What association rules can be found in this set, if the minimum support is 60% and the minimum confidence is 80% ?

Trans_id	Itemlist
T1	{K, A, D, B}
T2	{D, A C, E, B}
T3	{C, A, B, E}
T4	{B, A, D}

6. Consider the following example. Apply DT and NB classification algorithms and report the performance of each algorithm.

MOTOR	WHEELS	DOORS	SIZE	TYPE	CLASS
NO	2	0	Small	Cycle	bicycle

NO	3	0	Small	Cycle	tricycle
YES	2	0	Small	Cycle	motorcycle
YES	4	2	Small	automobile	Sports car
YES	4	3	medium	automobile	Mini van
YES	4	4	medium	automobile	sedan
YES	4	4	large	automobile	sumo

Analyze

1. Identify a problem *from your own experience* that you think would be amenable to data mining. Describe:

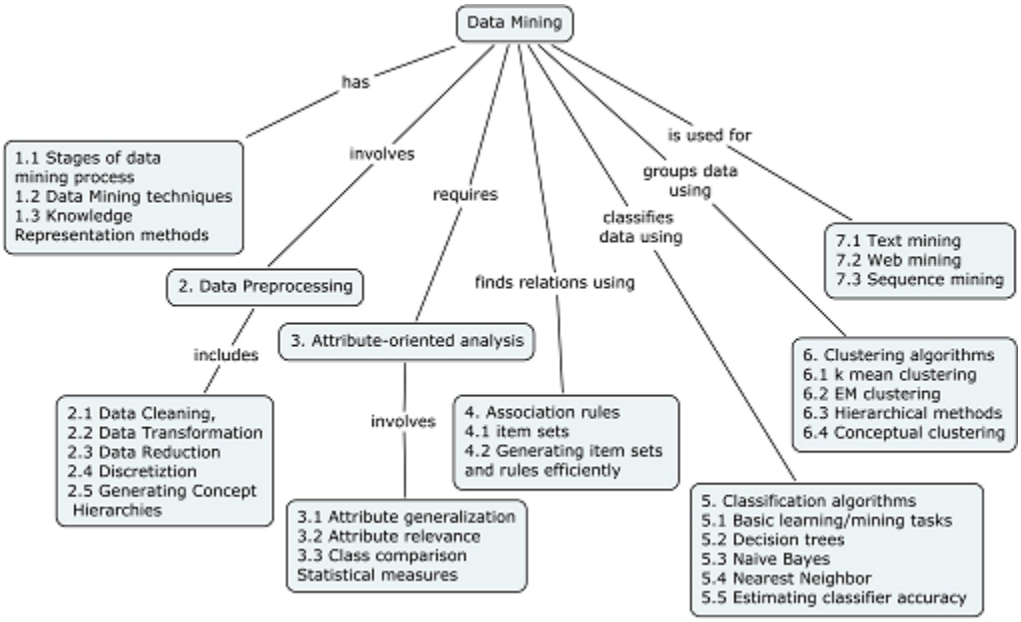
- What the data is?
- What type of benefit you might hope to get from data mining?
- What type of data mining (classification, clustering, etc.) you think would be relevant?
- Name one type of data mining that you think would *not* be relevant, and describe briefly why not.

2. Use the NB and J48 learning schemes, with the default settings to analyze the weather data. Use training set as the test option.

- Which of these classifiers are you more likely to trust when determining whether to play? Why?
- Report model percent error rate.

3. The apriori algorithm for generating association rules has many command line options. How do you modify these? What do the options mean? Can you modify the options?

Concept Map



Syllabus

Introduction to Data Mining -Data mining, Data Mining Goals, Stages of the data mining Process, Data Mining Techniques, Knowledge Representation Methods. **Data preprocessing** - Data cleaning, Data transformation, Data reduction, Discretization and generating concept hierarchies. **Attribute-oriented analysis**- Attribute generalization, Attribute relevance, Class comparison, Statistical measures. **Association rules** -Motivation and terminology, mining Transactional data, item sets, Generating item sets and rules efficiently. **Classification**- Basic learning/mining tasks, Decision trees, Naive Bayes, Nearest Neighbor, Estimating classifier accuracy (holdout, cross-validation, leave-one-out). **Clustering**-Basic issues in clustering, k means clustering, Expectation maximization (EM) clustering, Hierarchical methods: distance-based agglomerative and divisible clustering, Conceptual clustering: Cobweb. **Advanced techniques,** Text mining, Web mining, Sequence mining, WEKA PRACTICE, Case study on Medical data mining.

Textbook

1. Jiawei Han and Micheline Kamber, Data Mining: Concepts and Techniques,Morgan Kaufmann Publishers,2nd edition,2006.

Reference Books

1. Ian H. Witten and Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques (Second Edition), Morgan Kaufmann,ISBN: 0-12-088407-0,2005.

2. Arun K.Pujari, "Data Mining Techniques", University Press,ISBN – 81-7371-380-4,2005.

3. Dunham, M., Data Mining: Introductory and Advanced Topics. New Jersea: Pearson Education, Inc, 2003.

4. Hand, D., Mannila, H. and Smyth, P., Principles of Data Mining. Cambridge, Mass.: MIT Press, 2001.

Lecture Schedule

No.	Topic	No. of Lectures
1	**Introduction to Data Mining**	
1.1	Data mining, Data Mining Goals, Stages of the data mining Process	2
1.2	Data Mining Techniques	2
1.3	Knowledge Representation Methods	2
2	**Data preprocessing**	
2.1	Data cleaning	1
2.2	Data transformation	1
2.3	Data reduction	1
2.4	Discretization	2
2.5	Generating concept hierarchies	2
3	**Attribute-oriented analysis**	
3.1	Attribute generalization	1
3.2	Attribute relevance	2
3.3	Class comparison, Statistical measures	2

4	**Association rules**	
4.1	Motivation and terminology, mining Transactional data	2
4.2	Item sets, Generating item sets and rules efficiently	2
5	**Classification**	
5.1	Basic learning, mining tasks	1
5.2	Decision trees	2
5.3	Naive Bayes	2
5.4	Nearest Neighbor	2
5.5	Estimating classifier accuracy (holdout, cross-validation, leave-one-out)	2
6	**Clustering**	
6.1	Basic issues in clustering- k means clustering	2
6.2	Expectation maximization (EM) clustering	2
6.3	Hierarchical methods: distance-based agglomerative and divisible clustering	2
6.4	Conceptual clustering: Cobweb	2
7	**Advanced techniques**	
7.1	Text mining	2
7.2	Web mining	2
7.3	Sequence mining	2
7.4	WEKA PRACTICE	3
7.5	Case study on Medical data mining	2

Course Designers:

1. **Dr. M.Indra Devi midit@tce.edu**
2. **Dr. S.Appavu alias Balamurugan sbit@tce.edu**

Sub Code	Lectures	Tutorial	Practical	Credits
TCG	3	0	0	3

TCG Wireless and Mobile Computing

Preamble:

The course aims at exploring the fundamental concepts of wireless communications in terms of OSI layer. It also explores the latest developments and technologies in the mobile communication systems. An overview of WAP and toolkit enables the students to develop mobile applications.

Program Outcomes addressed:

a. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.
b. Graduates will demonstrate an ability to design a system, component or process as per needs and specifications.
c. Graduate will demonstrate skills to use modern engineering tools, software's and equipment to analyze problems.

Competencies: The students are able to

1. Understand the OSI layers of Wireless Communications.
2. Understand wireless telecommunication systems including GSM and 3G.
3. Understand the operation of Wireless LAN.
4. Develop mobile applications using J2ME toolkit.

Assessment Pattern:

Sl. No	Bloom's category	Test1	Test2	Test 3/End Semester Examination
1	Remember	50	30	20
2	Understand	50	70	80
3	Apply	0	0	0
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	0	0	0

Assignment: 10%

Course Level Learning Objectives:

Remember

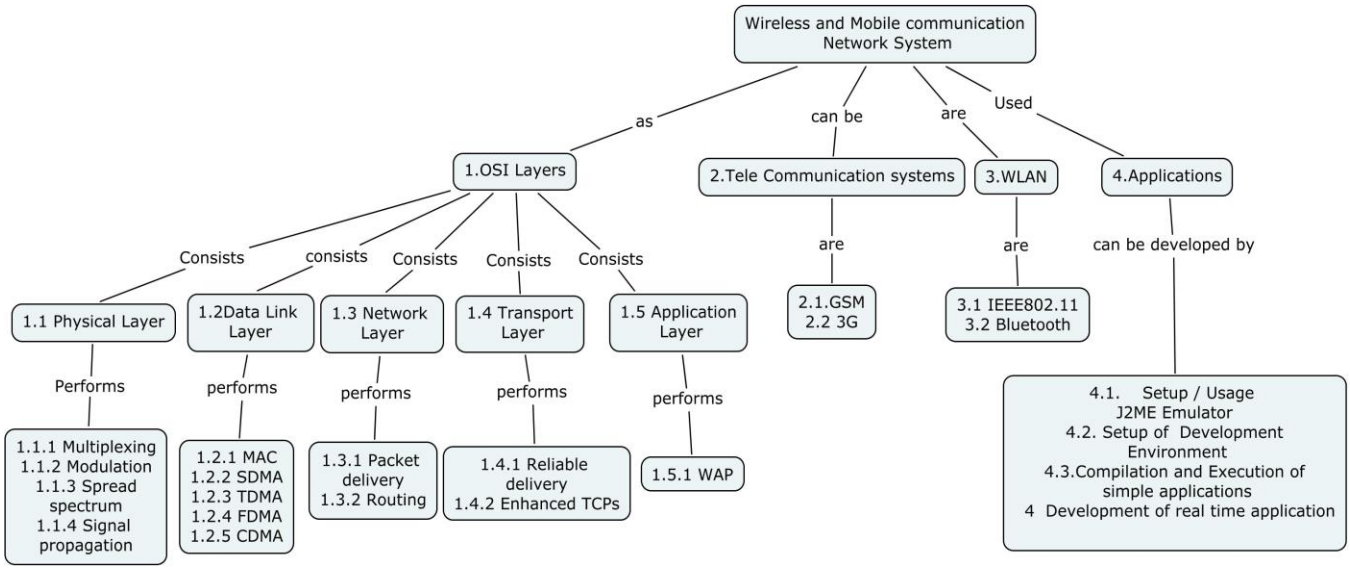
1. What are the characteristics of different multiplexing methods?
2. What are the benefits of reservation Schemes?
3. List the out the advantages of TDM over FDM.
4. Define Multiplexing.
5. Specify how Hidden terminal problem is solved by MAC.
6. How SDM is typically realized and combined with FDM?
7. What are WML decks and cards?
8. Which is the TDMA MAC scheme that provides lowest latency?

Understand

1. Describe how the collisions are avoided while transmission.
2. Mention the different types of telecommunication system Architectures.
3. Describe the steps involved Handover.
4. Compare and contrast the different MAC schemes.
5. Explain how TDMA differs from CDMA.
6. Draw and explain the GSM System Architecture.
7. Categorize the GSM components based on its function and explain them.
8. Give reasons for a handover in GSM and explain the problems associated with it.
9. Differentiate the Hard and soft handover.
10. Find out which method allows max user to share the single medium? How?
11. Write WMLScript to validate user inputs.
12. How is WMLScript used to access facilities of user agents?
13. How is WAP used for finding phone addresses, sports results?
14. Show how the packet will be delivered to the destination node in case of its current address in foreign network.
15. Illustrate why and where the encapsulation is needed in mobility environment.
16. Explain which of the MAC schemes can be used solve the hidden and exposed terminal problem
17. What are the consequences of using DAMA to solve Exposed terminal problem?
18. What are the consequences of using MACA to solve Exposed terminal problem?

19. What are the problems encountered if Transport layer functions of wired communication systems are used in wireless environment?
20. Explain how Mobile IP packets are delivered in a wireless network.

Concept Map



Syllabus:

OSI Layers - Physical Layer**:** Multiplexing, Modulation, Spread spectrum, Signal propagation; Data Link Layer: Motivation for Specialized MAC,SDMA,TDMA,FDMA,CDMA; Network Layer: IP Packet delivery, Routing; Transport Layer: Reliable delivery, Enhanced TCP's; Application Layer: WAP **Telecommunication Systems** – GSM,3G **Wireless LAN** - IEEE802.11 System Architecture, IEEE802.11 Protocol Architecture, Bluetooth Architecture **Developing Mobile Applications Using Wireless Toolkit –** Using toolkit, Using programming languages

Text Book

1. Jochen.H.Schiller "Mobile Communications" Addision- Wesley 2000

2. Kaveh Pahlavan & Prashant Krishnamurthy, Principles of Wireless networks, Pearson Education, 2003.

3. Jonathan Zdziarski, iPhone SDK, Oreilly, 2009

References

1. T.S.Rappaport, Wireless Communication, Prentice Hall, NJ 2002

2. Steele Lee and Gluis, JW, GSM, CDMA and 3G systems, 2001

3. William Stallings, Wireless Communications and Networks, Pearson Education, 2003

4. W.C.Y.Lee., Mobile Communication Engineering, Tata McGraw Hill, 1998.

Course contents and Lecture Schedule

Sl. No	Topics	No of Periods
1	**OSI Layers**	
1.1	Physical Layer**:** Multiplexing, Modulation, Spread spectrum, Signal propagation.	6
1.2	Data Link Layer: Motivation for Specialized MAC,SDMA,TDMA,FDMA,CDMA	6
1.3	Network Layer: IP Packet delivery, Routing	4
1.4	Transport Layer: Reliable delivery, Enhanced TCP's	4
1.5	*Application Layer: WAP*	3
2	**Telecommunication Systems**	
2.1	GSM	4
2.2	3G	3
3	**Wireless LAN**	
3.1	IEEE802.11 System and Protocol Architecture	6
3.2	Bluetooth Architecture	4
4	**Developing Mobile Applications Using Wireless Toolkit**	
4.1	Using toolkit	5
4.2	Using Programming Languages	5

Course Designers:

1. S.Muthuramalingam smrit@tce.edu
2. A.M.Abirami abiramiam@tce.edu

Sub code	Lectures	Tutorial	Practical	Credit
TCH	3	0	--	3

TCH Distributed Systems 3:0

Preamble:

A distributed system is a computer system consisting of several independent computers, connected by a network, that can work together to perform a task or provide a service. In this course we aim to provide students with a deeper understanding of distributed systems. In particular we focus on the principles, techniques, and practices relevant to the design and implementation of such systems. This course presents the principles underlying the functioning of distributed systems. It creates an awareness of the major technical challenges in distributed systems design and implementation. It exposes students to modern and classic technology used in distributed systems and their software.

Programme Outcomes addressed

a. Graduates will demonstrate an ability to identify, formulate and solve Distributed systems problems.

b. Graduates will demonstrate an ability to design a distributed system, component or process as per needs and specifications.

c. Graduate will demonstrate skills to use modern engineering tools, software and equipment to analyze problems.

d. An ability to design and conduct experiments, as well as to analyze and interpret data in Distributed system.

Competencies

1. Explain what a distributed system is, why you would design a system as a distributed system, and what the desired properties of such systems are;
2. Understand the principles underlying the functioning of distributed systems, describe the problems and challenges associated with these principles, and evaluate the effectiveness and shortcomings of their solutions;
3. Recognize how the principles are applied in contemporary distributed systems, explain how they affect the software design, and be able to identify features and design decisions that may cause problems;
4. Design a distributed system that fulfills requirements with regards to key distributed systems properties (such as scalability, transparency, etc.).

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3/End-semester examination
1	Remember	20	20	0
2	Understand	30	30	30
3	Apply	20	30	30
4	Analyze	20	0	20
5	Evaluate	0	0	0
6	Create	10	20	20

Learning Objectives:

Remember

1. What is an distributed system and what benefits does openness provide?
2. What is three-tired client-server architecture?
3. What is the role of middleware in a distributed system?
4. What is Concurrent server?
5. What is MPI-bsend and MPI-isend in MPI?
6. Does a stateless server need to take checkpoints?
7. What is the key difference between object references in CORBA and those in Globe?

Understand

1. What is meant by transparency, and give examples of different types of transparency.
2. Why is it sometimes so hard to hide the occurrence and recovery from failures in a distributed system?
3. What is the difference between a vertical distribution and horizontal distribution?
4. Why are transport-level communication services often inappropriate for building distributed applications?
5. Describe how connectionless communication between a client and a server proceeds when using sockets.
6. Explain difference between the primitives MPI-bsend and MPI-isend in MPI.
7. Consider a personal mailbox for a mobile user, implement as part of a wide-area distributed database. What kind of client-centric consistency would be most appropriate?

8. Describe a simple implementation of read-your-writes consistency for displaying Web pages that have just been updated.
9. Can the model of triple modular redundancy described in the text handle Byzantine failures? Consider a process P that requires access to file F which is locally available on the machine where P is currently running. When P moves to another machine, it still requires access to F. If the file to machine binding is fixed, how could the system wide reference to F be implemented?
10. Considering that a two-tired home-based approach is a specialization of a hierarchical location service, where is the root?

Apply

1. Consider the behavior of two machines in a distributed system. Both have clocks that are supposed to tick 1000 times per millisecond. One of them actually does, but the other ticks only 990 times per millisecond. If UTC updates come in once a minute, what is the maximum clock skew that will occur?
2. Consider a procedure incr with two integer parameters. The procedure adds one to each parameter. Now suppose that it is called with the same variable twice, for example, as incr(i,i). If I is initially 0, what value will it have afterward if call by reference is used? How about if copy/restore is used?
3. A file is replicated on 10 servers. List all the combinations of read quorum and write quorum that are permitted by the voting algorithm.
4. In the following 3 concurrently-executing processes table,

Process P1	Process P2	Process P3
x←1; print(y,z);	y← 1; print(x,z);	z← 1; print(x,y);

is 001110 a legal output for a sequentially consistent memory? Explain your answer.

Analyze

1. Consider a chain of process P1, P2, ….Pn, implementing a multi tiered client server architecture. Process Pi is client of process P_{i+1}, and P_i will return a reply to P_{i-1} only after receiving a reply from P_{i+1}. What are the main problems with this organization when taking a look at the request-reply performance at process P1?
2. Assume a client calls an asynchronous RPC to a server, and subsequently waits until the server returns a result using another asynchronous RPC. Is this

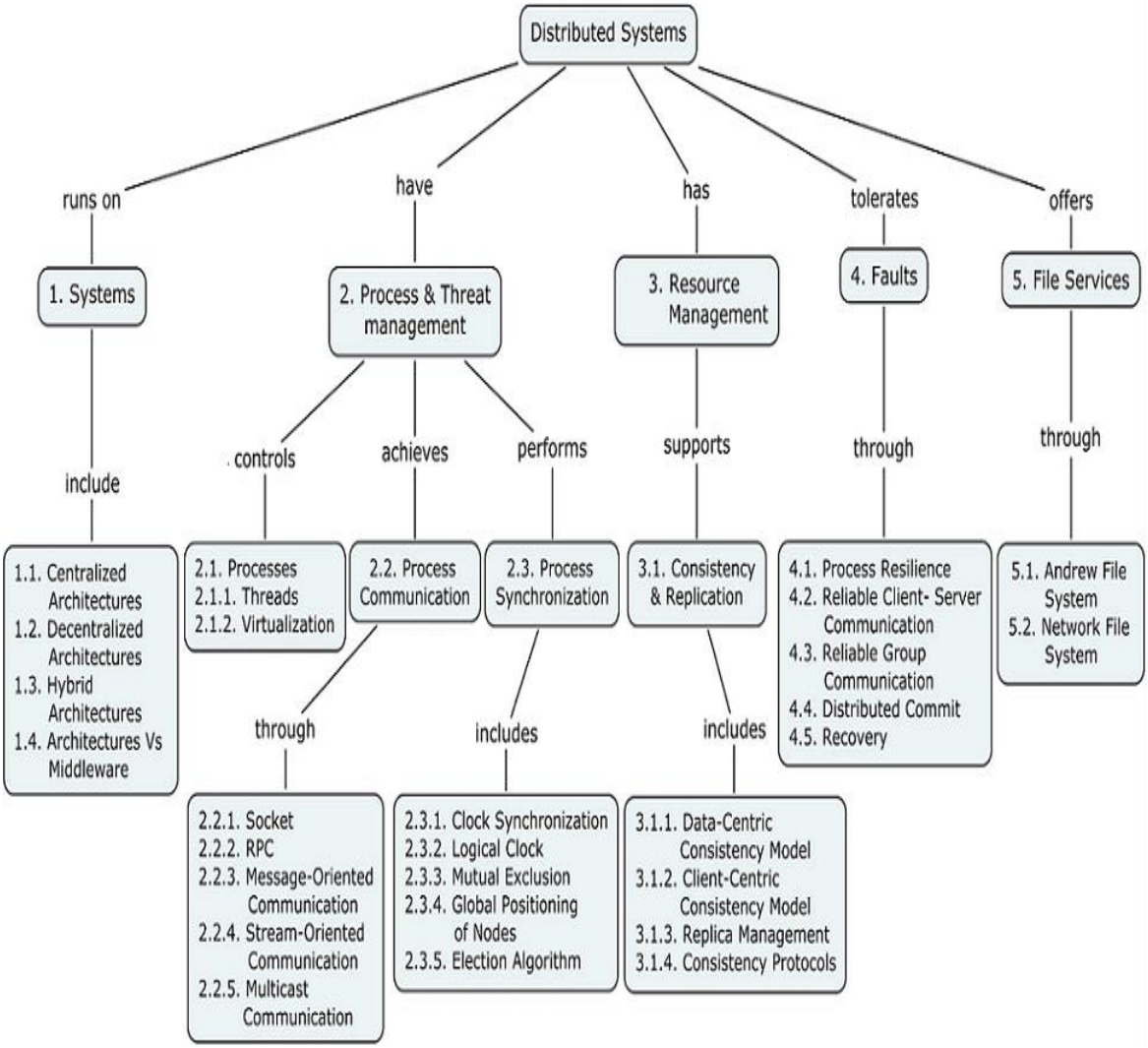
approach the same as letting the client execute a normal RPC? What if we replace the asynchronous RPCs with asynchronous RPCs?

3. Imagine a Web server that maintains a table in which client IP addresses are mapped to the most recently accessed Web pages. When a client connects to the serve, the server looks up the client in its table, and if found, returns the registered page. Is this server stateful or stateless?
4. Ricart and Agrawala's algorithm has the problem that if a process has crashed and does not reply to a request from another process to access resources, that lack of response will be interpreted as denial of permission. We suggested that all requests be answered immediately to make it easy to detect crashed processes. Are there any circumstances where even this method is insufficient? Explain your answer.

Create

1. Despite that multicasting is technically feasible, there is very little support to deploy it in the Internet. The answer tot his problem is to be sought in down-to-earth business models: no one really knows to make money out of multicasting. What scheme can you invent?
2. A Distributed system may have multiple, independent resource. Imagine that process 0 wants to assess resource A and Process1 wants to assess resource B. Create Ricart and Agarwala's algorithm lead to deadlocks.
3. Develop a multithreaded Java program that responds to remote clients' requests for meaning of words stored in a Dictionary. If a client program sends a message "King" to the server, the server program responds back with the meaning of word "King" by retrieving it from the dictionary (as a string). Use Java Sockets for communication between clients and the server.
4. UNIX systems provide many facilities to keep computers in synch, notably the combination of the crontab tool and various synchronization commands are powerful. Configure a UNIX system that keeps the local time accurate with in the range of a single second. Likewise. Configure an automatic backup facility by which a number of crucial files are automatically transferred to a remote machine once every 5 minutes. Your solution should be efficient when it comes to bandwidth usage.

Concept Map:



Syllabus

Introduction - Definition of a Distributed System, Goals, Types of Distributed Systems. Architectures - Architectural Styles, System Architecture, Architectures Vs Middleware. Processes – Threads, Virtualization. Communication – Fundamentals, Remote Procedure Call, Message-Oriented Communication, Stream-Oriented Communication, Multicast Communication. Synchronization - Clock Synchronization, Logical Clocks, Mutual Exclusion, Global Positioning of Nodes, Election Algorithm. Consistency and Replication – Introduction, Data-Centric Consistency Models, Client-Centric Consistency Models, Replica Management, Consistency Protocols. Fault Tolerance - Introduction , Process Resilience, Reliable Client-Server Communication,

Reliable Group Communication, Distributed Commit, Recovery. Case Studies: NFS, AFS.

Textbook:

1. Andrew S. Tanenbaum, Maarten Van Steen: Distributed Systems, Second Edition, Pearson Education,2008.

References:

1. Peterson L. and Davie B.: Computer Networks, A System Approach. San Mateo, CA: Morgon Kaufman, 3rd Edition, 2006.
2. Shooman M.L.: Reliability of Computer Systems and Networks: Fault Tolerance, Analysis and design. New York: John Wiley, 2002.
3. Verissimo P. and Rodrigues L.: Distributed Systems for Systems Architects. Kluwer Academic Publishers, 2001.

Course contents Lecture Schedule:

S. No.	Topics	No of periods
1	Introduction of Distributed Systems: Definition, Goals and Types	1
1.1	Architectures: Architectural Styles	1
1.2	Centralized Architecture	2
1.3	Decentralized Architecture	2
1.4	Hybrid Architecture	2
1.5	Architectures Vs Middleware	2
2.1	Processes	1
2.1.1	Threads	1
2.1.2	Virtualization	1
2.2	Communication	
2.2.1	Layer Protocols, Types of Communication	1
2.2.2	Remote Procedure call	2
2.2.3	Message Oriented Communication	2
2.2.4	Stream Oriented Communication	2
2.2.5	Multicast Communication	2
3	Synchronization	
3.1	Clock Synchronization	2
3.2	Logical Clocks	2
3.3	Mutual Exclusion	2

3.4	Global Positioning of Nodes	1
3.5	Election Algorithm	2
4	Consistency and Replication	
4.1	Data-Centric Consistency Models	2
4.2	Client- Centric Consistency Models	2
4.3	Replica Management	2
4.4	Consistency Protocols	2
5	Fault Tolerance	
5.1	Basic concepts of fault tolerance	1
5.2	Process Resilience	2
5.3	Reliable Client-Server Communication	2
5.4	Reliable Group Communication	2
5.5	Distributed Commit	1
5.6	Recovery	1
5.7	Case Studies: AFS,NFS	2

Course Designer

1. P. Venkata Krishnan - pvkit@tce.edu
2. S. Gilbertnancy - gilbertnancyl@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
TCI	3	0	-	3

TCI ARTIFICIAL INTELLIGENCE

Preamble

Artificial Intelligence aims at developing computer applications, which encompasses perception, reasoning and learning and to provide an in-depth understanding of major techniques used to simulate intelligence.

Program Outcomes Addressed

a. An ability to apply knowledge of engineering, information technology, mathematics and science.
b. An ability to design a system or component, or process to meet stated specifications.
c. An ability to identify, formulate and solve problems ranging from Optimization problems to Pattern recognition.

Competencies

1. Understand the various techniques involved in Artificial intelligence.
2. Apply searching techniques including informed search, uninformed search, adversial search, searching with constraints (CSP) to problems of traveling salesman, games, and scheduling.
3. Understand the behavior of agents including simple agent, goal-based agent, utility based agent, and reflex agent
4. Understand the representation of knowledge using first-order logic and ontological engineering
5. Apply learning approaches including 'learning from observation' (decision tree, inductive learning, ensemble learning), 'knowledge in learning' (explanation based learning, learning using relevance information, and inductive logic programming), and statistical learning (learning with complete data, learning with hidden variables, EM algorithm).

Assessment Pattern

No.	Bloom's Category	Test 1	Test 2	Test 3/End semester examination
1	Remember	40	40	20
2	Understand	40	40	30
3	Apply	20	20	50
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	0	0	0

Course Level Learning Objectives:

Remember

1. What is the use of heuristic functions?
2. Define artificial intelligence.
3. What is a constraint satisfaction problem?
4. What is a unification algorithm?
5. Explain the algorithm for steepest hill climbing.
6. Explain the characteristics of an AI technique.
7. Explain the role of cognitive science in modeling human behavior.
8. Explain the role of state space approach in solving AI problem.
9. Write short notes on
10. Baye's rule
11. Forward and backward chaining.
12. What is Artificial Intelligence? Explain how an AI system is different from a conventional computing System.

Understand

1. Explain the following search strategies.
   (i) Best first search
   (ii) A* search.
2. Illustrate the learning from examples by induction with suitable examples.
3. What is the role of an agent program?
4. Compare forward chaining and backward chaining.
5. Explain the characteristics of an AI technique.
6. Explain the role of cognitive science in modeling human behavior.

7. Explain the role of state space approach in solving AI problem.
8. With suitable examples illustrate how do we represent simple facts in logic?
9. Assume the facts
10. Steve only like's easy courses

    Science courses are hard.

    All courses in the arts Department are easy.

    TL 301 is an arts department course.

    Use resolution to answer the question which course would Steve like?
11. Prove each of the following statements:
    a. Breadth first search is a special case of uniform cost search.
    b. Breadth first, depth first and uniform cost search are special cases of best first search.
    c. Uniform cost search is a special case of A* search.

Apply

1. Consider the following sentences:

   John likes all kinds of food.

   Apples are food.

   Chicken is food.

   Anything anyone eats and isn't killed alive.

   Sue eats everything bill eats.

   (i) Translate these sentences into formulas in predicate logic.

   (ii) Prove that john likes peanuts using backward chaining.

   (iii) Convert the formulas of a part into clause form.

   (iv) Prove that john likes peanuts using resolution.
2. A sliding-tile puzzle consists of three black tiles, three white tiles and an empty space, thus:

B1	B2	B3		W1	W2	W3

   There are three legal ways of moving a tile, each with an associated cost:

   a. Slide into the adjacent empty location — cost 1

   b. Jump over one tile into the empty location — cost 1

   c. Jump over two tiles into the empty location — cost 2

   The goal is to have all the white tiles to the left of all the black tiles and to achieve this at minimum cost. The final position of the empty space is not important.

(a) Represent the problem using the following knowledge representation schemes:

(i) Production system rules

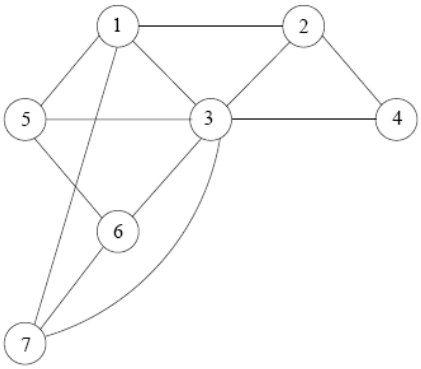
(ii) A semantic network

In one sentence, describe the different emphases of these two schemes.

(b) State two possible heuristics to help solve this problem.

(c) For a planner to solve this puzzle, what operators (i.e. planning actions) would be needed?

3. This question addresses the problem of coloring the following graph using a constraint satisfaction approach.



The colors available are amber, black and crimson which we will denote by A, B and C respectively. We want to assign a color to each node in the graph in such a way that if there is an edge (n1, n2) between any pair of nodes then n1 and n2 have different colors.

a) Represent the above problem as a constraint satisfaction problem.

b) Apply forward checking in the process of solving a constraint satisfaction problem. Illustrate your answer using the above graph colouring problem where the initial steps are, in order, 1 = A, 2 = B, 6 = B, 5 = C. In particular, you should show how the process can reduce branching and induce backtracking.

c) Apply constraint propagation using arc consistency in the process of solving a constraint satisfaction problem. Illustrate your answer using the same initial steps in the same order. Determine whether or not backtracking occurs earlier in this case and explain why.

4. This question considers the monkey-and-bananas problem, in which there is a monkey in a room with some bananas hanging out of reach from the ceiling, but a box is available that will enable the monkey to reach the bananas if he climbs onto it. Initially the monkey is at location A, the bananas at B and the box at C.

The monkey and box have height x, but if the monkey climbs onto the box he will have height y, the same as the bananas. The actions available to the monkey include Go from one place to another, Push an object from one place to another, Climb onto an object, and Grasp an object. Grasping results in holding the object if the monkey and object are in the same place at the same height.

a) Write the initial state description using a representation of your choice.
b) Write definitions of the four actions, providing at least some obvious preconditions, additions and deletions.
c) Suppose the monkey wants to fool the observers, who have gone to lunch, by grabbing the bananas but leaving the box in its original place. Write this as a goal (but not assuming the box is necessarily at location C) in the language of situation calculus.
d) If the box is filled with bricks, its position will remain the same when the Push operator is applied. Is this an example of the frame problem or the circumscription problem? Justify your answer.

5. Consider the following learning problem in which we wish to classify inputs, each consisting of a single real number, into one of two possible classes C1 and C2. There are three potential hypotheses where Pr(h1) = 3/10, Pr(h2) = 5/10 and Pr(h3) = 2/10. The hypotheses are the following functions

$$h_i(x) = x - \frac{i-1}{5}$$

and the likelihood for any hypothesis h_i is

$$\Pr(x \in C_1 | h_i, x) = \sigma(h_i(x))$$

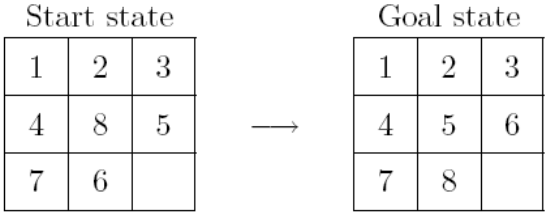
where $\sigma(y) = 1/(1+\exp(-y))$. You have seen three examples: (0.9,C_1), (0.95,C_2) and (1.3,C_2), and you now wish to classify the new point x = 1.1.

a) Explain how in general the maximum a posteriori (MAP) classifier works.
b) Compute the class that the MAP classifier would predict in this case.
c) The preferred alternative to the MAP classifier is the Bayesian classifier, computing $\Pr(x \in C_1 | x, s)$, where s is the vector of examples. Show that

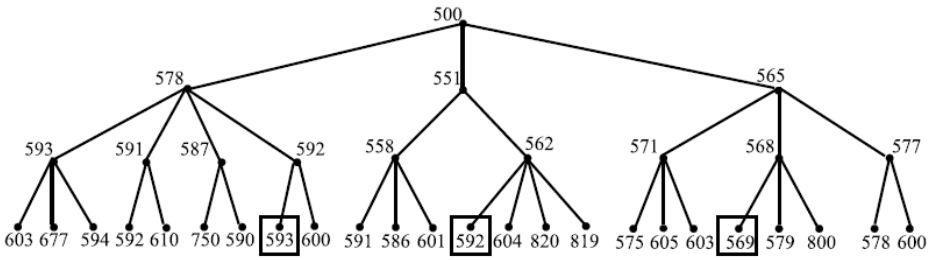
$$\Pr(x \in C_1 | x, s) = \sum_{h_i} \Pr(x \in C_1 | h_i, x) \Pr(h_i | s)$$

What are you assuming about independence in deriving this result?
d) Compute the class that the Bayesian classifier would predict in this case.

6. Draw a search tree for the 8-puzzle problem up to depth 4 (start state is depth 0) using the A* algorithm (omit repeated states) with the evaluation function f(n) = p(n)+h(n), where p(n) is the number of steps from the start state (start state is step 0) and h(n) is the number of misplaced tiles. Note that the actions for sliding tiles should be used in this order: right, left, up and down. Write the values of f and of its components p and h under each state. You may use an abbreviated notation indicating only the tiles that change.



8. Consider the following search tree.



The numbers by the nodes denote the sum of some path cost and heuristic. The boxed nodes are goals. Describe in detail the way in which the RBFS algorithm searches this tree. Your answer should indicate the order in which nodes are expanded, the reason that this order is used, and should state which of the three goals is found and why. Note that smaller numbers represent more desirable nodes.

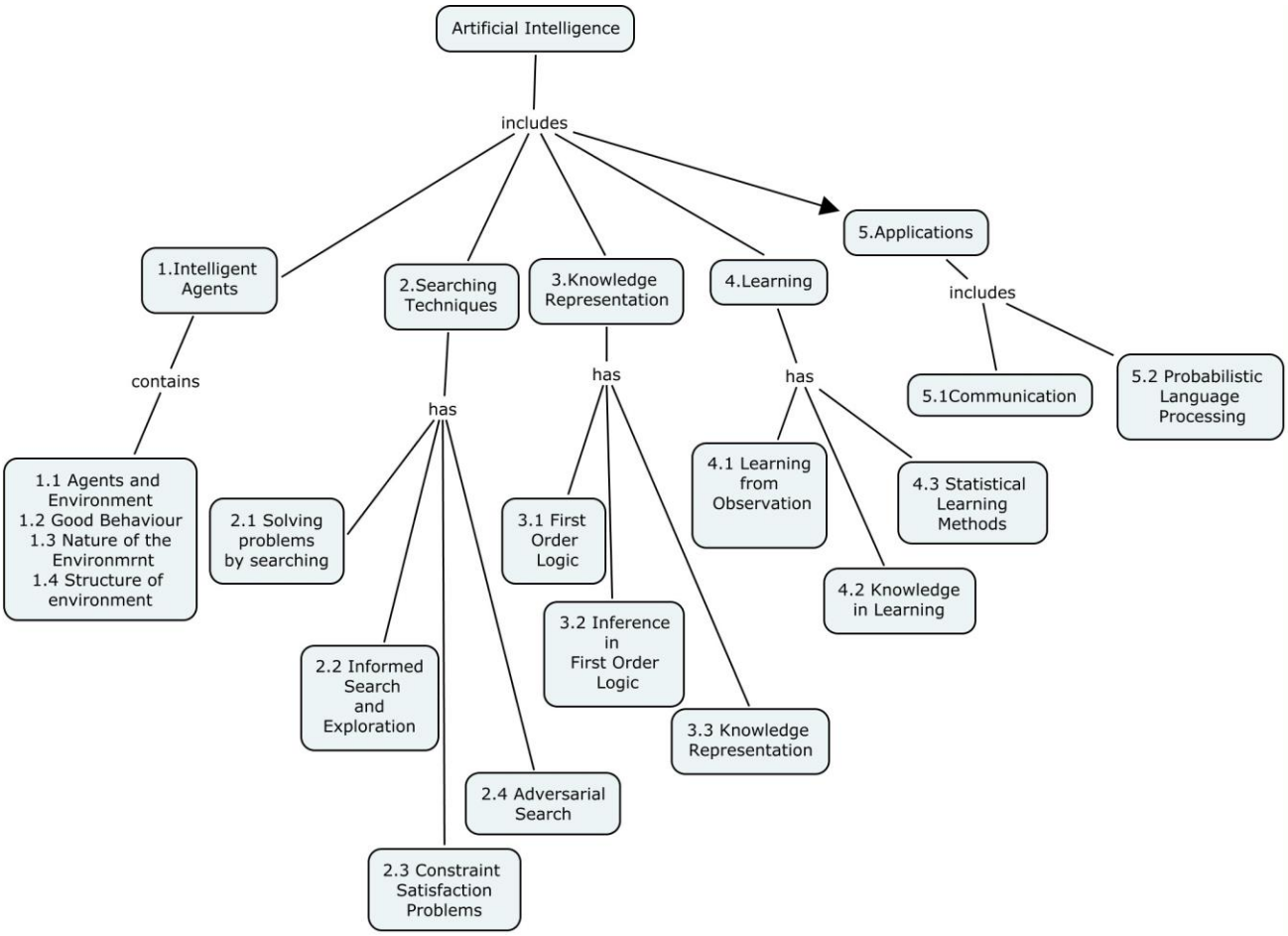
8. (a) What kinds of knowledge would an automated travel agent need to represent?

(b)Using the automated travel agent as an example, define five problems connected with knowledge representation.

(c)Would an expert system be a suitable technique for implementing an automated travel agent?

9. For a searching application, explain how genetic algorithms differ from conventional mathematical methods for optimization. What are the advantages and disadvantages of genetic algorithms?

Concept Map



Syllabus

INTRODUCTION - Intelligent Agents – Agents and environments - Good behavior – The nature of environments – structure of agents - Problem Solving - problem solving agents – example problems – searching for solutions – uniformed search strategies - avoiding repeated states – searching with partial information.

SEARCHING TECHNIQUES - Informed search and exploration – Informed search strategies – heuristic function – local search algorithms and optimistic problems – local search in continuous spaces – online search agents and unknown environments - Constraint satisfaction problems (CSP) – Backtracking search and Local search for CSP – Structure of problems - Adversarial Search – Games – Optimal decisions in games – Alpha – Beta Pruning – imperfect real-time decision – games that include an element of chance.

KNOWLEDGE REPRESENTATION - First order logic – representation revisited – Syntax and semantics for first order logic – Using first order logic – Knowledge

engineering in first order logic - Inference in First order logic – prepositional versus first order logic – unification and lifting – forward chaining – backward chaining - Resolution - Knowledge representation - Ontological Engineering - Categories and objects – Actions - Simulation and events - Mental events and mental objects. **LEARNING** - Learning from observations - forms of learning - Inductive learning - Learning decision trees - Ensemble learning - Knowledge in learning – Logical formulation of learning – Explanation based learning – Learning using relevant information – Inductive logic programming - Statistical learning methods - Learning with complete data - Learning with hidden variable - EM algorithm. **APPLICATIONS** - Communication – Communication as action – Formal grammar for a fragment of English – Syntactic analysis – Augmented grammars – Semantic interpretation – Ambiguity and disambiguation – Discourse understanding – Grammar induction - Probabilistic language processing - Probabilistic language models – Information retrieval – Information Extraction – Machine translation.

Text and Reference books:

1. Stuart Russell, Peter Norvig, "Artificial Intelligence – A Modern Approach", 2nd Edition, Pearson Education / Prentice Hall of India, 2004.
2. Nils J. Nilsson, "Artificial Intelligence: A new Synthesis", Harcourt Asia Pvt. Ltd., 2000.
3. Elaine Rich and Kevin Knight, "Artificial Intelligence", 2nd Edition, Tata McGraw-Hill, 2003.
4. George F. Luger, "Artificial Intelligence-Structures and Strategies for Complex Problem Solving", Pearson Education / PHI, 2002.

Course contents and Lecture Schedule

Sl. No	Topic	No. of Lectures
1	**INTRODUCTION –Intelligent Agents**	
1.1	Agents and environment	4
1.2	Good behavior	2
1.3	Nature of the Environment	2
1.4	Structure of environment	2

2	**SEARCHING TECHNIQUES**	
2.1	Solving problems by searching	2
2.2	Informed search and exploration	2
2.3	Constraint satisfaction problems	3
2.4	Adversarial search	3
3	**KNOWLEDGE REPRESENTATION**	
3.1	First order logic	3
3.2	Inference in first order logic	4
3.3	Knowledge representation	3
4	**LEARNING**	
4.1	Learning from observations	3
4.2	Knowledge in learning	3
4.3	Statistical learning methods	4
5	**APPLICATIONS**	
5.1	Communication	5
5.2	Probabilistic language processing	5

Course designer

1. M.K.Kavitha Devi – mkkdit@tce.edu
2. K.V.Uma – kvuit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
TCJ	3	0	-	3

TCJ SOFTWARE DESIGN

(Common to CSE/IT: C56) 3:0

Preamble: Software design is concerned with techniques, strategies, representations, and patterns used to determine how to implement a component or a system.

Program me Outcomes addressed

a. Graduates will identify, formulate and solve engineering problems.

b. Graduates will use techniques, skills, and modern engineering tools to implement and organize engineering works under given constraints.

c. Graduates will able to work on multidisciplinary teams.

Competencies

1. Explain the principles of software design
2. Understand the process of software architectural design
3. Understand the process of component level software design
4. Draw different levels of Data Flow Diagram to meet the given requirement specifications
5. Draw Jackson structured diagram to meet the given requirement specifications.
6. Create and specify the software design for a medium-size software product using Unified Modeling Language (UML).
7. Design Use Case, Class, State, Activity, Sequence, Collaboration and Deployment diagrams as per given system requirement specifications
8. Compare and contrast object-oriented analysis and design with structured analysis and design.
9. Design software systems for Web applications including e-library, air line reservation, e-banking and materials management.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	End-semester examination
1	Remember	20	20	20
2	Understand	20	20	20
3	Apply	40	30	20
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	20	30	40

Learning Objectives

Remember

1. What makes software design different from coding?
2. How are the concepts of coupling and software portability related?
3. List out the guidelines to be followed in UID.
4. Differentiate the sequence and collaboration diagrams
5. Name three other generic attributes that differentiate Web Applications for more conventional software applications?
6. Suggest three "golden rules" that would help guide the design of Web Applications.
7. What are the benefits of OOAD?
8. What is relationship? Give the relationships associated with the classes.
9. Mention the importance of model building.

Understand

1. Explain SSA/SD process in detail.
2. What is a Pattern? In what way it is used for modeling? Explain the modeling techniques that are used to model the Design Pattern and Architectural Pattern of an application.
3. What are the various building blocks of UML? Explain them in detail
4. Explain the relationships among the objects with an example
5. How the dynamic nature of a system can be known? Explain the UML diagrams that are used to show the dynamic nature of the system.
6. What is dependency? Where can it be used? Explain the stereotypes that can be used along with it.
7. Explain the following:
   (i) States (ii) transition (iii) Sub state
8. What is the purpose of a component diagram? Explain the modeling techniques that is used to model the source code, executable release, physical database, and adaptable system with suitable example.
9. Discuss the following diagrams with suitable example.
   (i) Component (ii) Deployment (iii) Collaboration

Apply

1. For the lending library, what operations of a library records and issues system will need to be modeled by using:
   a. interaction function processes
   b. Information function processes

2. In a simple banking system, one candidate for the role of an 'object' might be a customer account. Consider how each of the four major viewpoints might be used in modeling the operation of a bank account, and suggest forms that might be suitable for each viewpoint. (There is scope for using inheritance here, too, in that the bank may provide different forms of account, and although each will provide certain standard methods, the details of these might vary with the type of account involved.)
3. An issue records system for use in a public lending library is required. Suggest a set of suitable candidates for 'objects' in this system and for each of these identify the major attributes and provided operations.
4. Apply a "stepwise refinement approach" to develop three different levels of procedural abstraction for one or more of the following programs:
    a. Develop a check writer that, given a numeric dollar amount, will print the amount in words normally required on a check.
    b. Iteratively solve for the roots of a transcendental equation.
    c. Develop a simple round-robin scheduling algorithm for an operating system
5. Discuss the relative merits and difficulties of applying data flow-oriented design in the following areas: (a) embedded microprocessor applications, (b) engineering/scientific analysis, (c) computer graphics, (d) operating system design, (e) business applications, (f) database management system design
6. How does content analysis differ from interaction and functional analysis?

Create

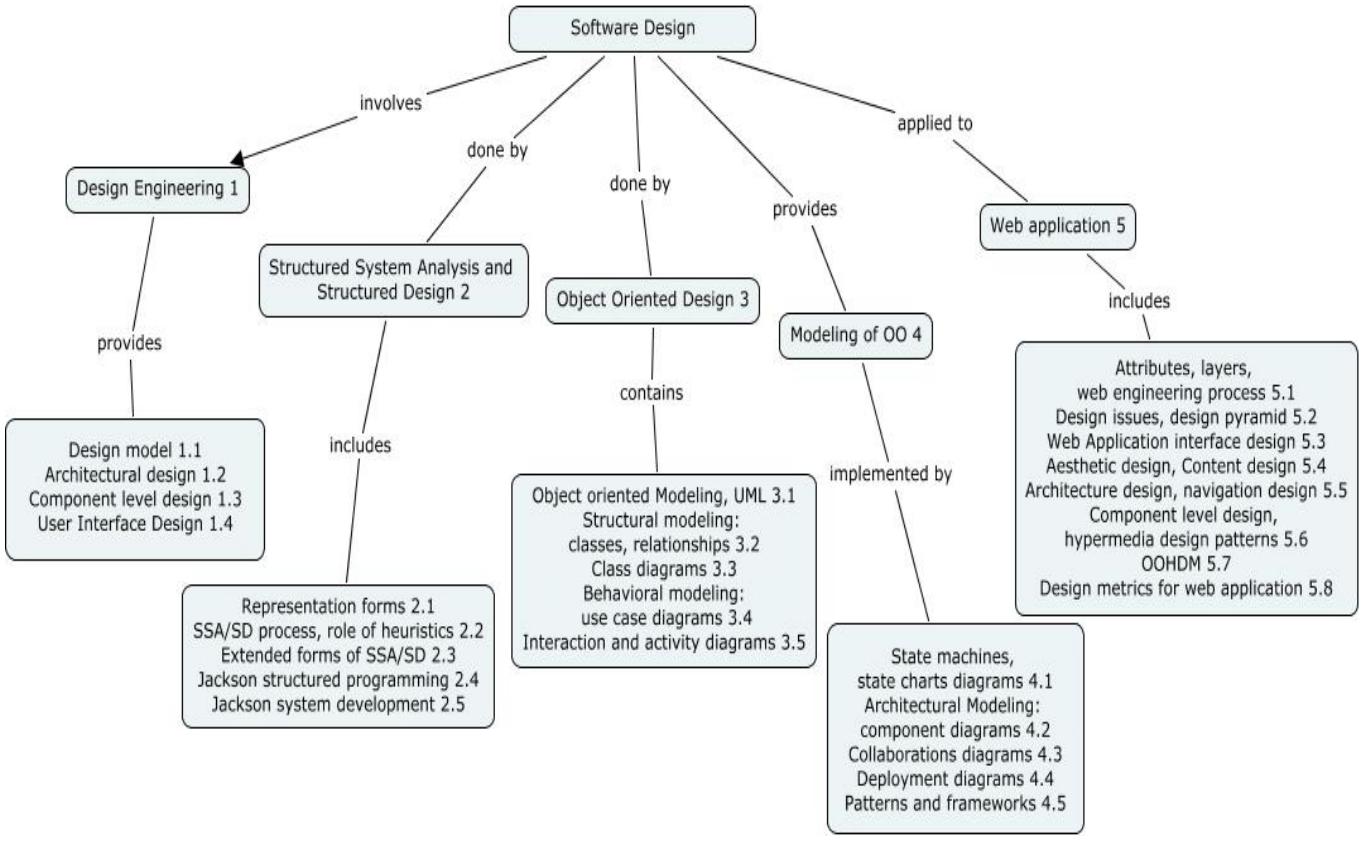
1. Draw the context diagram for each of the following systems:
    (a) a bank auto teller machine
    (b) a word-processing program
    (c) a payroll package that is required to produce a set of printed pay cheques each month and to provide pay statements showing amount paid and deductions
2. Draw a Structure Diagram that describes the address of a person in the 'standard American' format of surname, forename, initials, number, street, city, state, zip code, elaborating on the details of each of these in turn. How would this model cope with Jim Smith, who lives in Blacksmith's Cottage with an address that has no street name?
3. Construct an ESD that describes:
    (a) the 'customer' of a public lending library;

(b) the operational cycle of an automatic washing machine;

(c) the 'customer' of a car-hire firm.

4. Develop a set of screen layouts with a definition of major and minor menu items for the advanced *Safe Home* system
5. Develop an approach that would automatically integrate error messages and a user help facility. That is, the system would automatically recognize the error type and provide a help window with suggestions for correcting it. Perform a reasonably complete software design that considers appropriate data structures and algorithms.
6. Develop a design model, a user model, a system image, and a system perception for an Internet-based polling booth for public elections.
7. For the Traffic control management system draw the Use case , Class and Object diagrams using UML.
8. For the banking environment using the UML draw the following diagrams.
   (i) Use Case (ii) Sequence (iii) Activity

Concept Map



Course Content and Lecture Schedule

No	Topic	No of Lectures
1.	**Design Engineering (12)**	
1.1	Design concepts, Design model	1
1.2	Architectural design	4
1.3	Component level design	4
1.4	User Interface Design	3
2.	**Structured System Analysis and Structured Design (8)**	
2.1	Representation forms	1
2.2	SSA/SD process, role of heuristics	3
2.3	Extended forms of SSA/SD	2
2.4	Jackson structured programming	1
2.5	Jackson system development	1
3.	**Object Oriented Design (10)**	
3.1	Object oriented Modeling, UML	1
3.2	Structural modeling: classes, relationships	2
3.3	Class diagrams	2
3.4	Behavioral modeling: use case diagrams	2
3.5	Interaction and activity diagrams	3
4	**Modeling of OO (10)**	
4.1	State machines, state charts diagrams	3
4.2	Architectural Modeling: component diagrams	2
4.3	Collaborations diagrams	2
4.4	Deployment diagrams	2
4.5	Patterns and frameworks	1

5	**Web Engineering (10)**	
5.1	Attributes, layers, web engineering process	1
5.2	Design issues, design pyramid	1
5.3	Web Application interface design	1
5.4	Aesthetic design, Content design	1
5.5	Architecture design, navigation design	2
5.6	Component level design, hypermedia design patterns	1
5.7	OOHMD	2
5.8	Design metrics for web application	1

Syllabus

Design Engineering: Design concepts, Design model, Architectural design: software architecture, data design, architectural styles and patterns, architectural design, assessing alternative architectural designs, mapping data flow into a software architecture. Component level design: component, designing component-based design, conducting component level design, object constraint language, designing conventional components. User Interface Design: interface analysis, interface design steps.

Structured System Analysis and Structured Design: Representation forms, SSA/SD process, role of heuristics, extended forms of SSA/SD, Jackson structured programming, Jackson system development.

Object Oriented Design: Object oriented Modeling, UML, Structural modeling: classes, relationships, class diagrams. Behavioral modeling: use case, interaction and activity diagrams.

Modeling of OO: State machines, state charts diagrams. Architectural Modeling: component, collaborations, deployment diagrams, patterns and frameworks.

Web Engineering: Attributes, layers, web engineering process, design issues, design pyramid, Web Application interface design, aesthetic design, content design, architecture design, navigation design, component level design, hypermedia design patterns, OOHDM, design metrics for web application

Text Books :

1.Roger S. Pressman, Software Engineering A Practitioner's Approach, Sixth Edition, Mcgraw Hill International Edition.,2007

2. David Budgen "Software Design", Second Edition, Pearson Education.,2008

Reference Books:

1. Grady Booch, James Rumbaugh, Ivar Jacobson "The Unified Modeling Language – User Guide" Pearson Education, 2005

2. Ed Downs, Peter Clare, Jan Coe, "Structured System Analysis And Design Methods Application And Context", Prentice Hall, 1998.

Course Designers

1. A. Malini amcse@tce.edu
2. A. Askarunisa aacse@tce.edu

Sub code	Lectures	Tutorial	Practical	Credit
TCH	3	0	--	3

TCH Distributed Systems 3:0

Preamble:

A distributed system is a computer system consisting of several independent computers, connected by a network, that can work together to perform a task or provide a service. In this course we aim to provide students with a deeper understanding of distributed systems. In particular we focus on the principles, techniques, and practices relevant to the design and implementation of such systems. This course presents the principles underlying the functioning of distributed systems. It creates an awareness of the major technical challenges in distributed systems design and implementation. It exposes students to modern and classic technology used in distributed systems and their software.

Programme Outcomes addressed

a. Graduates will demonstrate an ability to identify, formulate and solve Distributed systems problems.

b. Graduates will demonstrate an ability to design a distributed system, component or process as per needs and specifications.

c. Graduate will demonstrate skills to use modern engineering tools, software and equipment to analyze problems.

d. An ability to design and conduct experiments, as well as to analyze and interpret data in Distributed system.

Competencies

1. Explain what a distributed system is, why you would design a system as a distributed system, and what the desired properties of such systems are;
2. Understand the principles underlying the functioning of distributed systems, describe the problems and challenges associated with these principles, and evaluate the effectiveness and shortcomings of their solutions;
3. Recognize how the principles are applied in contemporary distributed systems, explain how they affect the software design, and be able to identify features and design decisions that may cause problems;
4. Design a distributed system that fulfills requirements with regards to key distributed systems properties (such as scalability, transparency, etc.).

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3/End-semester examination
1	Remember	20	20	0
2	Understand	30	30	30
3	Apply	20	30	30
4	Analyze	20	0	20
5	Evaluate	0	0	0
6	Create	10	20	20

Learning Objectives:

Remember

1. What is an distributed system and what benefits does openness provide?
2. What is three-tired client-server architecture?
3. What is the role of middleware in a distributed system?
4. What is Concurrent server?
5. What is MPI-bsend and MPI-isend in MPI?
6. Does a stateless server need to take checkpoints?
7. What is the key difference between object references in CORBA and those in Globe?

Understand

1. What is meant by transparency, and give examples of different types of transparency.
2. Why is it sometimes so hard to hide the occurrence and recovery from failures in a distributed system?
3. What is the difference between a vertical distribution and horizontal distribution?
4. Why are transport-level communication services often inappropriate for building distributed applications?
5. Describe how connectionless communication between a client and a server proceeds when using sockets.
6. Explain difference between the primitives MPI-bsend and MPI-isend in MPI.

7. Consider a personal mailbox for a mobile user, implement as part of a wide-area distributed database. What kind of client-centric consistency would be most appropriate?
8. Describe a simple implementation of read-your-writes consistency for displaying Web pages that have just been updated.
9. Can the model of triple modular redundancy described in the text handle Byzantine failures? Consider a process P that requires access to file F which is locally available on the machine where P is currently running. When P moves to another machine, it still requires access to F. If the file to machine binding is fixed, how could the system wide reference to F be implemented?
10. Considering that a two-tired home-based approach is a specialization of a hierarchical location service, where is the root?

Apply

1. Consider the behavior of two machines in a distributed system. Both have clocks that are supposed to tick 1000 times per millisecond. One of them actually does, but the other ticks only 990 times per millisecond. If UTC updates come in once a minute, what is the maximum clock skew that will occur?
2. Consider a procedure incr with two integer parameters. The procedure adds one to each parameter. Now suppose that it is called with the same variable twice, for example, as incr(i,i). If I is initially 0, what value will it have afterward if call by reference is used? How about if copy/restore is used?
3. A file is replicated on 10 servers. List all the combinations of read quorum and write quorum that are permitted by the voting algorithm.
4. In the following 3 concurrently-executing processes table,

Process P1	Process P2	Process P3
x←1; print(y,z);	y← 1; print(x,z);	z← 1; print(x,y);

is 001110 a legal output for a sequentially consistent memory? Explain your answer.

Analyze

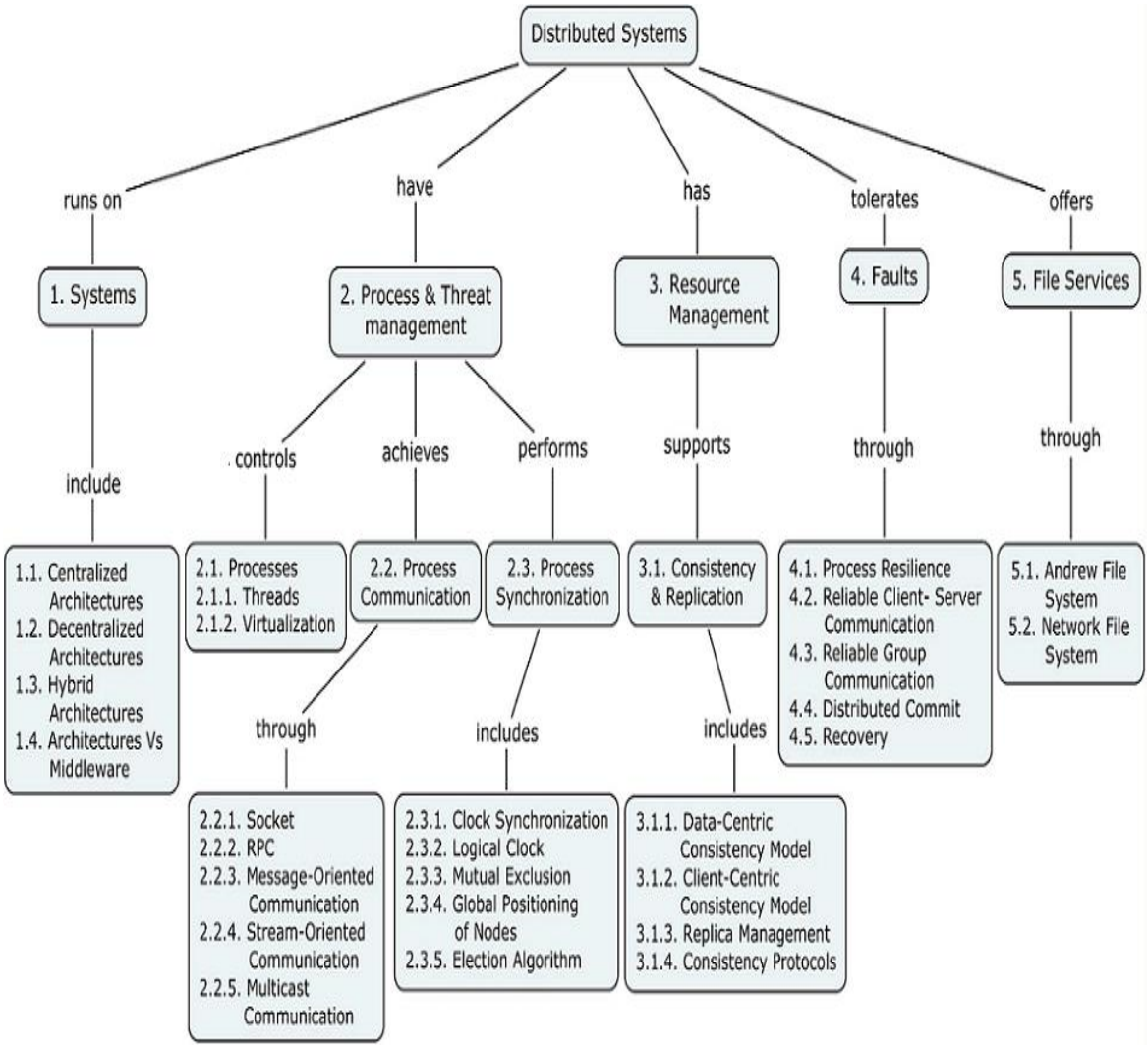
1. Consider a chain of process P1, P2, ….Pn, implementing a multi tiered client server architecture. Process Pi is client of process P_{i+1}, and P_i will return a reply to P_{i-1} only after receiving a reply from P_{i+1}. What are the main problems with this organization when taking a look at the request-reply performance at process P1?

2. Assume a client calls an asynchronous RPC to a server, and subsequently waits until the server returns a result using another asynchronous RPC. Is this approach the same as letting the client execute a normal RPC? What if we replace the asynchronous RPCs with asynchronous RPCs?
3. Imagine a Web server that maintains a table in which client IP addresses are mapped to the most recently accessed Web pages. When a client connects to the serve, the server looks up the client in its table, and if found, returns the registered page. Is this server stateful or stateless?
4. Ricart and Agrawala's algorithm has the problem that if a process has crashed and does not reply to a request from another process to access resources, that lack of response will be interpreted as denial of permission. We suggested that all requests be answered immediately to make it easy to detect crashed processes. Are there any circumstances where even this method is insufficient? Explain your answer.

Create

1. Despite that multicasting is technically feasible, there is very little support to deploy it in the Internet. The answer tot his problem is to be sought in down-to-earth business models: no one really knows to make money out of multicasting. What scheme can you invent?
2. A Distributed system may have multiple, independent resource. Imagine that process 0 wants to assess resource A and Process1 wants to assess resource B. Create Ricart and Agarwala's algorithm lead to deadlocks.
3. Develop a multithreaded Java program that responds to remote clients' requests for meaning of words stored in a Dictionary. If a client program sends a message "King" to the server, the server program responds back with the meaning of word "King" by retrieving it from the dictionary (as a string). Use Java Sockets for communication between clients and the server.
4. UNIX systems provide many facilities to keep computers in synch, notably the combination of the crontab tool and various synchronization commands are powerful. Configure a UNIX system that keeps the local time accurate with in the range of a single second. Likewise. Configure an automatic backup facility by which a number of crucial files are automatically transferred to a remote machine once every 5 minutes. Your solution should be efficient when it comes to bandwidth usage.

Concept Map:



Syllabus

Introduction - Definition of a Distributed System, Goals, Types of Distributed Systems. Architectures - Architectural Styles, System Architecture, Architectures Vs Middleware. Processes – Threads, Virtualization. Communication – Fundamentals, Remote Procedure Call, Message-Oriented Communication, Stream-Oriented Communication, Multicast Communication. Synchronization - Clock Synchronization, Logical Clocks, Mutual Exclusion, Global Positioning of Nodes, Election Algorithm. Consistency and Replication – Introduction, Data-Centric Consistency Models, Client-Centric Consistency Models, Replica Management, Consistency Protocols. Fault

Tolerance - Introduction , Process Resilience, Reliable Client-Server Communication, Reliable Group Communication, Distributed Commit, Recovery. Case Studies: NFS, AFS.

Textbook:

1. Andrew S. Tanenbaum, Maarten Van Steen: Distributed Systems, Second Edition, Pearson Education,2008.

References:

1. Peterson L. and Davie B.: Computer Networks, A System Approach. San Mateo, CA: Morgon Kaufman, 3rd Edition, 2006.
2. Shooman M.L.: Reliability of Computer Systems and Networks: Fault Tolerance, Analysis and design. New York: John Wiley, 2002.
3. Verissimo P. and Rodrigues L.: Distributed Systems for Systems Architects. Kluwer Academic Publishers, 2001.

Course contents Lecture Schedule:

S. No.	Topics	No of periods
1	Introduction of Distributed Systems: Definition, Goals and Types	1
1.1	Architectures: Architectural Styles	1
1.2	Centralized Architecture	2
1.3	Decentralized Architecture	2
1.4	Hybrid Architecture	2
1.5	Architectures Vs Middleware	2
2.1	Processes	1
2.1.1	Threads	1
2.1.2	Virtualization	1
2.2	Communication	
2.2.1	Layer Protocols, Types of Communication	1
2.2.2	Remote Procedure call	2
2.2.3	Message Oriented Communication	2
2.2.4	Stream Oriented Communication	2
2.2.5	Multicast Communication	2
3	Synchronization	
3.1	Clock Synchronization	2
3.2	Logical Clocks	2

3.3	Mutual Exclusion	2
3.4	Global Positioning of Nodes	1
3.5	Election Algorithm	2
4	Consistency and Replication	
4.1	Data-Centric Consistency Models	2
4.2	Client- Centric Consistency Models	2
4.3	Replica Management	2
4.4	Consistency Protocols	2
5	Fault Tolerance	
5.1	Basic concepts of fault tolerance	1
5.2	Process Resilience	2
5.3	Reliable Client-Server Communication	2
5.4	Reliable Group Communication	2
5.5	Distributed Commit	1
5.6	Recovery	1
5.7	Case Studies: AFS,NFS	2

Course Designer

1. P. Venkata Krishnan - pvkit@tce.edu
2. S. Gilbertnancy - gilbertnancyl@tce.edu

1.

BOARD OF STUDIES MEETING

B.Tech DEGREE (Information Technology) PROGRAM

Departmental Electives

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

REVISED CURRICULUM AND DETAILED SYLLABI

FOR

B.Tech DEGREE (Information Technology) PROGRAM

DEPARTMENTAL ELECTIVES

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2008-2009 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Department of Information Technology

Graduating Students of B.Tech program of IT will be able to

1. Explain and apply appropriate information technologies and employ appropriate methodologies to help an individual or organization achieve its goals and objectives.
2. Manage the information technology resources of an individual or organization.
3. Anticipate the changing direction of information technology and evaluate and communicate the likely utility of new technologies to an individual or organization.
4. Develop IT systems that would perform tasks related to E-governance and/or Health Care Management.
5. Work in a team using common tools and environments to achieve project objectives.

Sub. Code	Lecture	Tutorial	Practical	Credit
TCK	3	0	--	3

TCK Network Performance Analysis 3:0

Preamble

This course will enable the students to understand the principal factors that influence the performance of network systems. It helps the student to quantify the performance of network systems that supports data and multimedia applications, predict the impact of these factors on performances by analytical and simulation techniques.

Program Outcomes addressed

a. Graduates will demonstrate an ability to apply knowledge of mathematics, science and engineering.

c. Graduates will demonstrate an ability to design a system, component or process as per needs and specifications.

e. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

k. Graduates will demonstrate skills to use modern engineering tools, software and equipment to analyze problems.

Competencies:

At the end of the course, the students will be able to

1. Understand the modeling of queues using probabilities.
2. Understand how to analyze a simple queuing system.
3. Understand the state diagrams and Markov Chain.
4. Construct state diagrams for modeling of basic queues.
5. Derive steady state probabilities for simple queue.
6. Appraise the performance of Multi access systems such as MAC in terms of message traffic, flow control and access control.
7. Write simple simulations of Queuing systems using NS2, Glomosim.

Assessment Pattern

	Bloom's Category	**Test 1**	**Test 2**	**Test 3/End Semester Examination**
1	Remember	40	10	10
2	Understand	40	20	20
3	Apply	20	30	30
4	Analyze	0	40	40
5	Evaluate	0	0	0
6	Create	0	0	0

Course Level Learning Objectives

Remember

1. List out the six criteria to be considered for the design of network protocol.
2. What is the role of each layer in TCP/IP Reference model?
3. State the sum rule of probability?
4. What is Kendall's notation?
5. State Pollaczek-Khinchin formula?
6. What are the key properties of a Poisson process?
7. Define Birth-death process.
8. What do you mean by steady state

Understand

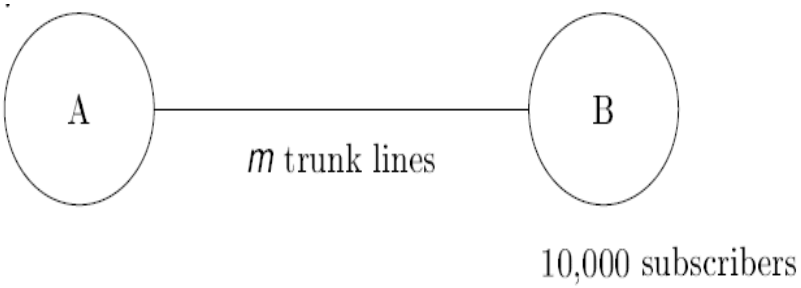
1. Differentiate dynamic and virtual circuit routing.
2. Compute the mean of the Poisson distribution, i.e. the expectation of k under the distribution P $(k/\mu) = \mu^k / k!\ e^{-\mu}$.
3. Distinguish Markov chains and state diagrams.
4. What is the average network delay when the system has reached steady state?
5. What is the use of utilization factor in M/M/1 queue?
6. What are the three factors that affects the busy time of ARQ system.
7. Explain how Jackson's theorem differs from Kleinrock's independence assumption?
8. What is the advantage of statistical multiplexing over time/frequency multiplexing?

Apply

1. There are three queues for registration from different courses at the University. Students for the first course, arrive with a rate of 4 students per minute. Students for the second course arrive with a rate of 2 students per minute and students for the third course arrive with a rate of 5 students a minute. Assuming the student arrivals are independent and occur as a Poisson process Find out the probability that after 15 minutes there have been a) 90 student arrivals? b) 180 arrivals?
2. You are driving past a hotel, wondering how long you will have to wait to go in. You notice on Thursday nights that the queue typically contains 20 people and that people arrive at a rate of 40 per hour. According to little's formula, how long would you expect to have to wait to get in?
3. Given a finite length M/M/1/m queue, what is the probability that an arrival is blocked from joining the queue (and therefore lost)?
4. You are at a queue in a post office and notice that the clerk spends 10% of his time between every three customers. If the typical number of people

waiting to be served is 3, what is the ratio of time spent waiting to being served?

5. The mean number in the system for the M/M/1 queue is given by considering the expectation under the steady state distribution. Compute this value.
6. Consider the telephone trucking problem. A telephone exchange A is to server 10,000 subscribers in a nearby exchange as shown in the given Figure. Suppose that during the busy hour, the subscribers generate Poisson traffic with a rate of 10 calls per minute, which requires trunk lines to exchange B for an average holding time of 3 minutes per call. Determine the number of trunk lines needed for a grade of service (probability of waiting) of 0.01.



Analyze

1. As a simple example let's consider a database (perhaps MySQL).We are constructing it from existing parts. We have two CPU/motherboard systems and two separate disk systems, their expected characteristics are shown in the Table. Which should we match with which? Are we better off building one faster system and one slower system? Or should we build two systems that have a similar performance?

CPU	System Expected Service Time/ms	Disk System	Expected Service Time/ms
A	423	1	215
B	210	2	390

2. 3. A call centre employs thirty operators selling tickets for a major sporting event. The call centre has sixty lines in total. When all operators are active new calls are placed in a queue for the first available operator. The call centre is to be modeled as an M/M/m/n queue, i.e. a queue in which there are m servers and a maximum state of n.

   (a) Draw a state diagram representing the queuing system.

   (b) Write down the detailed balance equations for the two cases where the queue state k < m and when m <k < n.

(c) For each of the two cases outlined in the previous part write down the steady state distribution as a function of P (0), the probability of the system being empty.

3. Analyze the Load Tripling system to study the effect of increasing load in an Erlang Loss system. Consider the following scenarios:

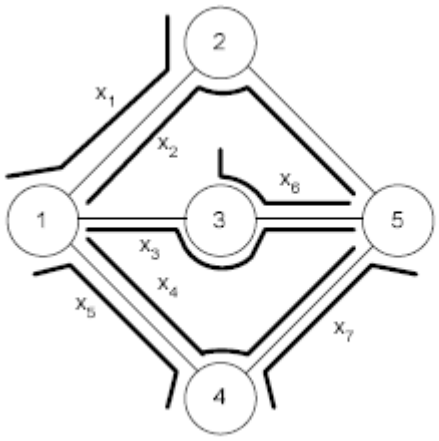
a) For a blocking probability of 0.01, what is the number of trunks needed for 6 Erlangs of offered load?

(b) For a blocking probability of 0.01, what is the number of trunks needed for 18 Erlangs?

(c) Comment on the increase in load compared to the increase in the number of trunks.

4. The Figure shows a small network with five nodes. In the network flow only occurs from lower numbered nodes to higher numbered nodes. The connections in the network have average service times in milliseconds as follows: Different routes through the network are indicated with thick lines.

$\bar{X}$	1	2	3	4	5
1	-	-	-	-	-
2	0.05	-	-	-	-
3	0.05	-	-	-	-
4	0.0333	-	-	-	-
5	-	0.05	0.05	0.033	-



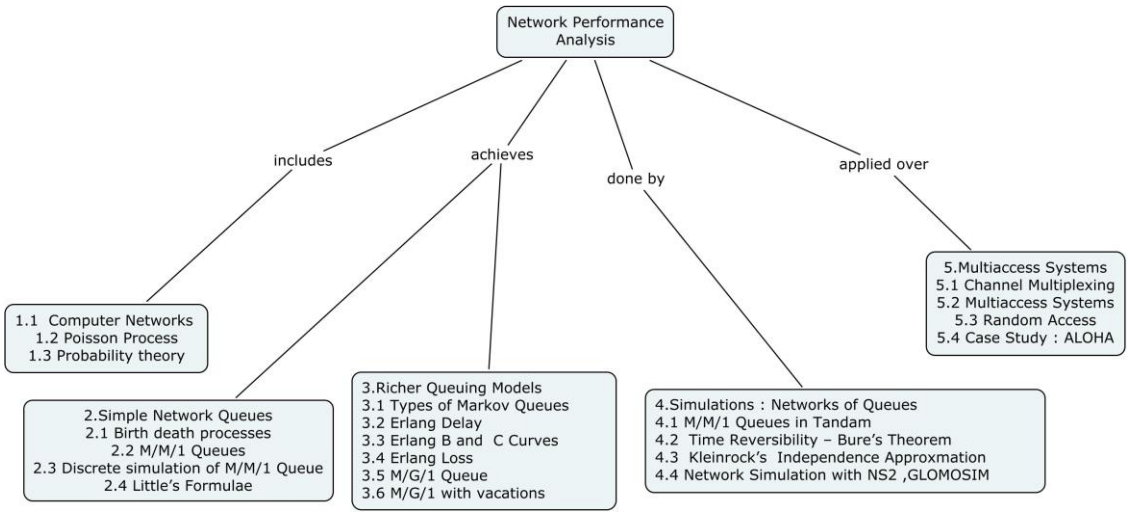
Each route has a rate associated with it as follows:

	x_1	x_2	x_3	x_4	x_5	x_6	x_7
λ	5	10	7	12	4	8	4

According to the Kleinrock independence assumption, what is the most congested link? What is the quickest route between node 1 & 5? In your answers you should consider propagation or processing delays at each node to be negligible.

5. Burke's theorem tells us that the departure process from the first queue will be a Poisson with rate λ. Now if we allow service times at the second queue to be independent of service times at the first queue we can break the dependence of the queue length at the second queue on the service time at the second queue. In this case the system can be modeled as two M/M/1 queue. What are the distributions for the states of the two systems?
6. In a CSMA/CD with a data rate of 10mbps, the maximum distance between any pair of station is found to be 2500 m for collision detection process. What should be the maximum distance if we increase the data rate to 100 mbps?

Concept Map:



Syllabus

Introduction: Computer Networks, Poisson Process, Probability theory .**Simple Network Queues:** Birth death processes, M/M/1 Queues, Discrete simulation of M/M/1 Queue, Little's Formulae. **Richer Queuing Models:** Types of Markov Queues, Erlang Delay, Erlang B and C Curves, Erlang Loss, M/G/1 Queue, M/G/1 with vacations. **Networks of Queues:** M/M/1 Queues in Tandam, Time Reversibility – Bure's Theorem, Kleinrock's Independence Approximation, Network Simulation with NS2 and Glomosim. **Multi access Systems:** Channel Multiplexing, Multi access Systems, Random Access, Case Study: ALOHA

Textbook

1. Dimitri Bertsekas Robert Gallager Data Networks 2nd edition, Prentice-Hall,1992, ISBN: 0132016745
2. Wah Chun Chan. Performance Analysis of Telecommunications and Local Area Networks. Kluwer, Dordrecht, The Netherlands, 2000. ISBN 0-7923-7701-X

References

1. Menasce, Almeida, Dowdy, "Performance By Design", 2004, Prentice Hall, ISBN 0-13-090673-5
2. Fortier, Michel, "Computer Systems Performance Evaluation and Prediction", 2003, Elsevier, ISBN 1-55558-260-5
3. Mieghem, "Performance Analysis of Communications Networks and Systems", 2006, Cambridge, ISBN 0-521-85515-2
4. John, Eeckhout, "Performance Evaluation and Benchmarking", 2006, CRC Press, ISBN 0-8493-3622-8
5. Raj Jain, The Art Of Computer Systems Performance Evaluation 1st Edition, Wiley ISBN 0-471-50336-3, 1991

Course Contents and Lecture Schedule:

S.No.	Topics	No of Lectures
1.	**Introduction**	
1.1	Networks Review	2
1.2	Poisson Process	2
1.3	Probability Review	2
2	**Simple Network Queues**	
2.1	Birth death processes	2
2.2	M/M/1 Queues	2
2.3	Discrete simulation of M/M/1 Queue	2
2.4	Little's Formulae	2
3	**Richer Queuing Models**	
3.1	Types of Markov Queues	2
3.2	Erlang Delay	1
3.3	Erlang B and C Curves	1
3.4	Erlang Loss	2
3.5	M/G/1 Queue	2
3.6	M/G/1 with vacations	2
4	**Simulations : Networks of Queues**	
4.1	M/M/1 Queues in Tandam	2
4. 2	Time Reversibility – Bure's Theorem	2
4. 3	Kleinrock's Independence Approximation	2

S.No.	Topics	No of Lectures
4.4	Network Simulation with NS2, Glomosim	4
5	**Multiaccess Systems**	
5.1	Channel Multiplexing	2
5.2	Multi access Systems	2
5.3	Random Access	2
5.4	Case Study : ALOHA	2
	Total	**42**

Course Designers:

1. S. Muthuramalingam smrit@tce.edu
2. S.Srinivasan ssnit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
TCL	3	0	-	3

TCL E-Commerce 3:0

Preamble

The growth of the Internet continues to have a tremendous influence on business. Companies and organizations of all types and sizes are rethinking their strategies and how they run their operations. This course challenges students to explore the realities and implications of e-commerce from a marketer's perspective. Business-to-consumer (B2C) and business-to-business (B2B) e-commerce markets are examined. The course introduces challenges in e-commerce.

Program Outcomes Addressed

a. Graduates will demonstrate an ability to apply knowledge of engineering, mathematics and science.

c. Graduates will demonstrate an ability to design a system, component, or process to meet desired needs.

e. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

Competencies

At the end of the course, students will be able to

1. Understand the basic concepts of e-commerce.
2. Ability to comprehend market dynamics.
3. Be aware of different types of e-commerce sites and different modes of payments.
4. Know about the security and legal issues in e-commerce.
5. Understand the technological building blocks used in construction of e-commerce applications such as software agents, desktop and broadband multimedia, mobile and cellular network.

Assessment Pattern

No.	Bloom's Category	Test 1	Test 2	Test 3/End semester examination
1	Remember	30	30	10
2	Understand	70	40	30
3	Apply	0	20	40
4	Analyze	0	10	20
5	Evaluate	0	0	0
6	Create	0	0	0

Course Level Learning Objectives:

Remember

1. What is e-commerce?
2. List some simple technological advances that derive the phenomenon of convergence
3. List some components of multimedia
4. What are smart cards?
5. What are the different types of smart cards?
6. What are the assumptions of smart cards?
7. List out the bundling strategies
8. What were expected to help improve a company's competitive position in 2010's to manage information overload?
9. What are the types of compression techniques?
10. What is meant by multiprocessing?

Understand

1. Explain the framework of e-commerce technologies
2. Explain e-commerce with media convergence with neat example
3. Explain the anatomy of e-commerce applications
4. Explain the network infrastructure for e-commerce
5. Explain EDI software implementation
6. Explain the details for the dimensions of internal electronic commerce systems

Apply

1. Design an airline reservation System for checking out of seats in online based system. Factors to be looked that they are easy to find appropriate web sites and how easy are they to use and how easy it to find a best price is. The following sequence of steps is recommended:
   a) Select a trip and some dates. The trip should be substantial, Trans Atlantic or Trans Pacific and should be from one major airline hub to another (so that there will be several competing carriers); examples could be London or Frankfurt to New York or San Francisco to Tokyo. The dates should be two or three months hence for, say, Friday to Monday at Easter or 23 December to 02 January.
   b) Use the web to find out a list of carriers that fly the required route.
   c) Use the websites of two or three selected airlines to find the cheapest available return flight.
   d) Find an online travel agent and find the cheapest price from that agent for the requirements.

   For all stages of the above problems, make notes on the ease of use of the web sites. Note the price comparisons between the airlines and, in particular

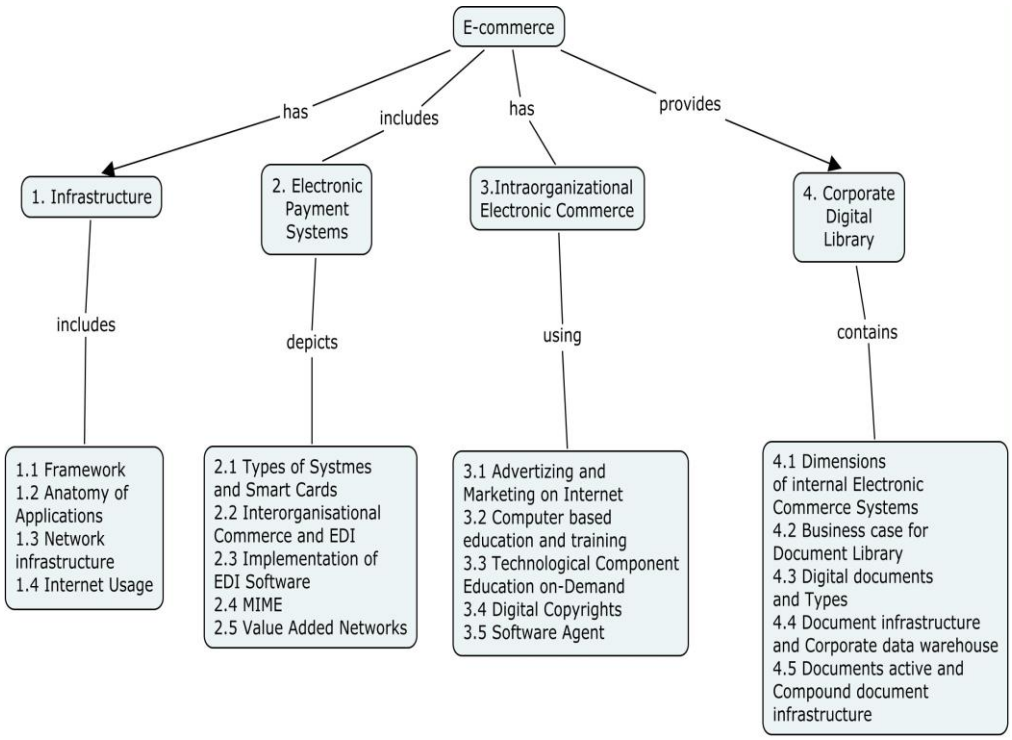
any rice differential between the airline site and the online travel agent. Compare notes on your bookings and of any fellow students.

2. Design an Interchange Agreement to cover the exchange of orders and invoices between Packaging Solutions and Pens and Things. You may use model agreements obtained from the web (or elsewhere) but the agreement must be fully tailored to the needs of these two organizations. Keep the agreement short, one page should suffice.
3. Design an plan for the implementation of EDI at Pens and Things. The plan is initially for the interchange of orders with Packaging Solutions but should take account of possible future developments with their assumptions

Analyze

1. In a web-based E-Commerce system, the VI is the web client and the web server is the main component of the service system. Justify your answer.
2. Explore the role and support of E-Commerce in the following applications:
   (i) Real estate business
   (ii) Insurance sector
   (iii) Jobs and employment sites
   (iv) Entertainment sites
   How much stronger is 128 bit key than 64 or 56 bit DES key?
3. Distinguish between directory and search engine. Which method produces a larger number of sites from which to choose?
4. Amazon.com is the example web-site for B2C type of E-Commerce. Defend your answer.

Concept Map



Syllabus

Introduction to E-commerce: Framework for e-commerce, Anatomy of e-commerce applications, Network Infrastructure for e-commerce, Internet using Infrastructure network. **Electronic Payment systems:** Types of electronic Payment Systems and Smart Cards, Inter organisational Commerce and EDI, Implementation of EDI software, MIME and Value added Networks. **Intra organizational Electronic Commerce:** Advertising and Marketing on Internet, Computer Based Education and Training, Technological Component of Education on-Demand, Digital Copyrights, Software Agent. **Corporate Digital Library:** Dimensions of Internal Electronic Commerce Systems, Making a business case for Document Library, Types of digital documents, Issues behind document infrastructure and corporate data warehouses, Document active and Compound document Infrastructure.

Text Book:

1. Kalkota and Whinston," Frontiers of Electronic Commerce", Pearson Edition, 2006.

Reference books:

1. Kamalesh K. Bajaj, "e-Commerce: The Cutting Edge & Business", Tata McGraw-Hill, 2003.
2. J. Christopher Westland, Theodre H K Clark, "Global Electronic Commerce-Theory and Case Studies" University Press, 2001.
3. P T Joseph, "E-Commerce – an Indian perspective", Prentice Hall, 2/e, 2005.
4. C S V Moorthy, "E-Commerce concepts, Models, Strategies", Himalaya Publications, 2/e, 2003.
5. Gari P Schneider, "Electronic Commerce", Thomson Course Technology, 4/e, 2004.
6. http://en.wikipedia.org/wiki/Electronic_commerce
7. http://cyber.law.harvard.edu/ecommerce/introduction.html

Course contents and Lecture Schedule

S. No	Topic	No. of Lectures
1	**Introduction to e-commerce**	
1.1	Framework for e-commerce	2
1.2	Anatomy of e-commerce applications	3
1.3	Network infrastructures for e-commerce	4
1.4	Internet using Infrastructure network	3
2	**Electronic Payment Systems**	

S. No	Topic	No. of Lectures
2.1	Types of Electronic Payment Systems and smart cards	2
2.2	Inter organizational Commerce and EDI	3
2.3	Implementation of EDI Software	3
2.4	MIME	1
2.5	Value Added Networks	1
2.6	Case Study based on one time password verification in bank sectors	1
3	**Intra organizational Electronic Commerce**	
3.1	Advertising and Marketing on Internet	3
3.2	Computer Based Education and Training	2
3.3	Technological Component of Education on-Demand	3
3.4	Digital Copyrights	1
3.5	Software Agent	2
4	**Corporate Digital Library**	
4.1	Dimensions of Internal Electronic Commerce Systems	2
4.2	Making a business case for Document Library	2
4.3	Digital documents and Types	2
4.4	Issues behind document infrastructure and corporate data warehouses	3
4.5	Documents active and Compound document infrastructure	2
	Total	**45**

Course designers

1. M.K.Kavitha Devi mkkdit@tce.edu
2. G.Prakash prakashguru@tce.edu

Sub Code	Lecture	Tutorial	Practical	Credit
TCM	3	0	--	3

TCM Software Modeling and Analysis 3:0

Preamble:

Software modeling and analysis is a software engineering approach that models a software system as a group of interacting objects. Each object represents some entity of interest in the system being modeled, and is characterized by its class, its state, and its behavior. Various models can be created to show the static structure, dynamic behavior, and run-time deployment of these collaborating objects. Software modeling and analysis applies modeling techniques to analyze the functional requirements for a system and elaborates the analysis models to produce implementation specifications. Software analysis focuses on what the system does and software modeling on how the system does it. The course is designed to enable a clear understanding and knowledge of the foundations, techniques, and methodologies in software modeling and analysis and its practice in the industry.

Program Outcomes addressed

a. Graduates will demonstrate an ability to apply knowledge of mathematics, science and engineering.

c. Graduates will demonstrate an ability to design a system or component or process to meet stated specifications.

e. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

Competencies

At the end of the course students will be able to

1. Understand the concepts of static modeling, design patterns related to software modeling.
2. Understand the purpose and significance of Software modeling and analysis in Software development life cycle.
3. Analyze the software requirements for various engineering applications through static and dynamic modeling.
4. Explain the various patterns such as creational, structural and behavioural used for software modeling.
5. Select the appropriate design pattern for engineering applications such as simulation of cell phone, Social networking site.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	End-semester examination
1	Remember	40	30	20
2	Understand	40	40	30
3	Apply	20	30	50
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	0	0	0

Course Level Learning Objectives

Remember

1. What is realization and collaboration?
2. What is meant by generalization set?
3. What is meant by ternary association?
4. What are gates? What are entry and exit gates?
5. What is an asynchronous message?
6. How is iteration over a collection depicted in sequence diagram?

Understand

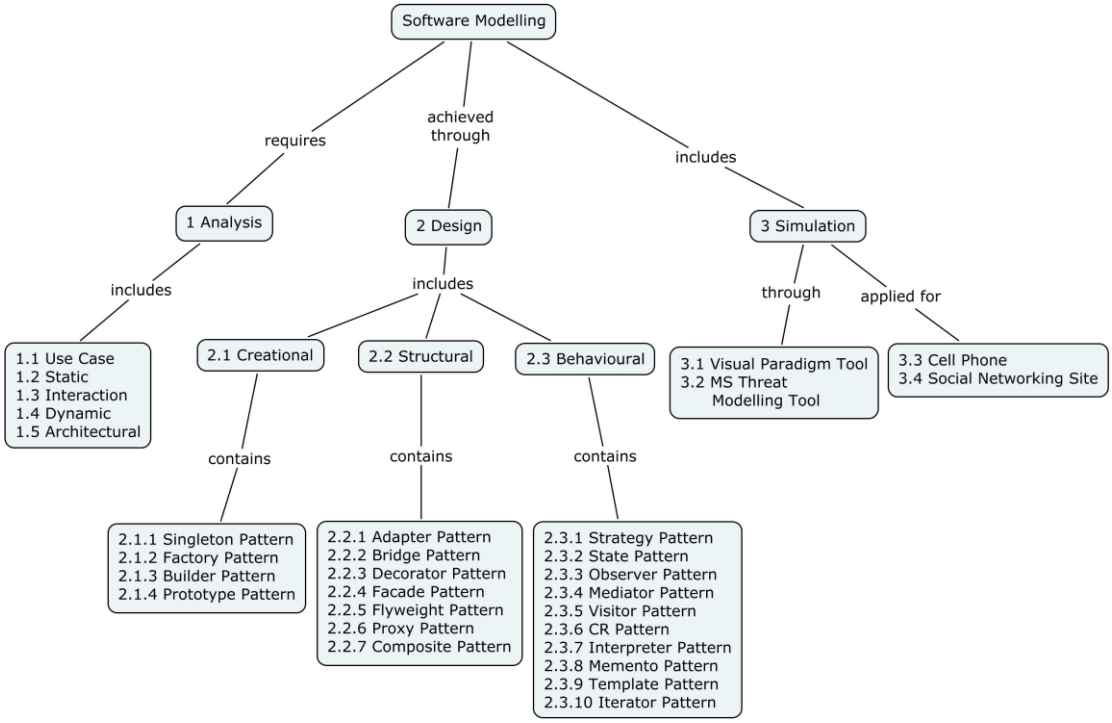
1. Which questions need to be deliberated upon while a) choosing system boundary b) finding actors c) finding use cases?
2. Explain the manner in which use cases can be used for the purpose of testing?
3. Explain the various types of subclass partitioning with the help of examples?
4. What is abstract class? In a class diagram how an abstract class is depicted?
5. What are communication diagram? How they are different from sequence diagram?
6. What is meant by recursive operation? Consider two java programs to find a) factorial of a number b) Fibonacci series, using recursion. Draw a sequence diagram depicting recursion?

Apply

1. Identify the use cases for the simulation of cell phone and draw Use case and Activity diagram?
2. Identify the Use cases for the social networking website and draw Activity and class diagram?

3. Consider a program to accept a Login name and password from the user of an email service. If both of them are Valid then the user must be directed to his inbox. If either of them is invalid he must get a message indicating it. Devise such scheme making assumptions about the collaboration objects and methods. Use the opt combination fragment of the sequence diagram?
4. Draw the state transition diagram and deployment diagram for college laboratory?
5. With your own examples, explain the various types of creational patterns?
6. Consider a bank that gives the statement of customers account at the email address specified by the customer, the moment any transaction is made. Simulate the working and explain the use of the observer pattern for this purpose?

Concept Map:



Syllabus

Analysis: Use Case - Static - Interaction - Dynamic - Architectural. **Design:** - **Creational** - Singleton Pattern, Factory Pattern, Builder Pattern, Prototype Pattern. **Structural** - Adapter Pattern, Bridge Pattern, Decorator Pattern, Facade Pattern, Flyweight Pattern, Proxy Pattern, Composite Pattern. **Behavioral** - Strategy Pattern, State Pattern, Observer Pattern, Mediator Pattern, Visitor Pattern, CR Pattern, Interpreter Pattern, Memento Pattern, Template Pattern,

Iterator Pattern. **Simulation** - Visual paradigm Tool – Microsoft Threat Modeling Tool - Cell Phone - Social Networking Site.

Textbook

1. Eric T Freeman, Elisabeth Robson, Bert Bates, Kathy Sierra, "Head First Design Patterns" O'Reilly Media, 2004.
2. Mahesh P. Matha, "Object-Oriented Analysis and design using UML", Prentice Hall of India, 2008.

References

1. Eric Gamma, Richard Helm, Ralph Johnson, John Vissides, Grady Booch, "Design Patterns ", Addison Wesley, 1995.
2. Craig Larman, "Applying UML and Patterns ", Prentice Hall, 1998.
3. Ali Bahrami, "Object Oriented System Development ", McGraw Hill International Edition, 1999.
4. William J Brown et al. " Anti-Patterns: Refactoring Software, Architectures and Projects in Crisis ", John Wiley, 1998.

Course Contents and Lecture Schedule:

S. No.	Topic	No. of Lectures
1	**Design Analysis**	
1.1	Use Case	1
1.2	Static	2
1.3	Interaction	2
1.4	Dynamic	2
1.5	Architectural	1
2	**Design Patterns**	
2.1	**Creational**	
2.1.1	Singleton Pattern	1
2.1.2	Factory Pattern	1
2.1.3	Builder Pattern	1
2.1.4	Prototype Pattern	1
2.1.5	State based testing	1
2.2	**Structural**	
2.2.1	Adapter Pattern	1
2.2.2	Bridge Pattern	1
2.2.3	Decorator Pattern	1
2.2.4	Facade Pattern	1
2.2.5	flyweight Pattern	1
2.2.6	Proxy Pattern	1

S. No.	Topic	No. of Lectures
2.2.7	Composite Pattern	1
2.3	**Behavioral**	
2.3.1	Strategy Pattern	1
2.3.2	State Pattern	1
2.3.3	Observer Pattern	1
2.3.4	Mediator Pattern	1
2.3.5	Visitor Pattern	1
2.3.6	CR Pattern	1
2.3.7	Interpreter Pattern	1
2.3.8	Memento Pattern	1
2.3.9	Template Pattern	1
2.3.10	Iterator Pattern	1
3	**Simulation**	
3.1	Visual paradigm Tool	2
3.2	Microsoft Threat Modeling Tool	2
3.3	Cell Phone	3
3.4	Social Networking Site	3
	Total	**40**

Course Designers:

1. S. Selvakumar ssit@tce.edu
2. P.Karthikeyan karthikit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
TCN	3	0	--	3

TCN Multimedia Mining 3:0

Preamble: Advances in multimedia acquisition and storage technology have led to tremendous growth in very large and detailed multimedia databases. These multimedia files are analyzed, useful information to users can be revealed. Multimedia mining deals with the extraction of implicit knowledge, multimedia data relationships, or other patterns not explicitly stored in multimedia files. Multimedia mining is more than just an extension of data mining, as it is an interdisciplinary endeavor that draws upon expertise in computer vision, multimedia processing, multimedia retrieval, data mining, machine learning, database and artificial intelligence. This course briefly describes the multimedia mining with applications to real time data sets

Program outcomes addressed:

a. Graduates will demonstrate an ability to apply the knowledge of engineering, mathematics and science
c. Graduates will demonstrate an ability to design a system or component or process to meet stated specifications.
e. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.
k. Graduates will demonstrate an ability to use techniques, skills, and modern engineering tools necessary for engineering practice.

Competencies:

At the end of the course, the students will be able to

1. Understand the various steps under Multimedia Mining.
2. Perform data preprocessing on Multimedia datasets.
3. Perform attribute selection on Multimedia datasets.
4. Perform association rule mining on Multimedia datasets.
5. Perform classification on Multimedia datasets.
6. Perform clustering on Multimedia datasets.
7. Familiarize with Multimedia mining tools for solving practical problems

Assessment Pattern:

	Bloom's Category	Test 1	Test 2	Test 3/End-semester examination
1	Remember	20	10	10
2	Understand	40	30	20
3	Apply	40	50	50
4	Analyze	0	10	20
5	Evaluate	0	0	0
6	Create	0	0	0

Course Level Learning Objectives
Remember

1. What is Multimedia mining?
2. List the various Multimedia mining techniques.
3. What is the need for image cleaning?
4. What are the techniques used for audio pre-processing?
5. What are the techniques used for video pre-processing?
6. What is correlation analysis?

Understand

1. Differentiate audio mining and video mining.
2. Differentiate text mining and image mining.
3. Distinguish between multimedia mining of structured data and unstructured data.
4. Discuss the working principles of various clustering algorithms in multimedia mining.
5. Describe the ID3 algorithm for audio mining.

Apply

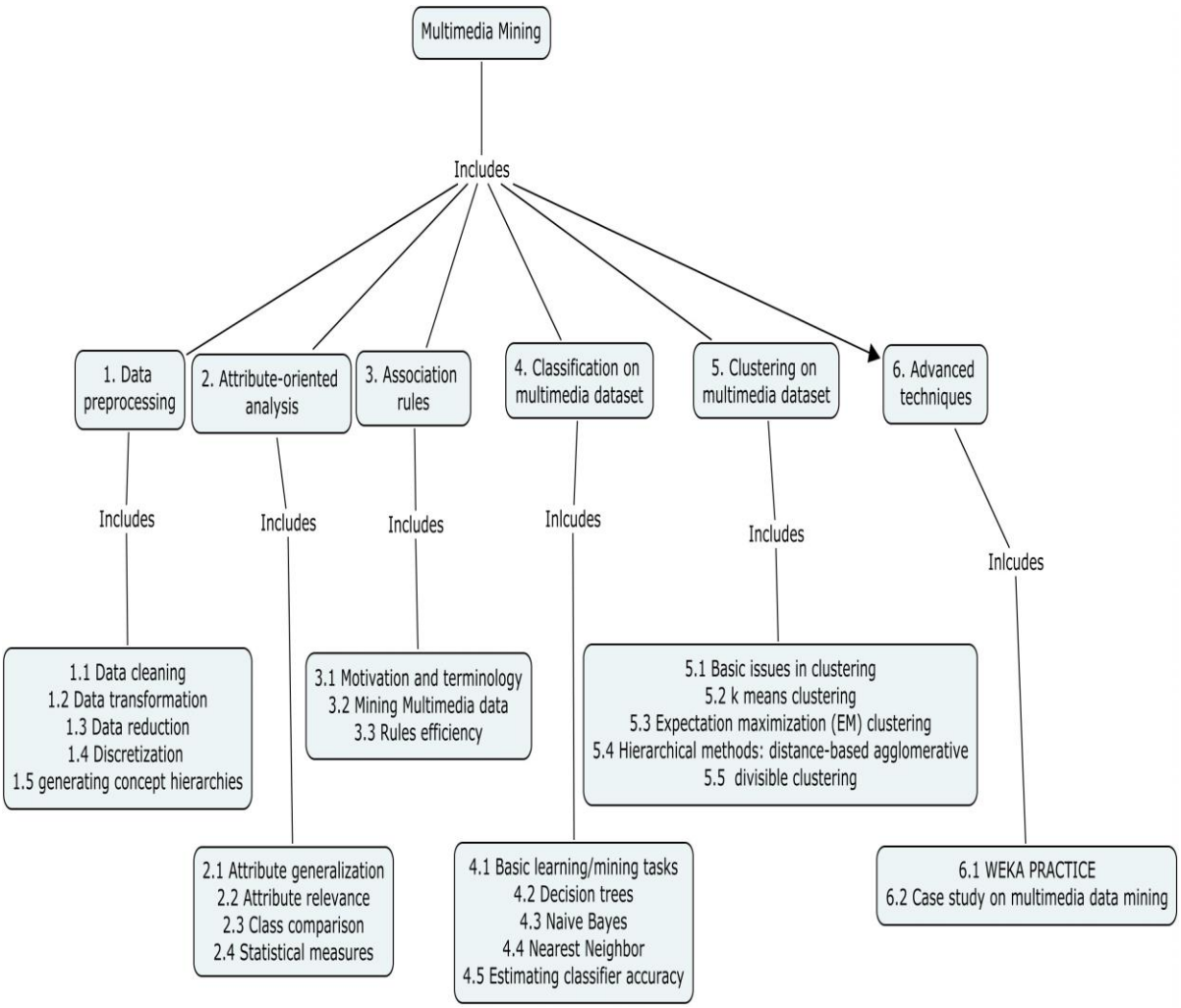
1. Apply association rule mining to the audio dataset of your own.
2. Apply the decision tree algorithms to image data set of your own.
3. Apply NB algorithm on image data set of your own.
4. Apply the k-means clustering to any multimedia mining applications
5. Apply DT and NB classification algorithms on multimedia datasets and report the performance of each algorithm.

Analyze

1. Use the NB and J48 learning schemes, with the default settings to analyze the multimedia data. Use training set as the test option and report model percent error rate.

2. The a priori algorithm for generating association rules has many command line options. How do you modify these? What do the options mean? Can you modify the options?

Concept Map:



Syllabus:

Introduction to Multimedia Mining –Multimedia mining, Multimedia Mining Goals, Stages of the Multimedia mining Process, Multimedia Mining Techniques, Knowledge Representation Methods, Indexing techniques for Image and Video, Content Based Image and Video Retrieval. **Data preprocessing** - Data cleaning, Data transformation, Data reduction, Discretization and generating concept hierarchies. **Attribute-oriented analysis**-Attribute generalization, Attribute relevance, Class comparison, Statistical measures. **Association rules** -Motivation

and terminology, mining Multimedia data, and rules efficiency. **Classification on multimedia dataset** - Basic learning/mining tasks, Decision trees, Naive Bayes, Nearest Neighbor, Estimating classifier accuracy (holdout, cross-validation, leave-one-out). **Clustering on multimedia dataset** - Basic issues in clustering, k means clustering, Expectation maximization (EM) clustering, Hierarchical methods: distance-based agglomerative and divisible clustering, Conceptual clustering: Cobweb. **Advanced techniques -** WEKA PRACTICE, Case study on multimedia data mining.

Text Books:

1. Zhongfei Zhang, Binghamton and Ruofei Zhang, "Multimedia Data Mining: A Systematic Introduction to Concepts and Theory" CRC Press, 2008.
2. Bhavani Thuraisingham, "Managing and Mining Multimedia Databases," CRC Press 2009

Reference Books:

1. Chabane Djeraba, "Multimedia Mining: A Highway to Intelligent Multimedia Documents (Multimedia Systems and Applications), Kluwer Academic Publisher, USA, 2007.
2. Multimedia and Data Mining Research Group of the DFKI. (http://madm.dfki.de/home)
3. Data Mining Tools. (http://data-mining.safe-install.com/)
4. Mining Image Datasets and Intelligent Image Database Searching System. (http://vision.ece.ucsb.edu/~jelena/research/mining.html)

Course contents and Lecture Schedule:

S.No.	Topic	No. of Lectures
0	**Introduction to Data Mining**	
	Multimedia mining, Multimedia Mining Goals, Stages of the Multimedia mining Process	2
	Multimedia Mining Techniques	1
	Knowledge Representation Methods	1
	Indexing techniques for Image and Video	1
	Content Based Image and Video Retrieval	1
1	**Data preprocessing for multimedia dataset**	
1.1	Data cleaning	2

S.No.	Topic	No. of Lectures
1.2	Data transformation	2
1.3	Data reduction	1
1.4	Discretization	2
1.5	Generating concept hierarchies	2
2	**Attribute-oriented analysis for multimedia dataset**	
2.1	Attribute generalization	2
2.2	Attribute relevance	2
2.3	Class comparison, Statistical measures	2
3	**Association rules for multimedia dataset**	
3.1	Motivation and terminology, mining Multimedia data	2
3.2	rules efficiency	2
4	**Classification on multimedia dataset**	
4.1	Basic learning, mining tasks	1
4.2	Decision trees	2
4.3	Naive Bayes	2
4.4	Nearest Neighbor	2
4.5	Estimating classifier accuracy (holdout, cross-validation, leave-one-out)	2
5	**Clustering on multimedia dataset**	
5.1	Basic issues in clustering, k-means clustering	2
5.2	Expectation maximization (EM) clustering	2
5.3	Hierarchical methods: Distance-based agglomerative and divisible clustering	2
5.4	Conceptual clustering: Cobweb	2
6	**Advanced techniques**	
6.1	WEKA PRACTICE	2
6.2	Case study on multimedia data mining	1
	Total	**45**

Course Designers:

1. S.Appavu alias Balamurugan sbitit@tce.edu
2. M. Amutha Prabakar ap_sse@tce.edu

Sub Code	Lecture	Tutorial	Practical	Credit
TCO	3	--	--	3

TCO Machine Learning 3:0

Preamble

The field of machine learning is concerned with the question of how to construct computer programs that improve automatically with experience. In recent years, many successful applications of machine learning have been developed, ranging from data-mining programs that learn to detect fraudulent credit card transactions, to autonomous vehicles that learn to drive on public highways. At the same time, there have been important advances in the theory and algorithms that form the foundation of this field. The goal of this course is to provide an overview of the state-of-art algorithms used in machine learning. This course will cover both the theoretical properties of these algorithms and their practical applications.

Program outcomes addressed

a. Graduates will demonstrate an ability to apply the knowledge of engineering, mathematics and science.

c. Graduates will demonstrate an ability to design a system or component or process to meet stated specifications.

e. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

k. Graduates will demonstrate an ability to use techniques, skills, and modern engineering tools necessary for engineering practices.

Competencies

At the end of the course, Students will be able to

1. Imbibe the fundamental concepts of Machine Learning.
2. Understand the different classification algorithms and How it will apply in to classification oriented problems.
3. Based on these various classification algorithms understand the merits, demerits and how to select suitable classification algorithm for particular application.
4. understand unsupervised learning techniques and the applications of machine learning techniques.
5. Gain the experience of doing independent study and research.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3 / End-semester examination
1	Remember	40	20	20
2	Understand	40	30	30
3	Apply	20	50	50
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	0	0	0

Course Level Learning Objectives

Remember

1. What is "Machine Learning"? Give examples of learning machines.
2. What is supervised learning?
3. What is unsupervised learning? Give examples of unsupervised learning tasks.
4. What is "generalization"?
5. What is "over-fitting"?
6. What are training/validation/test sets? What is "cross-validation"? Name one or two examples of cross-validation methods

Understand

1. How does the candidate elimination algorithm work? What are its limitations?
2. Describe a strategy that can be used to avoid over-fitting in decision trees.
3. How can a decision tree be converted into a rule set? Illustrate with an example.
4. Give the rules for (a) the optimal Bayesian hypothesis; (b) the maximum likelihood hypothesis. When are these same?
5. Describe the Naive Bayesian method of classification. What assumptions does this method make about the attributes and the classification? Give an example where this assumption is not justified. What is the Laplacian correction, and why is it necessary?
6. Describe the nearest neighbor's algorithm.

Apply

1. Select the appropriate classifier algorithm for the following problem. You run your classifier over a randomly selected test set of 1024 instances, and find that it gives the correct answer three-quarters of the time. Give the formula for the 95% confidence interval for the true accuracy of the classifier. Apply the hypothesis test to test the significant level is sufficient for choosing the null hypothesis. Further, if you increase the size of the test set by a factor of 16 (i.e. use a test set of 16,384 instances), what happens to the confidence interval?

2. A learning theory may be divided into the following parts:

(1) A hypothesis space;

(2) A representation for hypotheses

(3) A preference criterion over hypotheses, independent of the data

(4) A measure of how well a given hypothesis fits given data.

(5) A search strategy to find a good hypothesis for a given data set.

For the machine learning method of your choice, explain what each of these is. (For certain methods, some of these are degenerate.)

3. Consider a training set consisting of the following eight examples:

Examples labeled "0"	Examples labeled "1"
3,3,0	2,2,0
3,3,1	1,1,1
3,3,0	1,1,0
2,2,1	1,1,1

The questions below pertain to various feature selection methods that we could use with the logistic regression model.

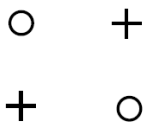
a. What is the mutual information between the third feature and the target label based on the training set?

b. Which feature(s) would a filter feature selection method choose? You can assume here that the mutual information criterion is evaluated between a single feature and the label.

c. Which two feature(s) would a greedy wrapper process choose?

d. Which features would a regularization approach with a 1-norm penalty $\sum_{i=1} |w_i|$ choose? Explain briefly.

4. Consider the following data set:



Circle all of the classifiers that will achieve zero training error on this data set. (You may circle more than one.)

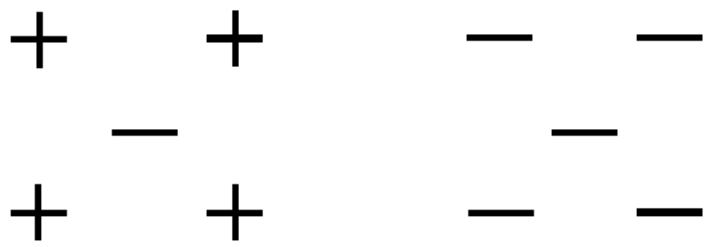
a. Logistic regression

b. SVM (quadratic kernel)

c. Depth-2 ID3 decision trees

d. 3-NN classifier

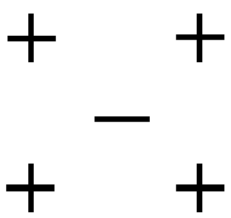
5. For the following dataset, circle the classifier which has larger Leave-One-Out Cross-validation error.



a. 1-NN

b. 3-NN

6. i. Consider training a boosting classifier using decision stumps on the following data set:



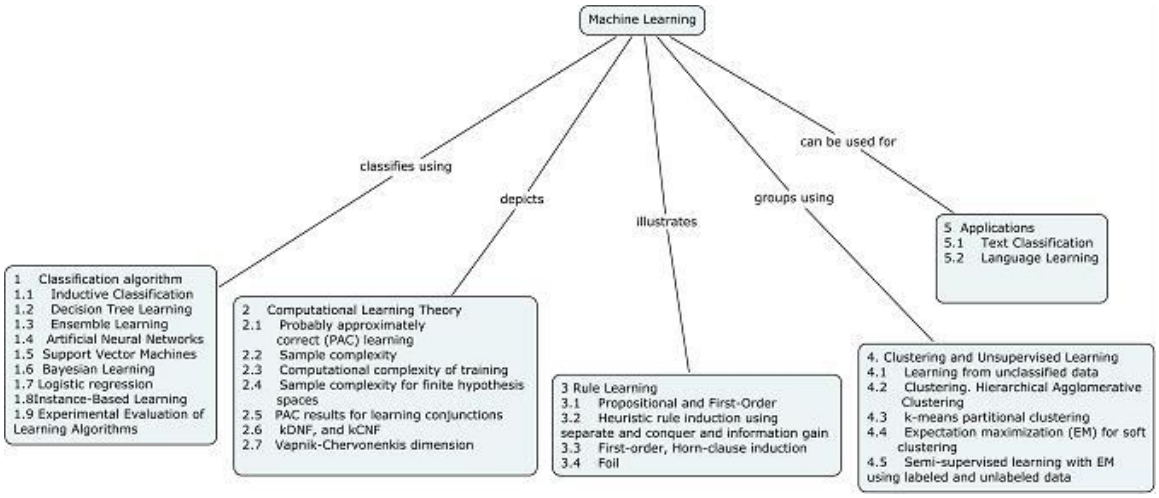
a. Which examples will have their weights increased at the end of the first iteration? Circle them.

b. How many iterations will it take to achieve zero training error? Explain.

c. Can you add one more example to the training set so that boosting will achieve zero training error in two steps? If not, explain why.

ii. Why do we want to use "weak" learners when boosting?

Concept Map



Syllabus

Introduction to the goals and applications of machine learning systems. **Classification Algorithms:** Inductive Classification, Decision Tree Learning , Ensemble Learning, Artificial Neural Networks, Support Vector Machines, Bayesian Learning, Logistic regression, Instance-Based Learning, and Experimental Evaluation of Learning Algorithms. **Computational Learning Theory**: probably approximately correct (PAC) learning. Sample complexity, Computational complexity of training. Sample complexity for finite hypothesis spaces. PAC results for learning conjunctions, kDNF, and kCNF, Vapnik-Chervonenkis dimension **Rule Learning** Propositional and First-Order, Heuristic rule induction using separate and conquer and information gain. First-order, Horn-clause induction and Foil **Clustering and Unsupervised Learning-**Learning from unclassified data. Clustering. Hierarchical Agglomerative Clustering. k-means partitional clustering. Expectation maximization (EM) for soft clustering. Semi-supervised learning with EM using labeled and unlabeled data, **Applications**: Text Classification, Language Learning

Textbooks

1. Tom Mitchell, Machine Learning, Mc Graw Hill, 1997.
2. Bishop, C. Pattern Recognition and Machine Learning. Berlin: Springer-Verlag, 2006.

References

1. Machine Learning at AAAI. (http://www.aaai.org/Pathfinder/html/machine.html)
2. Journal of Machine Learning Research (http://jmlr.org/)
3. mdl-research.org (http://www.mdl-research.org/)
4. Machine Learning Database Repository at UC Irvine (http://www.ics.uci.edu/~mlearn/MLRepository.html)
5. David Aha's list of machine learning resources (http://home.earthlink.net/~dwaha/research/machine-learning.html#home)
6. Avrim Blum's Machine Learning Page (http://www.cs.cmu.edu/~avrim/ML/)
7. UCI - Machine Learning Repository (http://archive.ics.uci.edu/ml/)
8. UTCS Machine Learning Research Group (http://www.cs.utexas.edu/users/ml)
9. Microsoft Bayesian Network Editor (MSBNx) (http://www.research.microsoft.com/adapt/MSBNx/)
10. Weka 3 -- Machine Learning Software in Java (http://www.cs.waikato.ac.nz/~ml/weka/index.html)
11. Journal of AI Research (online text) (http://www.jair.org/)
12. MLC++, A Machine Learning Library in C++ (http://www.sgi.com/Technology/mlc)

Course Contents and Lecture Schedule

S. No.	Topic	No. of Lectures
0	Introduction to the goals and applications of machine learning and aspects of developing a learning system	1
1	**Classification Algorithms**	
1.1	Inductive Classification	1
1.2	Decision Tree Learning	2
1.3	Ensemble Learning	2
1.4	Artificial Neural Networks	2
1.5	Support Vector Machines	2
1.6	Bayesian Learning	2
1.7	Logistic regression	2
1.8	Instance-Based Learning	2
1.9	Experimental Evaluation of Learning Algorithms	1
2	**Computational Learning Theory**	
2.1	Probably approximately correct (PAC) learning	1
2.2	Sample complexity	1
2.3	Computational complexity of training	1
2.4	Sample complexity for finite hypothesis spaces	1
2.5	PAC results for learning conjunctions	1
2.6	kDNF, and kCNF	1
2.7	Vapnik-Chervonenkis dimension	1
3	**Rule Learning**	
3.1	Propositional and First-Order	1
3.2	Heuristic rule induction using separate and conquer and information gain	1
3.3	First-order, Horn-clause induction	1
3.4	Foil	1
4	**Clustering and Unsupervised Learning**	
4.1	Learning from unclassified data	1
4.2	Clustering. Hierarchical Agglomerative Clustering	2
4.3	k-means partitional clustering	2
4.4	Expectation maximization (EM) for soft clustering	2
4.5	Semi-supervised learning with EM using labeled and unlabeled data	2
5	**Applications**	

S. No.	Topic	No. of Lectures
5.1	Text Classification	2
5.2	Language Learning	2
	Total	**41**

Course Designers

1. M.Indra Devi midit@tce.edu
2. P.Venkata Krishnan pvkit@tce.edu

Sub Code	Lecture	Tutorial	Practical	Credit
TCP	3	0	--	3

TCP Knowledge Based Systems 3:0

Preamble

Knowledge Based Systems are Artificial Intelligence based tools that work on knowledge base for effective decision making in more human oriented way using the knowledge stored in it. It uses artificial intelligence or expert system techniques in problem solving processes. It incorporates a store of expert knowledge with couplings and linkages designed to facilitate its retrieval in response to specific queries, or to transfer expertise from one domain of knowledge to another.

Program Outcomes addressed

c. Graduates will demonstrate an ability to design a system, component or process to meet stated specifications.

e. Graduates will demonstrate an ability to identify, formulate, and solve knowledge engineering problems.

k. Graduates will demonstrate an ability to use techniques, skills, and modern engineering tools necessary for engineering practice.

Competencies

At the end of the Course, Students will be able to

1. Understand the application of Knowledge-Based Systems for Integrated Development.

2. Understand the application of Expert System Model for On-Line Fault Diagnosis.

3. Familiarize themselves with Conceptual Modeling for Knowledge Based Systems.

4. Explain the way of developing Knowledge Based Intelligent Tutoring-System.

5. Know how and when it makes sense to apply Knowledge-based methods in a practical environment.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3/ End-semester examination
1	Remember	20	20	20
2	Understand	30	30	30
3	Apply	50	50	50
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	0	0	0

Course Level Learning Objectives

Remember

1. What is the use of knowledge based system?
2. List the two major elements of a knowledge based system.
3. List measures of knowledge representation.
4. List the five types of knowledge based systems.
5. What are the major limitations of Knowledge-based systems?
6. List examples of knowledge representation scheme.

Understand

1. Describe the various activities in the development of a Knowledge-Based System.
2. Describe the various roles of Knowledge Representation.
3. Describe the various Research Areas in Pure KBS development.
4. Describe the Challenges of Knowledge representation.
5. Describe the types of knowledge used by the various diagnostic approaches.
6. Describe Guidelines for Conceptual Modelling.

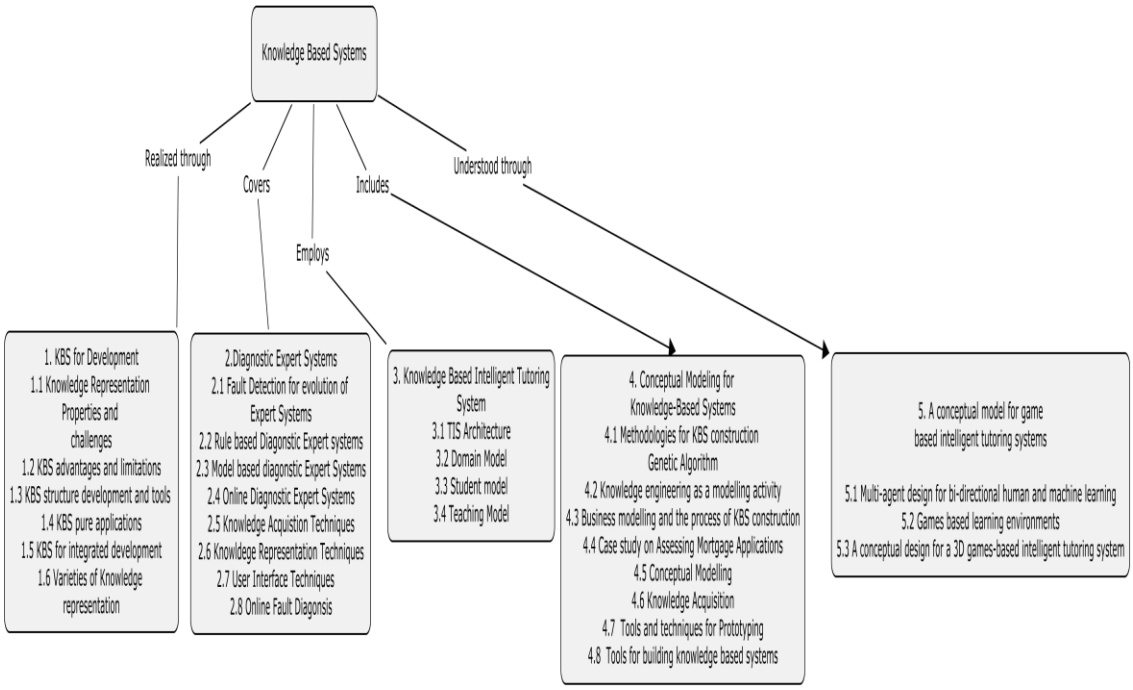
Apply

1. Consider the case study on Health Improvement and apply the various dimensions of Knowledge Based System for Integrated Development.
2. Consider the 3D games-based learning intelligent tutoring system and apply multi-agent and machine learning technologies for conceptual design.
3. Consider an application of an expert system model for online fault diagnosis to detect faults in hydraulic systems. This process is performed on-line and the system is able to respond to dynamically changing states by combining modeling information, on-line sensor measurements and symbolic data. The diagnosis of faults is performed through the knowledge base of the expert system. Apply a knowledge base architecture that permits the

interaction of sensor information, modeling information and experiential knowledge symbolic representation.

4. A client applies for the mortgage, the bank will assess the application and the client will be informed whether the loan is granted or not. The mortgage application forms the interface between the client (filling in the application form, answering questions) and the bank (processing, assessing and finalizing the application). The post condition indicates that the end result of the action is that the client is notified of the result, and that the result is a consequent of applying the business rules (financial as well as legal and policy knowledge) that the bank uses in assessing this kind of applications. The "assess" task will be marked as knowledge intensive, meaning that business-critical knowledge (the business rules) is required to perform the task. In a glossary, all terms (processes, types, attributes, conditions, etc.) needs to be defined. Assess a mortgage application with Input application as well as business rules as norms and the Output is the 'eligible' decision.
5. The goal of an assessment task is to evaluate a particular case against some norms or criteria and take a decision. In the example, a decision needs to be made whether to accept a mortgage application. The input knowledge role consists of data about the case, i.e., the mortgage application. The static knowledge role consists of norms or criteria which are used to evaluate the case in order to take a decision. These criteria include for example business rules that relate income to the amount requested. The output knowledge rule consists of the decision, eligible for a mortgage, yes or no. Identify the various attributes needed to describe a client, mortgage as well as a mortgage application through domain knowledge modeling along with knowledge bases containing example rules sets for solving the assessment problem.
6. Consider a Generic Assessment task for a mortgage application. The assessment action implements the assessment responsibility of the assessment system, explicitly making use of knowledge. Show the various inferences through Generic inference diagram for an assessment task and Activity diagram defining a control structure for data driven inferences using the actions specified in the inference diagram.

Concept Map:



Syllabus

Knowledge-Based Systems for Development: KBS Structure- KBS Advantages and Limitations, KBS Development - KBS Tools - KBS Advantages and Limitations- KBS Pure Applications - KBS for Integrated Development - Knowledge Representations – Properties and Challenges-Varieties of Knowledge Representation **Diagnostic Expert Systems:** Evolution of Expert Systems Technology For Fault Detection-Rule-Based Diagnostic Expert Systems-Model-based diagnostic expert systems-On-line diagnostic expert systems-Evolution of Knowledge Acquisition Techniques-Evolution of Knowledge Representation Techniques-Evolution Of User Interface Techniques For Expert Systems-An Expert System Model for On-Line Fault Diagnosis. **Development of Knowledge Based Intelligent Tutoring-System:** Architecture of the ITS-Domain Model-Student Models-Teaching Model **Conceptual Modeling for Knowledge Based Systems:** Methodologies for KBS construction-

Knowledge engineering as a modelling activity-Business modelling and the process of KBS construction-Business modelling-Case study on Assessing Mortgage Applications

Conceptual Modelling-Guidelines for Conceptual Modelling-Knowledge Acquisition Tools and techniques for Prototyping-Tools for building knowledge based systems.

A conceptual model for game based intelligent tutoring systems: Multi-agent design for bi-directional human and machine learning-Games based

learning environments-A conceptual design for a 3D games-based intelligent tutoring system.

Textbook

1. Advanced Knowledge Based Systems: Models, Applications & Research, Rajendra Akerkar and Priti Sajja, Techno mathematics Research Foundation publishers, India, 2009.

References

1. Rajendra Akerkar and Priti Sajja, "Knowledge-Based Systems", Jones & Bartlett Publishers, Sudbury, MA, USA, 2009.

2. Frank van Harmelen, Vladimir Lifschitz and Bruce Porter (Eds.), **"**Handbook of Knowledge Representation**"**, Elsevier, 2008.

3. Mills, C. & Dalgarno, B., "A conceptual model for game-based intelligent tutoring systems", In ICT: Providing choices for learners and learning. Proceedings ascilite Singapore 2007.

4. P-H. Speel, A. Th. Schreiber , W. van Joolingen , G. van Heijst, G.J. Beijer "Conceptual Modelling for Knowledge-Based Systems", Encyclopedia of Computer Science and Technology, Marce Dekker Inc., NewYork.

Course Contents and Lecture Schedule:

S.No.	Topic	No.of Lectures
1	**Knowledge-Based Systems for Development**	
1.1	KBS Structure, KBS Advantages and Limitations	2
1.2	KBS Development, KBS Tools	2
1.3	KBS Advantages and Limitations, KBS Pure Applications, KBS for Integrated Development	2
1.4	Knowledge Representations – Properties And Challenges	2
1.5	Varieties of Knowledge Representation	1
2	**Diagnostic Expert Systems**	
2.1	Evolution of Expert Systems Technology For Fault Detection	1
2.2	Rule-Based Diagnostic Expert Systems	1
2.3	Model-based diagnostic expert systems	1
2.4	On-line diagnostic expert systems	1
2.5	Evolution of Knowledge Acquisition Techniques	1
2.6	Evolution of Knowledge Representation Techniques	1
2.7	Evolution Of User Interface Techniques For Expert Systems	1

S.No.	Topic	No.of Lectures
2.8	An Expert System Model for On-Line Fault Diagnosis	2
3	**Development of Knowledge Based Intelligent Tutoring System**	
3.1	Architecture of The ITS	2
3.2	Domain Model	2
3.3	Student Models	2
3.4	Teaching Model	2
4	**Conceptual Modeling for Knowledge Based Systems**	
4.1	Methodologies for KBS construction	1
4.2	Knowledge engineering as a modelling activity	1
4.3	Business modelling	1
4.4	Business modelling and the process of KBS construction	1
4.5	Case study on Assessing Mortgage Applications	1
4.6	Conceptual Modelling	1
4.7	Guidelines for Conceptual Modelling	1
4.8	Knowledge Acquisition	1
4.9	Tools and techniques for Prototyping	1
4.10	Tools for building knowledge based systems	1
5	**A conceptual model for game based intelligent tutoring systems**	
5.1	Multi-agent design for bi-directional human and machine learning	3
5.2	Games based learning environments	3
5.3	A conceptual design for a 3D games-based intelligent tutoring system	3
	Total	**45**

Course Designer:

1. S.Senthamarai Kannan sskit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credits
TCQ	3	0	-	3

TCQ Bioinformatics 3:0

Preamble

Bioinformatics is a newly emerging branch in the field of life science. It is information technology applied to the management and analysis of biological data with the aid of computers. It is the science of using information to understand biology. It is a field in which biological information collected, compared, studied and analyzed to find the interrelation between them for solving structural, functional and evolutionary problems using computational technologies. The biological information stored in various databases is available online through internet. Bioinformatics refers to the creation and advancement of databases, software, computational and statistical techniques and theory to solve problems generated from the management and analysis of biological data.

Program Outcomes addressed

c. Graduates will demonstrate an ability to design a system, component or process as per needs and specifications.
d. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

Competencies

At the end of the course, the students will be able to

1. Understand how information is represented and transmitted in biological systems, starting at the molecular level.
2. Analyze the search engine operations related to the major online bioinformatics resources.
3. Apply variety of simulation techniques, in the context of computer modeling events from drug-protein interactions and probable protein folding configurations to the analysis of potential biological pathways.
4. Understand the Computational Methods for Biological Modeling and Simulation.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3/End-semester examination
1	Remember	30	20	10
2	Understand	50	30	30
3	Apply	20	30	30
4	Analyze	0	20	30
5	Evaluate	0	0	0
6	Create	0	0	0

Course Level Learning Objectives

Remember

1. What is bioinformatics?
2. What is Computational Biology?
3. Describe features of DNA sequence analysis.
4. Define about the search process.
5. Define the importance of bioinformatics.
6. List the search engine technologies.

Understand

1. List out the different kinds of DNA sequences.
2. If you had 1,500 base pair pieces of random DNA and you wanted to know how many of them had homology to known genes, what would you do to determine that?
3. What technique is used to measure the number of copies of a gene or an RNA molecule in human tissues?
4. What are the limitations of blotting techniques and what alternatives can you suggest?
5. Can you give me an example of a project you were involved with that illustrates your interest and skills in bringing people together?
6. How would you value a biotech company as opposed to a consumer products company?

Apply

1. Calculate the score of the DNA sequence alignment shown below using the following scoring rules: +1 for a match, -2 for a mismatch, -3 for opening a gap, and -1 for each position in the gap.

```
AACCTGTTGTGTACGGCTCG
|||| |||| |||| |||
AACC---TGTGAACGGATCG
```

2. Apply the Needleman and Wunsch algorithms for global alignment.

3. Apply dynamic programming algorithms for multiple pairwise sequence alignment problems.
4. Apply the Bayesian methods to estimate the evolutionary distance between DNA sequences.
5. Apply the PAM250 algorithm(point--accepted-mutation) and derive the substitution matrix for the following data:

   (...)

   T 3

   W -5 17

   Y -3 0 10

   V 0 -6 -2 4

   T W Y V
6. The program BLAST for database sequence similarity search is based on an idea of searching for "words", that is, stretches of matching residues in sequences. How do mismatches and small gaps appear in BLAST search, for instance like in a fragment of BLAST hit shown below?

   gatgacgagctgg-tcgggctccgcac

   ||||||||| | ||||||||||||

   Gatgacgagtcgactcgggctccgcac

Analyze

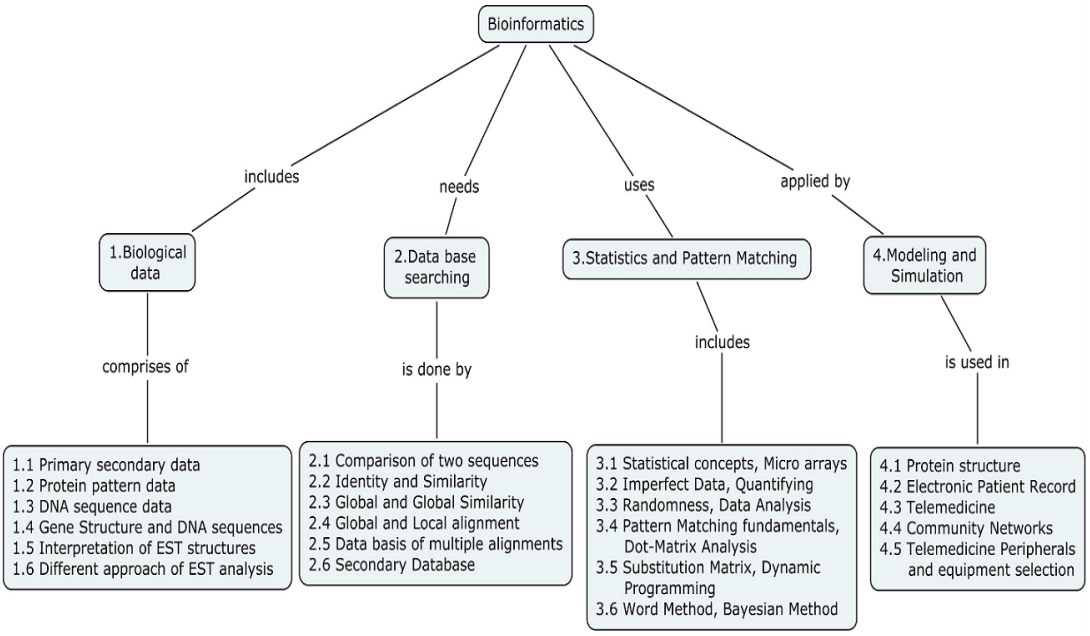
1. Illustrate the idea behind "weighting" sequences in the package CLUSTAL W for multiple alignments? How are these weights calculated?
2. If a match from a sequence database search is reported to have an E-value of 0.0, should it be considered highly insignificant or highly significant?
3. Using a gap penalty of 2 (i.e., deducting 2 from the score for each gap), find the best alignment of AWAP and APP using a Needleman-Wunsch dynamic programming approach. Show your working (i.e., draw and fill in the S-W matrix).
4. lf your original sequence was 400 residues long, and the sequence returned from the database was 420 residues long, how many times would we expect to see a HSP scoring as highly as this one:

   Hit from database: NFSSSQ

   User sequence: NFSTSQ

   In a typical alignment of two general sequences.
5. Compare the various heuristic methods employed for database searching and how will use these methods for database searches.
6. The program BLASTP (protein-protein database search) detects the presence of putative conserved motifs in a query sequence, prior to the output of sequence similarity hits. What kind of algorithm and database can be used for such an efficient search?

Concept Map:



Syllabus

Introduction - Watson's Definition, Top down Vs Bottom up Approach, Information Flow, Convergence, Data Bases, Data Life Cycle, Database Technology, Interfaces, Implementation. **Biological data basics** – Primary secondary data Basics, Protein pattern data Basics, DNA sequence data Basics, Genes Structure and DNA sequences, Interpretation of EST structures, Different approach to EST analysis. **Database Searching** – Comparison of Two sequences, Identity and similarity, Global and Global similarity, Global and Local Alignment, Data basis of multiple alignments, Secondary Database. **Statistics and Pattern Matching** – Statistical concepts, Micro arrays, Imperfect Data, Quantifying, Randomness, Data Analysis, Pattern Matching fundamentals, Dot-Matrix Analysis, Substitution Matrix, Dynamic Programming, Word Method, Bayesian Method. **Modeling & Simulation** – Protein Structure, Electronic Patient record, Telemedicine, Community Networks, Telemedicine Peripherals and equipment selection.

Textbook

1. Bryan Bergeron, M.D – Bioinformatics Computing – Second Edition, Pearson Education, 2003.

References

1. T.K. Attwood and D.J Perry – Smith, Introduction to Bioinformatics, Long man, Essex.1999.
2. Coiera E. Guide to Medical informatics, The internet and Telemedicine, Chajsman & Hall Medical, London 1997.
3. Bernser, E.S. Clinical Decision support systems, Theory and Practice, Springer – Verlag, New York, 1999.

Course contents and Lecture Schedule

S.No.	Topic	No of Lectures
0.	**Introduction**- Watson's Definition, Top down Vs Bottom up Approach, Information Flow, Convergence, Data Bases, Data Life Cycle, Database Technology, Interfaces, Implementation.	7
1.	**Biological data basics**	
1.1	Primary secondary data Basics	1
1.2	Protein pattern data Basics	1
1.3	DNA sequence data Basics	1
1.4	Genes Structure and DNA sequences	2
1.5	Interpretation of EST structures	2
1.6	Different approach to EST analysis.	2
2.	**Database Searching**	
2.1	Comparison of Two sequences - Identity and similarity	2
2.2	Global and Global similarity	2
2.3	Global and Local Alignment	2
2.4	Data basis of multiple alignments – Secondary Database	2
3.	**Statistics and Pattern Matching**	
3.1	Statistical concepts, Micro arrays	3
3.2	Imperfect Data, Quantifying	2
3.3	Randomness, Data Analysis	2
3.4	Pattern Matching fundamentals, Dot-Matrix Analysis	2
3.5	Substitution Matrix, Dynamic Programming	2
3.6	Word Method, Bayesian Method	2
4.	**Modeling & Simulation**	
4.1	Protein Structure – Electronic Patient record	2
4.2	Telemedicine	2
4.3	Community Networks	2
4.4	Telemedicine Peripherals and equipment selection	2
	Total	**45**

Course Designers

1. S.Gilbert Nancy gilbertnancyl@tce.edu

2. K.V.Uma kvuit@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
TCR	3	0	-	3

TCR Grid Computing 3:0

(Common to CSE: CCF)

Preamble: Grid Computing is based on an open set of standards and protocols that enable communication across heterogeneous, geographically dispersed environments.

Program Outcomes addressed

b. Graduates will demonstrate an ability to identify, formulate and solve complex engineering problems.

e. Graduates will demonstrate an ability to use techniques, skills, and modern grid computing tools to implement and organize engineering works under given constraints.

Competencies:

At the end of the course, the students will be able to

1. Compare various grid computing techniques.
2. Design an environment for Grid Computing.
3. Determine the various scheduling Techniques.
4. Providing Grid based solutions.
5. Simulate a Scheduler.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3/ End-semester examination
1	Remember	20	20	20
2	Understand	20	10	10
3	Apply	40	50	50
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	20	20	20

Course level Learning Objectives

Remember

1. What are the software and Hardware requirements to create a Grid?
2. What is the significance of SOAP?
3. What is the role of XML in Grid Computing?
4. List out the various methods used for Scheduling.
5. Give a note on Resource information
6. Define Grid Security.

Understand

1. Explain the various Blocks in the architecture of a Grid.
2. Explain the various features of SOAP Techniques.
3. Explain how web architecture is framed.
4. Discuss the role of a Meta Scheduler.
5. Discuss strengths and limitations of Grid Security.
6. Explain the various concepts involved in a Grid programming model.

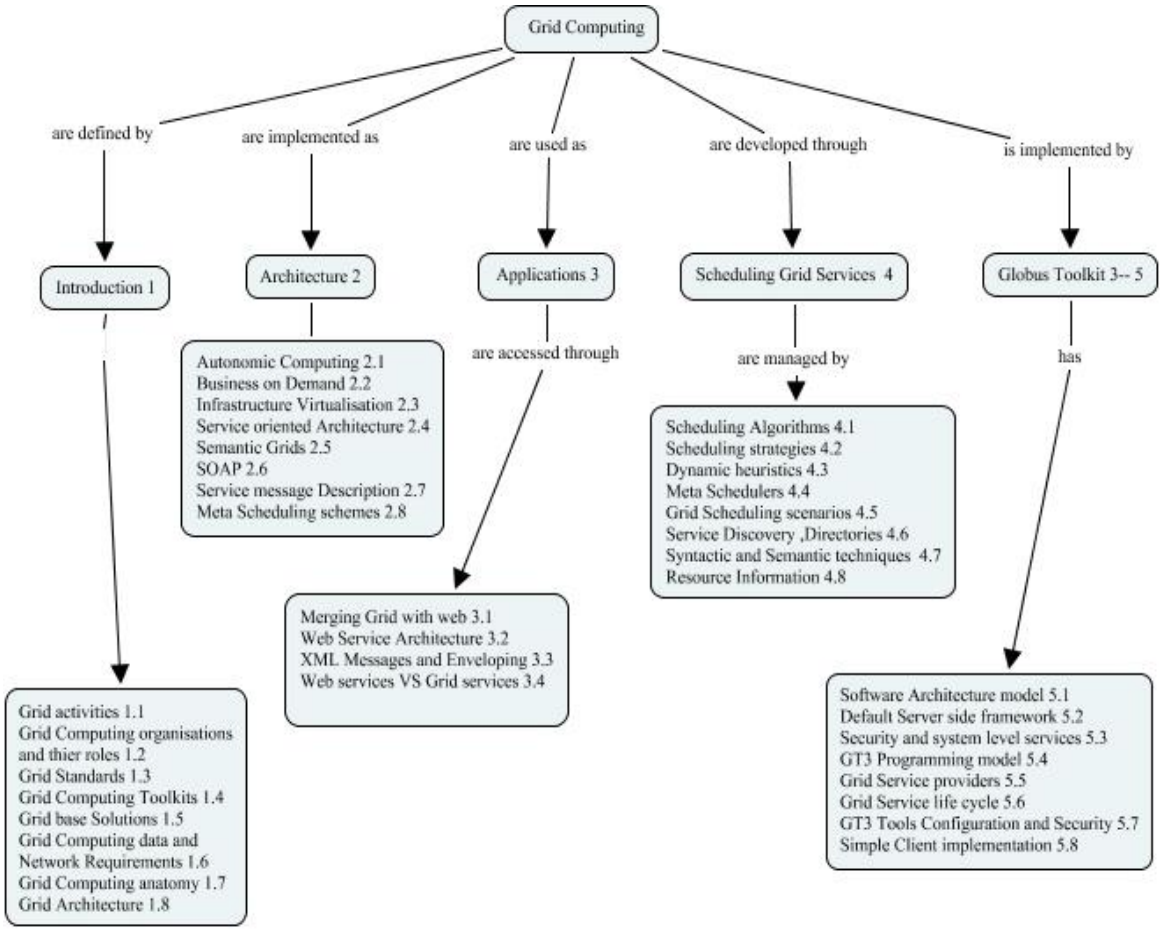
Apply

1. What's wrong with master/slave grid architectures?
2. Can Internet-based grid applications be made secure?
3. Why is fault tolerance vital to the success of grid computing?
4. Is it safe to rely on extensive use of remote PCs?
5. How is grid computing charged back to different cost centers?
6. Does web service architecture suits to work with grid architecture.

Create

1. Create a computational grid that is capable of solving complex applications like satellite image processing, Specify the hardware and software requirement of crating the execution environment. Justify your specifications.
2. Construct a control flow graph for processing both batch and parallel jobs in a grid computing environment. Suggest a scheduling algorithm that suits to both cases.
3. A Local Area Network contains five high end servers supported by gigabit Ethernet switch and CAT6 cabling and with 50 nodes similar setup exists in the other side .when both are connected by using a Bridge, how will you deploy an application to this environment that could run on both the LANS. What are the discovery services required to utilize all the resources available in the environment.
4. Design a web portal through which you can avail the resources available in a Grid environment. List out the parameters that have to be included to access the Grid services through the web service.
5. Create simple client server architecture with the help of GLOBUS Tool kit 3 and list out the steps involved in deploying an application to the server. Explain the various accessories in software to be installed along with the Globus tool kit to obtain maximum utilization of the grid.

Concept Map



Syllabus

INTRODUCTION: Grid activities -Grid computing organizations and their roles-Grid standards -Grid Computing Toolkits -Grid base solutions-Grid computing data, & network requirements. Grid computing anatomy –Grid Architecture; **ARCHITECTURE:** Autonomic computing -Business on demand -infrastructure virtualization – service oriented architecture- semantic grids-SOAP-Service message description-meta scheduling schemes; **APPLICATION:** Hadoop-Map Reducing concepts-Merging the Grid with web-Web Service Architecture- XML messages and Enveloping- Web services Vs Grid services; **SCHEDULING GRID SERVICES:** Scheduling algorithms-Scheduling strategies: – Dynamic heuristics - Meta-schedulers - Grid scheduling scenarios –, Service Discovery-Service directories - syntactic and semantic techniques- Resource information; **GLOBUS TOOLKIT 3:** software architecture model- default server side framework – security and system level services.GT3 programming model**-** Grid service providers- - Grid service life cycle-GT3tools configuration and security - simple client implementation

Text Books:

1. Joshy Joseph & Craig Fellenstein "Grid Computing", IBM Press, 2003.
2. Frédéric Magoulès, Thi-Mai-Huong Nguyen,Lei Yu "Grid Resource Management" , CRC press.

Reference Books:

1. William Daniel Minoli, "E A networking approach to Grid Computing ", Wiley interscience, 2004 .
2. Micheal L.Pinedo, "Scheduling, theory algorithms and systems", 3rd edition,springer 2006
3. Ian Foster & Carl Kesselman , "The Grid2 Blueprint for a New Computing Infrastructure"- Morgan Kaufmann, 2003
4. www.buyya.com

Course Contents and Lecture Schedule

S.No.	Topic	No of Lectures
1.	**INTRODUCTION**	
1.1	Grid activities.	2
1.2	Grid computing organizations and their roles.	1
1.3	Grid standards.	1
1.4	Grid Computing Toolkits.	2
1.5	Grid base solutions.	1
1.6	Grid computing data and network requirements.	1
1.7	Grid computing anatomy.	1
1.8	Grid Architecture.	2
2.	**ARCHITECTURE**	
2.1	Autonomic computing.	2
2.2	Business on demand.	1
2.3	Infrastructure virtualization.	1
2.4	Service oriented architecture.	2
2.5	Semantic grids.	1
2.6	SOAP	1
2.7	Service message description.	1
2.8	Meta scheduling schemes.	1
3.	**APPLICATIONS**	
3.1	Hadoop and Map Reducing Concepts	2
3.2	Merging Grid WITH Web.	1
3.3	Web Service Architecture.	1

3.4	XML messages and Enveloping.	2
3.5	Web services Vs Grid services.	2
4	**SCHEDULING GRID SERVICES:**	
4.1	Scheduling algorithms.	1
4.2	Scheduling strategies.	1
4.3	Dynamic heuristics.	1
4.4	Meta-schedulers.	2
4.5	Grid scheduling scenarios.	1
4.6	Service Discovery, directories.	2
4.7	Syntactic and semantic Techniques.	1
4.8	Resource information.	1
5	**GLOBUS GT3 TOOLKIT**	
5.1	Software architecture model.	1
5.2	Default server side framework.	1
5.3	Security and system level services.	2
5.4	GT3 Programming Model.	1
5.5	Grid service providers.	2
5.6	Grid service life cycle.	1
5.7	GT3tools configuration and security.	2
5.8	simple client implémentation.	1
	Total	**50**

Course Designers

1. N.Balaji nbcse@tce.edu
2. N.Shivakumar shiva@tce.edu

REVISED CURRICULUM AND DETAILED SYLLABI

FOR

B.Tech DEGREE (Information Technology) PROGRAM

GENERAL ELECTIVES

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2008-2009 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Sub Code	Lectures	Tutorial	Practical	Credits
TGA	3	0	-	3

TGA Database Management Systems 3:0

Preamble: This course aims at facilitating the student to understand the various functionalities of DBMS, to perform many operations related to creation, usage and maintenance of databases for real-world applications and to emphasize the need for design of database systems

Programming Outcomes Addressed

c. Graduates will demonstrate an ability to design a system, component or process as per needs and specifications.

e. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

k. Graduates will demonstrate skills to use modern engineering tools, software and equipment to analyze problems.

Competencies

At the end of the course the student will be able to

1. Understand the concepts of Data Base Management System.
2. Develop Entity Relationship (ER) diagrams for applications.
3. Create, alter and modify databases.
4. Retrieve data from databases including row selection, limiting the selection and single-row functions.
5. Develop stored procedures and functions to interact with the databases.
6. Participate in the certification courses.

Assessment Pattern

	Bloom's Category	**Test 1**	**Test 2**	**Test 3/ End-semester Examination**
1.	Remember	20	20	10
2.	Understand	30	30	40
3.	Apply	50	50	50
4.	Analyze	0	0	0
5.	Evaluate	0	0	0
6.	Create	0	0	0

Course Level Learning Objectives

Remember

1. What is a schema?
2. What are the different constraints exist in SQL?
3. What is an ER model?
4. Define strong entity set with an example.
5. What is meant by functional dependency?
6. What is an index in database?

Understand

1. Describe the three schema architecture of DBMS.
2. How do you classify DBMS?
3. Explain the join dependency with an example.
4. Outline the desirable properties of decompositions.
5. Explain the different components of DBMS.
6. Describe various Data Dictionary views.

Apply

1. Compare and contrast the DBMS with respect to File System.
2. How do integrity constraints simplify the retrieval of data from the database?
3. Distinguish between 2NF and 3NF with suitable examples.
4. Consider the following relations:

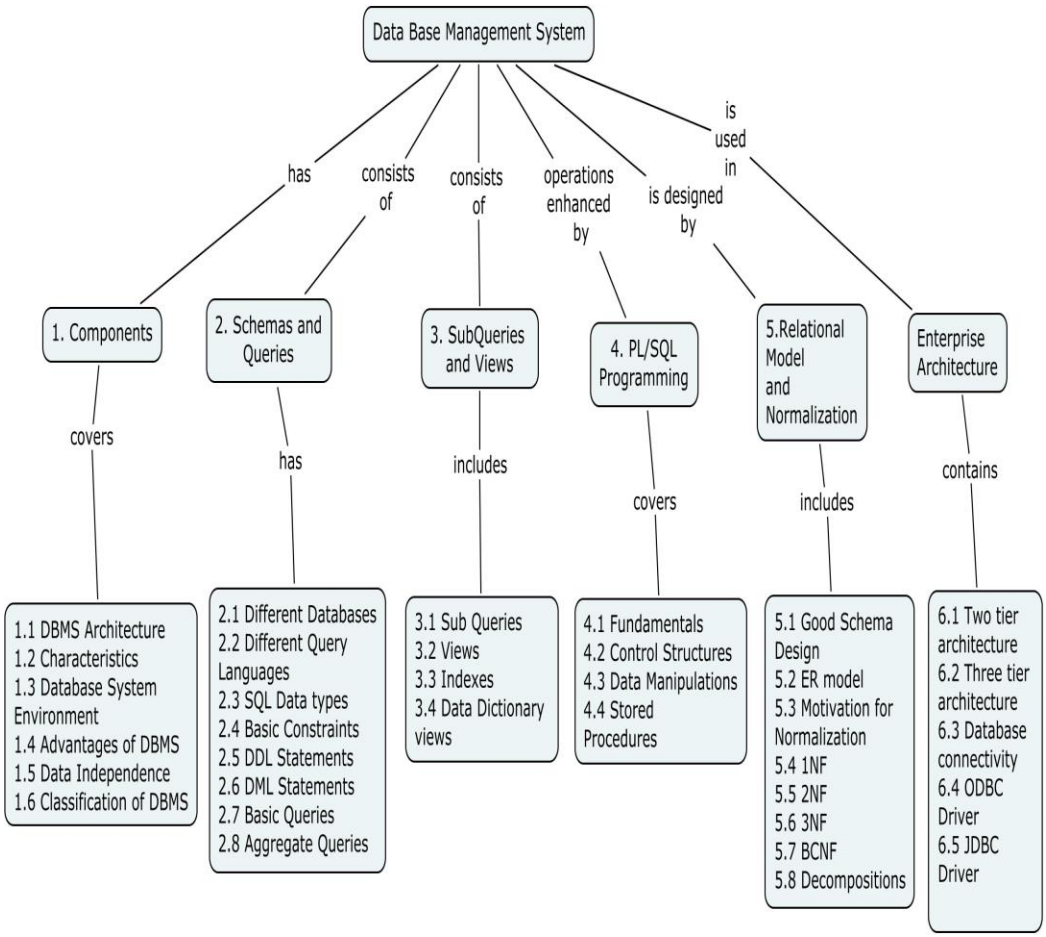
   Employee(empID, fname, lname, address, DOB, sex, position, dNo)
   Department(dptNo, dName, mgr, empID)
   Project(pNo, pName, dNo)
   WorksOn(empID, pNo, hours_worked)

   Write SQL statements for the following:

   i. List the name and address of all employees who work for IT dept.
   ii. List the total hours worked by each employee, arranged in order of department number and within department, alphabetically by employee lname.
   iii. List the total number of employees in each department for those departments with more than 10 employees.
   iv. List the project number, project name and the number of employees who work on that project.
   v. List the number of hours worked on each project group by manager.
5. Write PL/SQL procedure to list the total hours worked by each employee, arranged in order of department number and within department, alphabetically by employee lname

6. Write PL/SQL function to list the employees if their hours worked goes below 10 hours/week.

Concept Map



Syllabus

DBMS Components – Overview of DBMS Architecture, Characteristics of Database approach, Advantages of using DBMS approach, Data Independence, Database System Environment, Classification of DBMS **Schema Definitions, Basic Constraints and Queries** – Overview of databases – Oracle, Sybase and DB2, Overview of Database Query Languages – SQL, PL/SQL and MySQL, SQL data definition and data types, basic constraints in SQL, SQL DDL statements, SQL DML Statements, Basic queries in SQL, Aggregate Queries in SQL **Sub Queries and Views** - Sub Queries in SQL, Views in SQL, Indexes in SQL, Data Dictionary Views **Programming Language Extension to SQL** (PL/SQL) – Fundamentals, Control Structures, Data Manipulations, Stored Procedures and Functions **Relational Data Model and Normal Forms** –Importance of a good schema design, ER model, Motivation for Normal forms, Definitions of 1NF, 2NF, 3NF and BCNF, Decompositions **Enterprise Architecture** – Two tier architecture, Three tier architecture, Database Connectivity, ODBC Driver, JDBC Driver.

Text Book:

1. Ramez Elmasri and Shamkant B.Navathe, "Fundamentals of Database Systems", 4th edition, Pearson Education, 2004.
2. Nilesh Shah, "Database Systems using Oracle", 2nd edition, Prentice Hall of India Pvt Ltd, 2007.

Reference Books:

1. Abraham Silberschatz, Henry F.Korth and Sudarshan, "Database System Concepts", 5th edition, Mcgraw Hill, 2006.
2. Introduction to Data Base Management Systems, Atul Kahate, Pearson Education, 1st Edition, 2004.
3. Database Systems, Jeffrey Ullman, Jennifer Widom, Pearson Education, 3rd Edition, 2009.
4. Web Technologies, Atul Kahate, Tata Mcgraw Hill, 2nd Edition, 2008.

Course contents and Lectures Schedule:

S.No.	Topic	No. of Lectures
1	**DBMS Components**	
1.1	Overview of DBMS architecture	1
1.2	Characteristics of DBMS	1
1.3	Database System Environment	1
1.4	Advantages of using DBMS approach	1
1.5	Data Independence	1
1.6	Classification of DBMS	1
2	**Schema Definitions, Basic Constraints and Queries**	
2.1	Overview of databases – Oracle, Sybase, DB2	1
2.2	Database Query Languages – SQL, MySQL, PL/SQL,	1
2.3	SQL data definition and data types	1
2.4	Basic constraints in SQL	1
2.5	SQL DDL Statements	1
2.6	SQL DML Statements	2
2.7	Basic Queries in SQL	1
2.8	Aggregate Queries in SQL	1
3	**Sub Queries and Database Views**	
3.1	Sub Queries in SQL	2
3.2	Views in SQL	1
3.3	Indexes in SQL	1
3.4	Data Dictionary Views	1
4	**Programming Language Extension to SQL** (PL/SQL)	

S.No.	Topic	No. of Lectures
4.1	Fundamentals	2
4.2	Control Structures	2
4.3	Data Manipulations	2
4.4	Stored Procedures and Functions	2
5	**Relational Data Model Normal Forms**	
5.1	Importance of a good schema design	1
5.2	ER model	2
5.3	Motivation for Normal forms	1
5.4	First Normal Form	1
5.5	Second Normal Form	1
5.6	Third Normal Form	1
5.7	BCNF Normal Form	1
5.8	Decompositions	1
6	**Enterprise Architecture**	
6.1	Two tier Architecture	1
6.2	Three tier Architecture	1
6.3	Database Connectivity	1
6.4	ODBC Driver	1
6.5	JDBC Driver	1
	Total	**42**

Course Designers:

1. G.S.R. Emil Selvan emil@tce.edu
2. A.M.Abirami abiramiam@tce.edu

REVISED CURRICULUM AND DETAILED SYLLABI

FOR

B.Tech DEGREE (Information Technology) PROGRAM

DEPARTMENTAL ELECTIVE

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2009-2010 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
TCS	3	0	--	3

TCS Principles of Operating Systems 3:0

Preamble: Operating System provides a very wide range of generic data devices to I/O, store and process information in a computer system. This is achieved by managing devices such as keyboard, display unit, process, memory and other I/O devices. Therefore it is very essential course for an IT professional. It covers the fundamental principles, management of resources, protection and security in detail.

Programme Outcomes addressed

a) Graduates will demonstrate an ability to identify, formulate and solve engineering problems.
b) Graduates will demonstrate an ability to design a system, component or process as per needs and specifications.
c) Graduate will demonstrate skills to use modern engineering tools, software and equipment to analyze problems.

Competencies: The students are able to

1. Explain the operating system structure.
2. Understand how the resources, CPU memory I/O devices and Files are managed by OS.
3. Apply the scheduling algorithms for finding average waiting time and turnaround time.
4. Allocate memory for blocks by using replacement algorithms to replace a page when a page fault occurs.
5. Understand how access control is done.

Assessment Pattern

	Bloom's Category	**Test 1**	**Test 2**	**End-semester examination**
1	Remember	30	30	20
2	Understand	40	40	40
3	Apply	30	30	40
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	0	0	0

Course Level Learning Objectives

Remember

1. What is a thread?
2. What are the characteristics of mobile operating systems?
3. Define locality of reference
4. What is the use of translation look aside buffer?
5. What is internal fragmentation?
6. Define seek time.
7. Why protection is needed in a file sharing systems?

Understand:

1. Explain how memory, CPU and IO protection is achieved?
2. State the critical section problem. Discuss the solutions.
3. Describe how files can be allocated space on disk. Give suitable examples.
4. Discuss demand paging in detail.
5. Explain the methods of implementing the file system.
6. Discuss briefly the various issues involved in implementing Inter Process Communication (IPC) in message passing system.

Apply:

1. Assume the following workload in a system. Assume that all jobs arrive at time 0 in the order given below.

Job	Burst Time(ms)	Priority
J1	9	3
J2	3	1
J3	2	6
J4	8	5
J5	4	2
J6	1	4

   Draw a Gantt chart using FCFS, RR and SJF. Calculate the average waiting time and turnaround time for each of the scheduling algorithms.

2. Assume that 412K, 300K, 236K and 112K processes are to be placed in memory partitions of 100K, 200K, 300K and 500K.Out of First-fit, Best-Fit and Worst-fit algorithms, which uses the memory more efficiently?

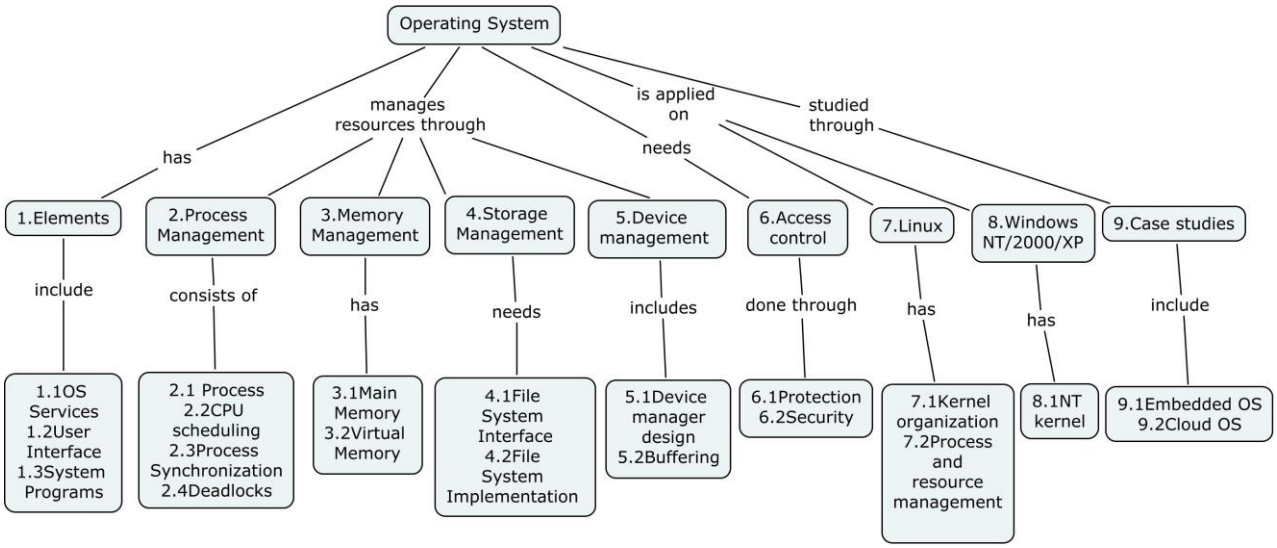
3. Construct a Resource allocation Graph for the following scenario. At time 't' process P1 requests for a resource X, Process P2 requests for a resource Y. Both the resources are available and they are allocated to the requesting process. At time t, where t1> t, both the processes are still holding the resources, however process P1 requests for Y which is held by P2, Process P2 requests for X held by requests by P1. Will there be a deadlock?
4. Consider the following page reference string:

   3, 4, 5, 1, 7, 8, 4 2, 6, 6, 3, 1, 2, 5, 5, 7, 1, 1.

   Find the number of page faults that would occur for the following page replacement algorithms (i) FIFO (ii) LRU (iii) optimal

5. Consider a disk queue with requests for I/O to blocks on cylinders in the following order: 100, 32, 132, 46, 89, 90, 2, 58. The disk head is initially at 67.Compute the total head movement for the following algorithms: (i) FCFS (ii) SSTF (iii) SCAN.
6. Assume that two file systems are available in a computer system. One file system allocates 4KB of disk space as a single 4-KB block. Another file system allocates 4-KB block as eight 512-byte blocks. How is the free space management performed?

Concept Map



Syllabus:

Elements – OS services, User Interface, System Programs. **Process management** – Process - Process state, Process control block, Threads, CPU scheduling – CPU I/O burst cycle, CPU synchronization, Preemptive scheduling, Process synchronization – Critical Section problem, Semaphores, Deadlocks – Prevention, Avoidance, Detection M**emory management** – Main memory – Address binding, Logical vs. physical address, dynamic loading, Dynamic linking and shared libraries, swapping, Contiguous memory allocation, Paging Virtual memory – Demand paging, Page replacement **Storage management** – File system Interface – File concept, Access methods, Directory structure File system Implementation - File system Structure, Implementation, Directory Implementation. **Device management –** Device manager design**,** buffering **Access Control** – Protection – Goals, Access matrix, Access control, Revocation of access control, Security – Security problem, Program threats **Linux** – Kernel Organization, Process and resource management **Windows NT/2000/XP** – NT kernel. Case studies – Embedded OS, Cloud OS.

Text Books:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne "Operating System Concepts", 7th Edition, John Wiley and Sons, 2005.
2. Gary Nutt, "Operating Systems", 3rd edition, Pearson education, 2004, Chapters 5, 20, 21.

Reference Books:

1. William Stallings Operating Systems: Internals and Design Principles, 5th Edition, Prentice Hall, 2005.
2. Andrew S.Tanenbaum, "Modern Operating Systems", 3rd Edition, Prentice Hall, 2007.

Course contents and Lecture Schedule:

No.	Topic	No. of Lectures
1	**Elements**	
1.1	OS services	1
1.2	User Interface	1
1.3	System Programs	1
2	**Process management**	

2.1	Process	
2.1.1	Process state	1
2.1.2	Process control block	1
2.1.3	Threads	1
2.2	CPU scheduling	
2.2.1	CPU I/O burst cycle	1
2.2.3	CPU synchronization	2
2.2.4	Preemptive scheduling	2
2.3	Process synchronization	
2.3.1	Critical Section problem	2
2.3.2	Semaphores	2
2.4	Deadlocks	
2.4.1	Prevention	1
2.4.2	Avoidance	1
2.4.3	Detection	1
3	M**emory management**	
3.1	Main memory	
3.1.1	Address binding , Logical vs. physical address	1
3.1.2	Dynamic linking and shared libraries	1
3.1.3	Swapping	1
3.1.4	Contiguous memory allocation	1
3.1.5	Paging	1
3.2	Virtual memory	
3.2.1	Demand paging	1
3.2.2	Page replacement	1
4	**Storage management**	
4.1	File system Interface	
4.1.1	File concept	1

4.1.2	Access methods	1
4.1.3	Directory structure	1
4.2	File system Implementation	
4.2.1	File system Structure	1
4.2.2	Implementation	1
4.2.3	Directory Implementation	1
5	**Device management**	
5.1	Device manager design	1
5.2	Buffering	1
6	**Access Control**	
6.1	Protection	
6.1.1	Goals	1
6.1.2	Access matrix	1
6.1.3	Access control	1
6.1.4	Revocation of access control	1
6.2	Security	
6.2.1	Security problem	1
6.2.2	Program threats	1
7	**Linux**	
7.1	Kernel Organization	1
7.2	Process and resource management	1
8	**Windows NT/2000/XP**	
8.1	NT kernel	1
9	**Case Studies**	
9.1	Embedded OS	1
9.2	Cloud OS	2

Course Designers:

S.Srinivasan ssnit@tce.edu

R.Muthu Selvi rmsit@tce.edu

REVISED CURRICULUM AND DETAILED SYLLABI

FOR

B.Tech DEGREE (Information Technology) PROGRAM

GENERAL ELECTIVES

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2008-2009 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Sub Code	Lectures	Tutorial	Practical	Credits
TGA	3	0	-	3

TGA Database Management Systems 3:0

Preamble: This course aims at facilitating the student to understand the various functionalities of DBMS, to perform many operations related to creation, usage and maintenance of databases for real-world applications and to emphasize the need for design of database systems

Programming Outcomes Addressed

c. Graduates will demonstrate an ability to design a system, component or process as per needs and specifications.

e. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

k. Graduates will demonstrate skills to use modern engineering tools, software and equipment to analyze problems.

Competencies

At the end of the course the student will be able to

1. Understand the concepts of Data Base Management System.
2. Develop Entity Relationship (ER) diagrams for applications.
3. Create, alter and modify databases.
4. Retrieve data from databases including row selection, limiting the selection and single-row functions.
5. Develop stored procedures and functions to interact with the databases.
6. Participate in the certification courses.

Assessment Pattern

	Bloom's Category	**Test 1**	**Test 2**	**Test 3/ End-semester Examination**
1.	Remember	20	20	10
2.	Understand	30	30	40
3.	Apply	50	50	50
4.	Analyze	0	0	0
5.	Evaluate	0	0	0
6.	Create	0	0	0

Course Level Learning Objectives

Remember

1. What is a schema?
2. What are the different constraints exist in SQL?
3. What is an ER model?
4. Define strong entity set with an example.
5. What is meant by functional dependency?
6. What is an index in database?

Understand

1. Describe the three schema architecture of DBMS.
2. How do you classify DBMS?
3. Explain the join dependency with an example.
4. Outline the desirable properties of decompositions.
5. Explain the different components of DBMS.
6. Describe various Data Dictionary views.

Apply

1. Compare and contrast the DBMS with respect to File System.
2. How do integrity constraints simplify the retrieval of data from the database?
3. Distinguish between 2NF and 3NF with suitable examples.
4. Consider the following relations:

   Employee(empID, fname, lname, address, DOB, sex, position, dNo)

   Department(dptNo, dName, mgr, empID)

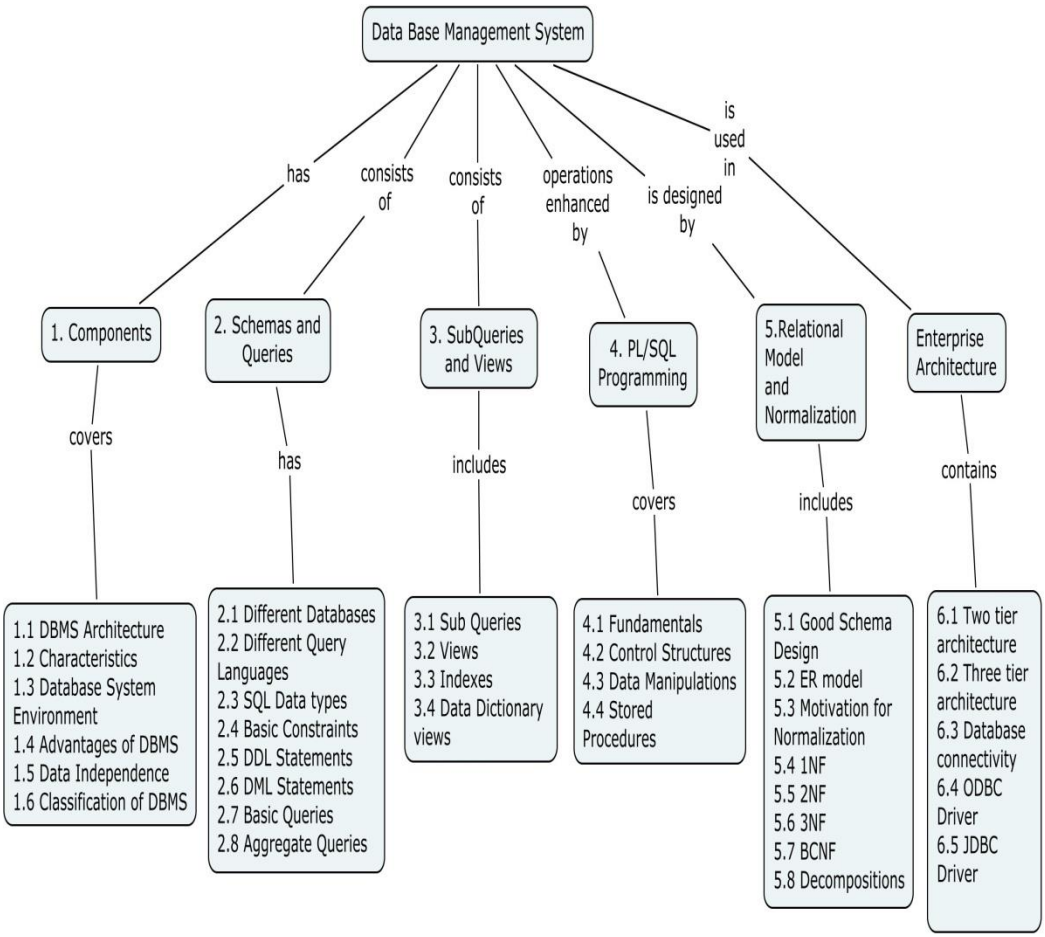
   Project(pNo, pName, dNo)

   WorksOn(empID, pNo, hours_worked)

   Write SQL statements for the following:

   i. List the name and address of all employees who work for IT dept.
   ii. List the total hours worked by each employee, arranged in order of department number and within department, alphabetically by employee lname.
   iii. List the total number of employees in each department for those departments with more than 10 employees.
   iv. List the project number, project name and the number of employees who work on that project.
   v. List the number of hours worked on each project group by manager.
5. Write PL/SQL procedure to list the total hours worked by each employee, arranged in order of department number and within department, alphabetically by employee lname
6. Write PL/SQL function to list the employees if their hours worked goes below 10 hours/week.

Concept Map



Syllabus

DBMS Components – Overview of DBMS Architecture, Characteristics of Database approach, Advantages of using DBMS approach, Data Independence, Database System Environment, Classification of DBMS **Schema Definitions, Basic Constraints and Queries** – Overview of databases – Oracle, Sybase and DB2, Overview of Database Query Languages – SQL, PL/SQL and MySQL, SQL data definition and data types, basic constraints in SQL, SQL DDL statements, SQL DML Statements, Basic queries in SQL, Aggregate Queries in SQL **Sub Queries and Views** - Sub Queries in SQL, Views in SQL, Indexes in SQL, Data Dictionary Views **Programming Language Extension to SQL** (PL/SQL) – Fundamentals, Control Structures, Data Manipulations, Stored Procedures and Functions **Relational Data Model and Normal Forms** –Importance of a good schema design, ER model, Motivation for Normal forms, Definitions of 1NF, 2NF, 3NF and BCNF, Decompositions **Enterprise Architecture** – Two tier architecture, Three tier architecture, Database Connectivity, ODBC Driver, JDBC Driver.

Text Book:

1. Ramez Elmasri and Shamkant B.Navathe, "Fundamentals of Database Systems", 4th edition, Pearson Education, 2004.

2. Nilesh Shah, "Database Systems using Oracle", 2nd edition, Prentice Hall of India Pvt Ltd, 2007.

Reference Books:

1. Abraham Silberschatz, Henry F.Korth and Sudarshan, "Database System Concepts", 5th edition, Mcgraw Hill, 2006.
2. Introduction to Data Base Management Systems, Atul Kahate, Pearson Education, 1st Edition, 2004.
3. Database Systems, Jeffrey Ullman, Jennifer Widom, Pearson Education, 3rd Edition, 2009.
4. Web Technologies, Atul Kahate, Tata Mcgraw Hill, 2nd Edition, 2008.

Course contents and Lectures Schedule:

S.No.	Topic	No. of Lectures
1	**DBMS Components**	
1.1	Overview of DBMS architecture	1
1.2	Characteristics of DBMS	1
1.3	Database System Environment	1
1.4	Advantages of using DBMS approach	1
1.5	Data Independence	1
1.6	Classification of DBMS	1
2	**Schema Definitions, Basic Constraints and Queries**	
2.1	Overview of databases – Oracle, Sybase, DB2	1
2.2	Database Query Languages – SQL, MySQL, PL/SQL,	1
2.3	SQL data definition and data types	1
2.4	Basic constraints in SQL	1
2.5	SQL DDL Statements	1
2.6	SQL DML Statements	2
2.7	Basic Queries in SQL	1
2.8	Aggregate Queries in SQL	1
3	**Sub Queries and Database Views**	
3.1	Sub Queries in SQL	2
3.2	Views in SQL	1
3.3	Indexes in SQL	1
3.4	Data Dictionary Views	1
4	**Programming Language Extension to SQL** (PL/SQL)	
4.1	Fundamentals	2
4.2	Control Structures	2

S.No.	Topic	No. of Lectures
4.3	Data Manipulations	2
4.4	Stored Procedures and Functions	2
5	**Relational Data Model Normal Forms**	
5.1	Importance of a good schema design	1
5.2	ER model	2
5.3	Motivation for Normal forms	1
5.4	First Normal Form	1
5.5	Second Normal Form	1
5.6	Third Normal Form	1
5.7	BCNF Normal Form	1
5.8	Decompositions	1
6	**Enterprise Architecture**	
6.1	Two tier Architecture	1
6.2	Three tier Architecture	1
6.3	Database Connectivity	1
6.4	ODBC Driver	1
6.5	JDBC Driver	1
	Total	**42**

Course Designers:

1. G.S.R. Emil Selvan emil@tce.edu
2. A.M.Abirami abiramiam@tce.edu

REVISED CURRICULUM AND DETAILED SYLLABI

FOR

B.Tech DEGREE (Information Technology) PROGRAM

GENERAL ELECTIVES

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2009-2010 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Subject Code	Lecture	Tutorial	Practical	Credit
TGB	3	0	--	3

TGB Object Oriented Programming with Java 3:0

Preamble

This course provides a comprehensive introduction to the object-oriented approach through the widely-used Java programming language. Java is a multipurpose, easy to learn, powerful platform where virtually a complete application in any domain can be implemented. Java is the most widely used language for developing open source software systems. With Java, there comes huge library support which reduces the development time and enhances software reuse for faster development. This course concentrates on object oriented aspects of Java and from this course the student will gain a solid basis for further study of the object-oriented software development.

Programming Outcomes Addressed

a. Graduates will demonstrate an ability to apply knowledge of engineering, information technology, mathematics and science.

b. Graduates will demonstrate an ability to design and conduct experiments, as well as to analyze and interpret data.

c. Graduates will demonstrate an ability to design a system or component, or process to meet stated specifications.

d. Graduates will demonstrate an ability to identify, formulate and solve engineering problems.

e. Graduates will demonstrate an ability to use techniques, skills, and modern engineering tools to implement and organize engineering works under given constraints.

Competencies

At the end of the course the student will be able to

1. Understand the concepts of object oriented programming like abstraction, encapsulation and polymorphism.
2. Implement exception handling mechanisms, packages, and interfaces in Java.
3. Create Java applications that are portable and secure.
4. Design applications for the Internet.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test3/ End semester Examination
1	Remember	20	10	10
2	Understand	30	20	10
3	Apply	50	50	60
4	Analyze	0	0	0
5	Evaluate	0	0	0
6	Create	0	20	20

Learning Objectives

Remember

1. Define polymorphism.
2. What is a constructor?
3. How data encapsulation is achieved in java?
4. Define Dynamic Method Dispatch.
5. What are abstract classes?
6. Define interface.
7. List and explain the different access modifiers.
8. What is the role of the finally block in exception handling?
9. What are I/O streams?
10. What are AWT controls?

Understand

1. What is the difference between a constructor and a member method?
2. List the similarities and differences between an abstract class and an interface.
3. How super keyword is used with respect to inheritance?
4. What are the three forms in which the final keyword can be used?
5. Differentiate method overloading and method overriding.
6. How will you execute a program in a package that used a class in another package?
7. How exception handling is done in Java.
8. What is the difference between Integer and int?
9. What is the difference between exception and error?
10. Demonstrate how throw and throws differ in Exception handling.

Apply

1. Write a recursive function in Java to generate a Fibonacci series.
2. Write a Java program that prints the numbers in the range [1 – 100] such that the difference between successive numbers increase by 1.That is it should print 1, 2, 4, 7, 11, 16, etc.

3. Create a applet that accepts two input string using <param> tag and concatenate the strings and display it in status window.
4. Write a program to i) convert lower case string to upper case ii) to compare two strings.
5. Let 'Series' be an interface with int sum(int n) as its method. Let class1 and class2 be the two classes implementing the 'Series' interface to perform sum of numbers upto n and squared sum of numbers upto n respectively. From the main class use both the implementations through an interface reference.
6. Write a JAVA program to implement the Employee Details in a company.
   Let employee be the parent class with name, designation, no of days worked.
   Write an abstract method to compute salary depending on the no of days worked.
   Derive two subclasses Clerk and Manager.
   Override the salary computation method in both the subclasses.
   Use constructor overloading in the parent class.
   (Note: For salary computation include HRA, DA, Medical Allowance for clerk and Manager separately)
7. Write a JAVA program to implement the activities in the BANK.
   Let bank be the parent class with account no, name.
   Write an abstract method for interest calculation.
   Derive three subclasses Fixed Deposit, Variable Recurring Deposit and Savings account transactions.
   Override the interest calculation method in all the subclasses.
   Let the Saving account class has withdraw and deposit methods in addition to the interest calculation method. Use method overloading also. (Note: Assume different interest rates for the different types).

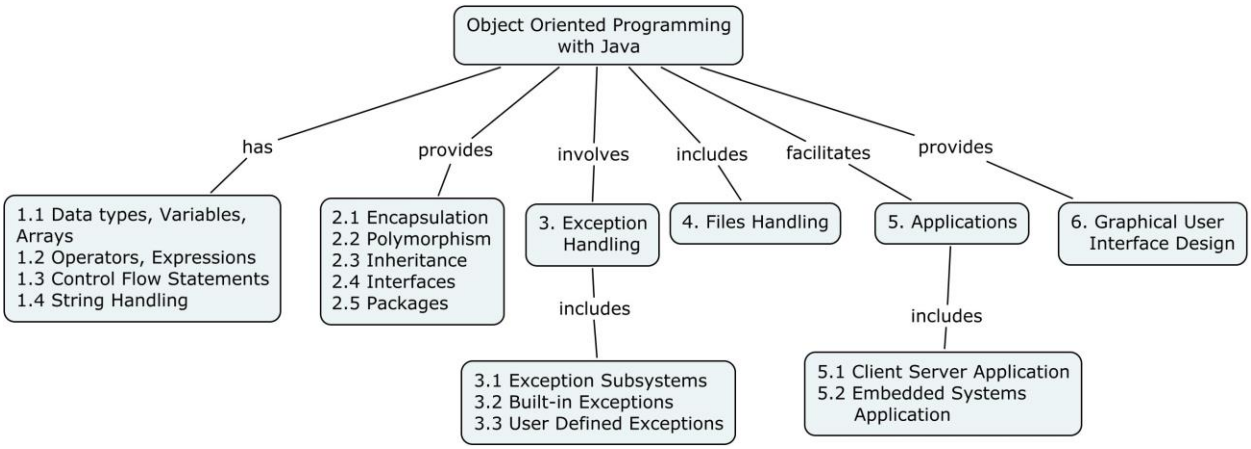
Create

1. Create the Client/Server application to calculate the factorial of a number using Remote Method Invocation in JAVA.
2. Design an interface Queue, with methods to add and remove elements (integers). Furthermore, a method to check whether the queue is empty or not should exist. Implement the queue with an array. If the array becomes too small to hold all added elements, create a new larger (double the size of instance) array and copy all elements of the small array to the new one.
3. For a Bank application create a user defined exception subclass for the following conditions (i) Account holder's age should not be less than 18, if it is less then a joint

account holder has to be specified (ii) Minimum balance after withdrawal should not be less than 500. From the main class, create the account and do withdrawal operation.

4. Design a web page using AWT components and handle the events generated by each of them.
5. Create two packages pack1 and pack2 for an airline reservation system. Let the Flight's private details like flight name, no., fare be in the pack1 and reservation details like no. of seats available in pack2. From pack2 get the no. of tickets required, check its availability and calculate the fare.
6. Create a calculator using the AWT controls to get two numbers, perform one of the operations like addition, subtraction, multiply and divide and display the result.

Concept Map



Course Contents and Lectures schedule

0	**Genesis and Overview of JAVA**	2
1.	**Programming Constructs**	
1.1	Data types, Variable, Arrays	2
1.2	Operators and Expressions	2
1.3	Control Flow Statements	2
1.4	String Handling	2
2	**Object Oriented Constructs**	
2.1	Encapsulation - Classes and Objects	3
2.2	Polymorphism - Constructor Overloading, Method Overloading	3
2.3	Inheritance	3

2.4	Interfaces	2
2.5	Packages	2
3	**Exception Handling**	
3.1	Exception subsystems	3
3.2	Built-in Exceptions	2
3.3	User Defined Exceptions	2
4	**File Handling**	**2**
5	**Applications**	
5.1	Client Server Application	2
5.2	Embedded Application	2
6	**Graphical User Interface Design - Applets**	**4**
	Total Hours	**40**

Syllabus

Genesis and Overview of JAVA. Programming Constructs: Data types, Variables, Arrays, Operators and Expressions, Control Flow Statements, String Handling. **Object Oriented Constructs:** Encapsulation – Classes and Objects, Polymorphism – Constructor Overloading and Method Overloading, Inheritance, Interfaces and Packages. **Exception Handling:** Exception Subsystem, Built-in Exceptions, User Defined Exceptions. **File Handling, Applications:** Client Server Application, Embedded Application**. Graphical User Interface Design:** Applets.

Text Book

1. Herbert Schildt: The complete reference – Java2, Fifth Edition, Mc Graw Hill Companies, 2002.
2. Deitel and Deitel, Java How to Program, Seventh Edition, Pearson Education, 2007

References

1. Thinking in Java, Bruce Eckel's Free e-book http://www.codeguru.com/java/tij/

Web Resources on Java

1. Tutorial on Java http://www.java2s.com/Tutorial/Java/CatalogJava.htm
2. Oracle tutorial for Java http://download.oracle.com/javase/tutorial/

Course Designer

1. J. Jennifer Ranjani jenni@tce.edu

REVISED CURRICULUM AND DETAILED SYLLABI

FOR

GENERAL ELECTIVES

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2012-2013

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2000 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
TGC	3	-	-	3

TGC SOFTWARE ENGINEERING 3:0

Preamble This course presents methods, tools and procedures that enable to control the process of software development and provide the student with a foundation for building quality software in a productive manner. The course highlights the application of systematic, disciplined, quantifiable approach to the development, operation and maintenance of software.

Program Outcomes addressed

Graduates will demonstrate

c. an ability to design a system, component or process as per needs and specifications.

d. an ability to identify, formulate and solve engineering problems.

e. an ability to use techniques, skills, and modern engineering tools to implement and organize engineering works under given constraints.

Competencies

Students will be able to

1. Comprehend different process models and select suitable model for their project requirements.
2. Appraise the software engineering practices and methodologies.
3. Apply testing tactics and tools on software applications.
4. Use appropriate managerial solutions for software projects.

Assessment Pattern

	Bloom's Category	**Test 1**	**Test 2**	**Test 3**	**End-semester examination**
1	Remember	30	20	10	10
2	Understand	30	40	50	50
3	Apply	40	30	30	30
4	Analyze	0	10	10	10
5	Evaluate	0	0	0	0
6	Create	0	0	0	0

Course Level Learning Objectives:

Remember

1. List the components of S/W engineering.
2. Recall the important elements of SCM.

3. Mention the different approaches to the sizing problem.
4. Recognize how an acceptance test is executed and evaluated.
5. Exhaustive testing is not possible. Justify.
6. List out the software risks.

Understand

1. Write down the differences between Incremental process model and evolutionary process model.
2. Explain the application where agile process model is implemented.
3. Discuss different design techniques used in software design.
4. Discuss how testing is performed in various levels of the software. Give examples.
5. Explain how Risk estimation is done during project development.
6. Discuss about Software reviews.

Apply

1. Using the Cocomo models, estimate the effort required to build a S/W for a Simple ATM that produces 24 screens, 15 reports and will require approximately 110 S/W components. Assume average complexity and average developer / environment maturity. Use the application composition model with object points.
2. Prepare an SRS document for Online shopping S/W.
3. Draw Use-Case diagram for some of the use-cases of the library management system.
4. Draw the CFG and Calculate the cyclomatic Complexity for the following program.

```
int compute_gcd(x, y)
int x, y;
{
while (x! = y){
if (x>y) then
x= x – y;
else y= y – x;
 }
 return x;
}
```

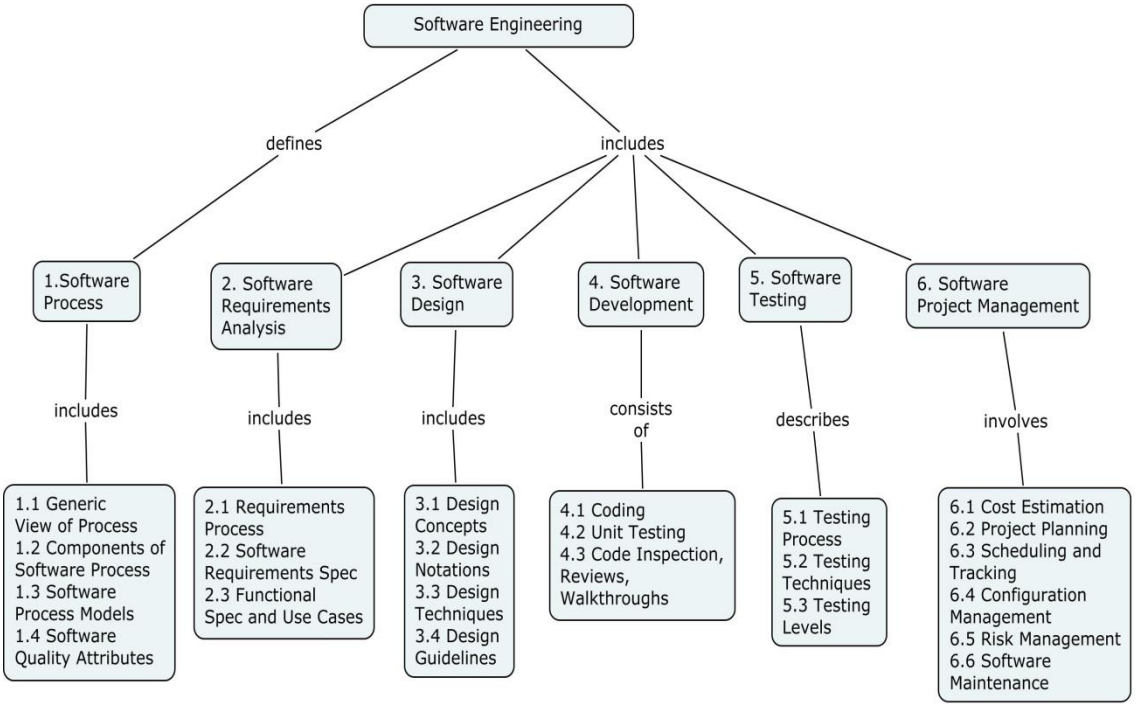
5. Select which software life cycle model is applicable for weather monitoring system.
6. Implement the Empirical estimation model for any real time application.

Analyze

1. Differentiate the various processing models highlighting their advantages and disadvantages.

2. Analyze the tasks that the software Engineer needs to perform to ensure that he has specified a system that properly meets the customer needs and satisfy the customer's expectations.
3. Validation test is performed by software developer or the software user. Justify your answer.
4. Analyze the different types of risks that can be possible in a development project.
5. Compare and contrast testing techniques and testing levels.
6. Justify the need for User acceptance testing.

Concept Map



Syllabus

Software Process – Generic View of Process, Components of Software Process, Software Process Models, Software Quality Attributes

Software Requirements Analysis – Requirements Process, Software Requirement Specifications, Functional Specifications and Use Cases

Software Design – Design Concepts, Design Notations, Design Techniques, Design Guidelines

Software Development – Coding, Unit Testing, Code Inspection, Reviews and Walkthroughs

Software Testing – Testing Process, Testing Techniques, Black Box Testing, White Box Testing, Testing Levels, Integration Testing, System Testing, User Acceptance Testing

Software Project Management – Cost Estimation, Project Planning, Scheduling and Tracking, Configuration Management, Risk Management, Software Maintenance

Textbook:

1. Pankaj Jalote.An Integrated Approach to Software Engineering, 2nd edition, Narosa Publishing House, 2011.
2. Richard Fairley, Software Engineering Concepts, TATA McGraw Hill, 2004.

Reference Books:

1. Roger S.Pressman, Software Engineering A Practitioner's Approach, sixth Edition, McGraw Hill International Edition.,2007
2. Ian Somerville-Software Engineering, Pearson Education India, New Delhi, 2006.

Some useful links:

1. http://elearning.tvm.tcs.co.in
2. http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/

Course Contents and Lectures schedule

Sl. No	Topic	No of Lectures
1	**Software Process**	
1.1	Generic View of Process	1
1.2	Components of Software Process	1
1.3	Software Process Models	4
1.4	Software Quality Attributes	1
2	**Software Requirements Analysis**	
2.1	Requirements Process	1
2.2	Software Requirement Specifications	2
2.3	Functional Specifications and Use Cases	2
3	**Software Design**	

3.1	Design Concepts	1
3.2	Design Notations	1
3.3	Design Techniques	2
3.4	Design Guidelines	1
4	**Software Development**	
4.1	Coding	1
4.2	Unit Testing	2
4.3	Code Inspection, Reviews and Walkthroughs	2
5	**Software Testing**	
5.1	Testing Process	1
5.2	Testing techniques	
5.2.1	Black Box testing	4
5.2.2	White Box testing	4
5.3	Testing Levels	
5.3.1	Integration testing	1
5.3.2	System testing	1
5.3.3	User Acceptance testing	2
6	**Software Project Management**	
6.1	Cost Estimation	2
6.2	Project Planning	2
6.3	Scheduling and Tracking	1
6.4	Configuration Management	2
6.5	Risk Management	2
6.6	Software Maintenance	1
	Total Lectures	**45**

Course Designer:

A.M.Abirami abiramiam@tce.edu

Sub Code	Lectures	Tutorial	Practical	Credit
TGD	3	0	--	3

TGD Object Oriented Programming with C++ 3:0

Preamble:

This course introduces basic concepts of object oriented programming and makes the students to apply these programming concepts towards problem solving. It helps them to effectively build and use ADTs when compared to the traditional structured programming. The course helps in learning object oriented programming using the "C++" language by having hands on Training. Topics include class definitions, polymorphism by overloading functions and operators, single and multiple inheritance, virtual functions, abstract classes, templates, and exception handling.

Prerequisites:

An experience in any programming language is desirable.

Program Outcomes addressed

Graduates will demonstrate

b. An ability to design and conduct experiments, as well as to analyze and interpret data.

c. An ability to design a system or component, or process to meet stated specifications

d. An ability to identify, formulate and solve engineering problems

Competencies:

The students will be able to

1. Differentiate among traditional programming and object oriented programming.
2. Profile their programs to identify which parts of the program are consuming the most resources.
3. Interpret object oriented constructs, Data hiding, object creation and destruction.
4. Describe the consequences of incorporating Object Oriented Programming concepts such as Inheritance, Polymorphism, Data abstraction, Templates and Exceptions into design.
5. Design real time examples corresponding to their field of study.

Assessment Pattern

	Bloom's Category	Test 1	Test 2	Test 3	End Semester Examination
1	Remember	40	40	30	30
2	Understand	40	30	40	40
3	Apply	20	20	20	20
4	Analyze	0	10	10	10
5	Evaluate	0	0	0	0
6	Create	0	0	0	0

Course Level Learning Objectives

Remember

1. Define bool data type in C++?
2. Differentiate pointers and references.
3. Give the difference between a structure and a class.
4. List the memory allocation operators.
5. Differentiate between constructor and destructor.
6. Explain conversion constructor?

Understand

1. Explain the difference between Procedural languages and Object Oriented languages.
2. Mention the need for OOP? Explain its benefits.
3. Explain the different types of data types that are used in C++?
4. Explain control structures available in C++ with syntax and examples.
5. Can you think of a situation where your program would crash without reaching the breakpoint which you set at the beginning of main ()?
6. Explain the following functions (with example) for manipulating file pointers: seekg(), seekp(), tellg(), tellp().

Apply

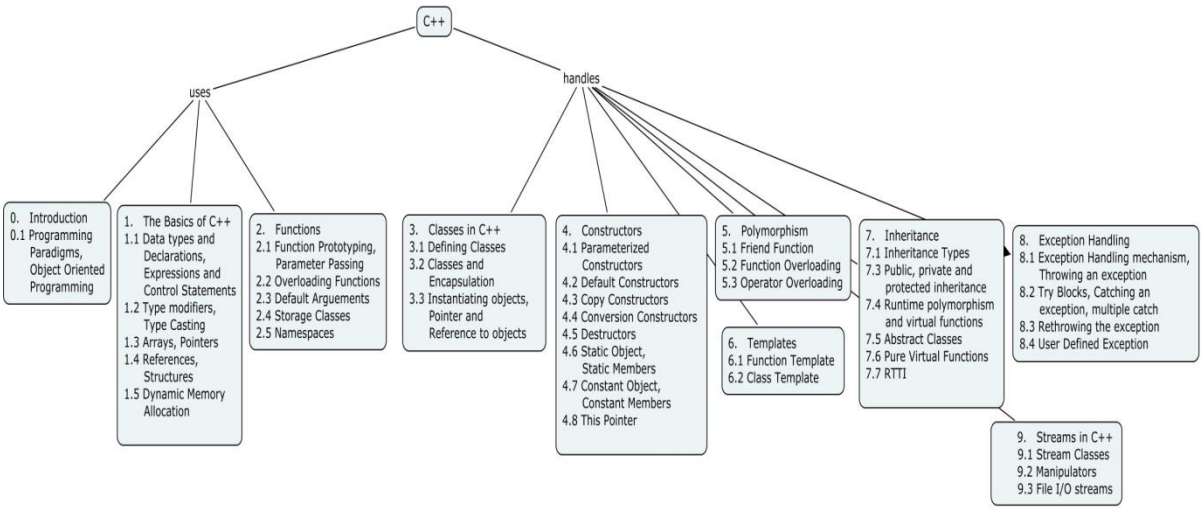
1. Write a program that performs string reversal. Assume that s1 ends up with the reverse of the string s2, and that s1 points at enough store that is adequate for reversal.
2. Write a C++ program using function template for sorting an array(assume for float array and int array)
3. Write a C++ file program to store the details of 10 students .Get inputs from user (name,m1,m2,m3).Find total & average and store it in a file called studin.dat

4. Write a program having a base class Student with data member rollno and member function getnum() to input rollno and putnum() to display rollno.
   A class Test is derived from class Student with data member marks and member functions getmarks() to input marks and putmarks() to display marks. Class Sports is also derived from class Student with data member score and member functions getscore() to input score and putscore() to display score. The class Result is inherited from two base classes, class Test and Class Sports with data member total and a member function display() to display rollno, marks, score and the total(marks+score).
5. Write a C++ file program to store the details of employee in a file called emp.dat (empno, empname., deptno, salary). Read the contents of file emp.dat and if deptno =10 increment salary by 10% and store the new salary along with empname, empno and deptno in a new file called sal.dat.
6. Write a C++ program which implements a generic Queue ADT. Demonstrate by writing an application to show the operations such as enqueue and dequeue for char data and double data. Can this queue be given to a consumer application without the source code? Justify.

Analyze

1. Compare parameter passing mechanism using pointers and reference as arguments. Discuss the pros and cons
2. Is it possible to mix new,delete with malloc and free while writing programs. Discuss its impact.
3. Compare inline functions with macros. Discuss the merits and demerits using both.
4. Can a C++ and C structure be treated as same? Justify
5. Can a structure in C++ consider nothing but class with members as public by default. Justify
6. How does C++ file I/O library compare with its counterpart in C.Discuss how object orientation is achieved in the former?

Concept Map



Syllabus:

Introduction - Programming Paradigms - object oriented programming– **The Basics of C++** - Data types and Declarations – Expressions and Control Statements – Type Modifiers – Type Casting. Arrays- Pointers – References - Structures - Dynamic Memory Allocation – New and Delete Operators – Functions – Function prototyping – Passing parameters – Call by Reference - Overloading Functions – Default Arguments –Inline Functions – Storage Classes – Namespace.

Classes in C++ - Defining Classes – Access specifiers – Classes and Encapsulation – Member Functions – Instantiating objects – pointer and reference to objects – **Constructors** – Parameterized Constructors – Default Constructors – Copy Constructors – Conversion Constructors – Destructors. Static Object – Static Members – Constant Object – Constant Members –This Pointer

Polymorphism – Friend Function – Function Overloading – Operator Overloading - Overloading unary and binary operator – **Templates** – Function Template – Class Template - **Inheritance** - Inheritance Types – Single Inheritance - multiple inheritance - multi level inheritance - hierarchical inheritance - hybrid inheritance. Public, private and protected inheritance- Runtime polymorphism and virtual functions– Abstract Classes - Pure Virtual Functions –RTTI – Dynamic cast – Type Id

Exception Handling - Exception Handling mechanism - Throwing an exception - Try Blocks– catching an exception – multiple catch – Rethrowing the exception – User Defined Exception - Exception Specification - **Streams in C++** - Stream classes – Manipulators - File I/O streams.

Text Book

1. Ira Pohl: Oriented Programming using C++, Second Edition, Pearson Education 2001.
2. Herbert Schildt, C++ Programmer's Reference, 2nd edition, McGraw-Hill, Berkely.

Reference Books

1. BjarneStourstrup: The C++ programming language, Third Edition 2001.
2. Stanley B.Lippman: C++ Primer, Third Edition 2002.

Course Contents and Lectures schedule

No.	Topic	No. of Lectures
0	**Introduction**	
0.1	Programming Paradigms, object oriented programming	1
1	**The Basics of C++**	
1.1	Data types and Declarations, Expressions and Control Statements	2
1.2	Type Modifiers, Type Casting	1
1.3	Arrays, Pointers	2
1.4	References, Structures	2
1.5	Dynamic Memory Allocation, New and Delete Operators	1
2	**Functions**	
2.1	Function prototyping, Passing parameters, Call by Reference	2
2.2	Overloading Functions	1
2.3	Default Arguments, Inline Functions	2
2.4	Storage Classes	1
2.5	Namespace	1
3	**Classes in C++**	
3.1	Defining Classes, Access specifiers	1
3.2	Classes and Encapsulation, Member Functions	1
3.3	Instantiating objects, Pointer and Reference to objects	1
4	**Constructors**	
4.1	Parameterized Constructors	1
4.2	Default Constructors	1
4.3	Copy Constructors	1
4.4	Conversion Constructors	1
4.5	Destructors	1
4.6	Static Object, Static Members	1

4.7	Constant Object, Constant Members	1
4.8	This Pointer	1
5	**Polymorphism**	
5.1	Friend Function	1
5.2	Function Overloading	1
5.3	Operator Overloading, Overloading unary and binary operator	1
6	**Templates**	
6.1	Function Template	1
6.2	Class Template	1
7	**Inheritance**	
7.1	Inheritance Types	1
7.2	Public, private and protected inheritance	1
7.3	Runtime polymorphism and virtual functions	1
7.4	Abstract Classes	1
7.5	Pure Virtual Functions	1
7.6	RTTI - Dynamic cast, Type Id	1
8	**Exception Handling**	
8.1	Exception Handling mechanism, Throwing an exception	1
8.2	Try Blocks, Catching an exception, multiple catch	1
8.3	Rethrowing the exception	1
8.4	User Defined Exception, Exception Specification	1
9	**Streams in C++**	
9.1	Stream classes	1
9.2	Manipulators	1
9.3	File I/O streams	1
	Total Lectures	**45**

Course designer:

A. Sheik Abdullah asait@tce.edu