

**CURRICULUM FRAMEWORK AND SYLLABUS**  
**FOR OUTCOME BASED EDUCATION IN**  
**Master of Computer Applications (M.C.A) Degree Program (2Years)**  
**FOR THE STUDENTS ADMITTED FROM THE**  
**ACADEMIC YEAR 2020-2021 ONWARDS**  
**under Choice Based Credit System**



**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](http://www.tce.edu)

## **THIAGARAJAR COLLEGE OF ENGINEERING, MADURAI 625 015**

### **DEPARTMENT OF COMPUTER APPLICATIONS**

#### **VISION**

“Be the very pinnacle of academic and research excellence in Computer Applications”

#### **MISSION**

##### **As a Department, We are committed to**

- Achieve academic excellence in Computer Applications through innovative teaching and learning processes.
- To prepare the students to be professionally competent to face the challenges in the industry.
- Promote inter-disciplinary research among the faculty and the students to create state of art research facilities.
- To promote quality and ethics among the students.
- Motivate the students to acquire entrepreneurial skills to become global leaders.

#### **Programme Educational Objectives (PEO)**

Post graduates of MCA program will be

**PEO1:** Utilizing strong technical aptitude and domain knowledge to develop smart software solutions for the upliftment of society.

**PEO2:** Applying research and entrepreneurial skills augmented with a rich set of communication, teamwork and leadership skills to excel in their profession.

**PEO3:** Showing continuous improvement in their professional career through life-long learning, appreciating human values and ethics.

## Graduate Attributes for MCA Programme (GA)

### 1. Computational Knowledge:

Apply knowledge of computing fundamentals, computing specialization, mathematics, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined problems and requirements.

### 2. Problem Analysis:

Identify, formulate, research literature, and solve *complex* computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.

### 3. Design /Development of Solutions:

Design and evaluate solutions for *complex* computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.

### 4. Conduct Investigations of Complex Computing Problems:

Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

### 5. Modern Tool Usage:

Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to *complex* computing activities, with an understanding of the limitations.

### 6. Professional Ethics:

Understand and commit to professional ethics and cyber regulations, responsibilities, and norms of professional computing practice.

### 7. Life-long Learning:

Recognize the need, and have the ability, to engage in independent learning for continual development as a computing professional.

### 8. Project management and finance:

Demonstrate knowledge and understanding of the computing and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

**9. Communication Efficacy:**

Communicate effectively with the computing community, and with society at large, about *complex* computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.

**10. Societal and Environmental Concern:**

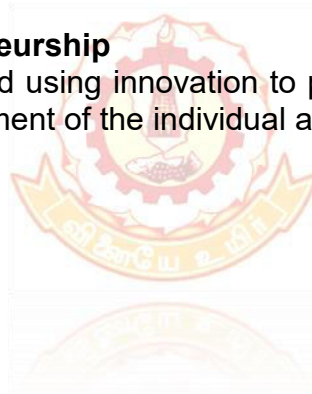
Understand and assess societal, environmental, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practice.

**11. Individual and Team Work:**

Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary environments.

**12. Innovation and Entrepreneurship**

Identify a timely opportunity and using innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society at large.



## **Programme Outcomes (PO) for Master of Computer Applications**

On completion of MCA programme, the students are expected to

- PO1:** Apply knowledge of computing fundamentals, computing specialization, mathematics, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined problems and requirements.
- PO2:** Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.
- PO3:** Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.
- PO4:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
- PO5:** Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.
- PO6:** Understand and commit to professional ethics and cyber regulations, responsibilities, and norms of professional computing practice.
- PO7:** Recognise the need, and have the ability, to engage in independent learning for continual development as a computing professional.
- PO8:** Demonstrate knowledge and understanding of the computing and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
- PO9:** Communicate effectively with the computing community, and with society at large, about complex computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.

**PO10:** Understand and assess societal, environmental, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practice.

**PO11:** Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary environments.

**PO12:** Identify a timely opportunity and using innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society at large.



**PEO-PO Mapping:**

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
PEO1												
PEO2												
PEO3												

**PO-GA MAPPING:**

	GA1	GA2	GA3	GA4	GA5	GA6	GA7	GA8	GA9	GA10	GA11	GA12
PO1												
PO2												
PO3												
PO4												
PO5												
PO6												
PO7												
PO8												
PO9												
PO10												
PO11												
PO12												

**THIAGARAJAR COLLEGE OF ENGINEERING, MADURAI – 625 015**  
**DEPARTMENT OF COMPUTER APPLICATIONS**  
**SCHEDULING OF COURSES**

Sem-ester	Theory					Theory cum Practical	Practical			Credits
1 <sup>st</sup> (25)	20CA110 Mathematical Foundations of Computer Science 3:1	20CA120 Programming in C 3:1	20CA130 Computer Organization and Architecture 4:0	20CA140 Data Structures 3:1	20CA150 Database Management Systems 4:0	-	20CA170 Data Structures using C Programming Laboratory 0:2	20CA180 RDBMS Laboratory 1:2		25
2 <sup>nd</sup> (26)	20CA210 Software Engineering 4:0	20CA220 Object Oriented Programming using C++ 3:1	20CA230 Design and Analysis of Algorithms 3:1	20CA240 Operating Systems 4:0	20CA250 Data Warehousing and Data Mining 3:1		20CA270 C++ Programming Laboratory 0:2	20CA280 Algorithms Implementation and Analysis using C Laboratory 0:2	20CA290 Professional Communication 1:1	26
3 <sup>rd</sup> (24)	20CA310 Operations Research 4:0	20CA320 Internet and Java Programming 4:0	20CA330 Object Oriented Modeling and Design Patterns 3:0	20CAPX0 Elective I 3:0	20CAPX0 Elective II 3:0		20CA370 Java Programming Laboratory 0:2	20CA380 Applications Development Laboratory 1:2	20CA390 Data Warehousing and Data Mining Laboratory 0:2	24
4 <sup>th</sup> (12)	20CA410 Project 0:12									12
Total Credits										87



**THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015****Master of Computer Applications (M.C.A) Degree Program (2 Years)****COURSES OF STUDY****(For the candidates admitted from 2020-2021 onwards)****FIRST SEMESTER**

Course code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
<b>THEORY</b>						
20CA110	Mathematical Foundations of Computer Science	BS	3	1	-	4
20CA120	Programming in C	PC	3	1	-	4
20CA130	Computer Organization and Architecture	PC	4	-	-	4
20CA140	Data Structures	PC	3	1	-	4
20CA150	Database Management Systems	PC	4	-	-	4
<b>PRACTICAL</b>						
20CA170	Data Structures using C Programming Laboratory	PC	-	-	4	2
20CA180	RDBMS Laboratory	PC	-	1	4	3
<b>Total</b>			<b>17</b>	<b>4</b>	<b>8</b>	<b>25</b>

**SECOND SEMESTER**

Course code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
<b>THEORY</b>						
20CA210	Software Engineering	PC	4	-	-	4
20CA220	Object Oriented Programming using C++	PC	3	1	-	4
20CA230	Design and Analysis of Algorithms	PC	3	1	-	4
20CA240	Operating Systems	PC	4	-	-	4
20CA250	Data Warehousing and Data Mining	PC	3	1	-	4
<b>PRACTICAL</b>						
20CA270	C++ Programming Laboratory	PC	-	-	4	2
20CA280	Algorithms Implementation and Analysis using C Laboratory	PC	-	-	4	2
20CA290	Professional Communication	HSS	-	1	1	2
<b>Total</b>			<b>17</b>	<b>4</b>	<b>9</b>	<b>26</b>

**THIRD SEMESTER**

Course code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
<b>THEORY</b>						
20CA310	Operations Research	BS	4	-	-	4
20CA320	Internet and Java Programming	PC	4	-	-	4
20CA330	Object Oriented Modeling and Design Patterns	PC	3	-	-	3
20CAPX0	Elective I	PE	3	-	-	3
20CAPX0	Elective II	PE	3	-	-	3
<b>PRACTICAL</b>						
20CA370	Java Programming Laboratory	PC	-	-	4	2
20CA380	Applications Development Laboratory	PC	-	1	4	3
20CA390	Data warehousing and Data Mining Laboratory	PC	-	-	4	2
<b>Total</b>			<b>17</b>	<b>1</b>	<b>12</b>	<b>24</b>

**FOURTH SEMESTER**

Course code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
<b>PRACTICAL</b>						
20CA410	Project	PC	-	-	24	12
<b>Total</b>			<b>24</b>			<b>12</b>

BS : Basic Sciences  
 PC : Programme Core  
 PE : Programme Elective  
 HSS : Humanities and Social Sciences

L : Lecture  
 T : Tutorial  
 P : Practical

**Note:**

1 Hour Lecture/week is equivalent to 1 credit

1 Hour Tutorial/week is equivalent to 1 credit

2 Hour Practical/week is equivalent to 1 credit

**Total credits to be earned for the award of degree: 87**

**THIAGARAJAR COLLEGE OF ENGINEERING, MADURAI – 625 015****Master of Computer Applications (M.C.A) Degree Program****SCHEME OF EXAMINATIONS  
(For the candidates admitted from 2020-2021 onwards)****FIRST SEMESTER**

S.No.	Sub. Code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum	Marks for
				Continuous Assessment *	Terminal Exam **	Max. Marks	Pass	Total
<b>THEORY</b>								
1	20CA110	Mathematical Foundations of Computer Science	3	50	50	100	25	50
2	20CA120	Programming in C	3	50	50	100	25	50
3	20CA130	Computer Organization and Architecture	3	50	50	100	25	50
4	20CA140	Data Structures	3	50	50	100	25	50
5	20CA150	Database Management Systems	3	50	50	100	25	50
<b>PRACTICAL</b>								
6	20CA170	Data Structures using C Programming Laboratory	3	50	50	100	25	50
7	20CA180	RDBMS Laboratory	3	50	50	100	25	50

**SECOND SEMESTER**

S.No.	Sub. Code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
<b>THEORY</b>								
1	20CA210	Software Engineering	3	50	50	100	25	50
2	20CA220	Object Oriented Programming using C++	3	50	50	100	25	50
3	20CA230	Design and Analysis of Algorithms	3	50	50	100	25	50
4	20CA240	Operating Systems	3	50	50	100	25	50
5	20CA250	Data Warehousing and Data Mining	3	50	50	100	25	50
<b>PRACTICAL</b>								
6	20CA270	C++ Programming Laboratory	3	50	50	100	25	50
7	20CA280	Algorithms Implementation and Analysis using C Laboratory	3	50	50	100	25	50
8	20CA290	Professional Communication	3	50	50	100	25	50

**THIRD SEMESTER**

S.No.	Sub. code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
<b>THEORY</b>								
1	20CA310	Operations Research	3	50	50	100	25	50
2	20CA320	Internet and Java Programming	3	50	50	100	25	50
3	20CA330	Object Oriented Modeling and Design Patterns	3	50	50	100	25	50
4	20CAPX0	Elective I	3	50	50	100	25	50
5	20CAPX0	Elective II	3	50	50	100	25	50
<b>PRACTICAL</b>								
6	20CA370	Java Programming Laboratory	3	50	50	100	25	50
7	20CA380	Applications Development Laboratory	3	50	50	100	25	50
8	20CA390	Data warehousing and Data Mining Laboratory	3	50	50	100	25	50

**FOURTH SEMESTER**

S.No.	Sub. code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment *	Terminal Exam **	Max. Marks	Terminal Exam	Total
<b>PRACTICAL</b>								
1	20CA410	Project	3	150	150	300	75	150

\* Continuous Assessment (CA) 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 100/300 and subsequently be reduced to 50/150 marks for the award of terminal examination marks

**LIST OF ELECTIVE COURSES OFFERED  
FOR THE M.C.A DEGREE PROGRAMME**

Sub. Code	Sub. Name	Category	No. of Hours / Week			Credits
			L	T	P	
20CAPA0	Electronic Commerce and Electronic Business	PE	3	-	-	3
20CAPB0	Business Analytics	PE	3	-	-	3
20CAPC0	Software Project Management	PE	3	-	-	3
20CAPD0	Machine Learning	PE	3	-	-	3
20CAPE0	Data Analytics	PE	3	-	-	3
20CAPF0	Programming In C# Using .Net	PE	3	-	-	3
20CAPG0	Cloud Computing	PE	3	-	-	3
20CAPI0	Internet of Things	PE	3	-	-	3
20CAPJ0	Cyber Security and law	PE	3	-	-	3
20CAPK0	Software Quality and Testing	PE	3	-	-	3

**MCA CURRICULUM 2020-2021 ONWARDS****CHOICE BASED CREDIT SYSTEM****Credit Distribution:**

<b>S.No.</b>	<b>Category of courses</b>	<b>Credits</b>	<b>Percentage of Credits to Total Credits</b>
1	Foundation Courses	8	9.2%
2	Professional Core- Theory	43	48.4%
3	Professional Core - Practical	16	18.4%
4	Professional Electives	6	6.9%
5	Employability Enhancement Courses	14	16.1%
<b>Total Credits</b>		<b>87</b>	<b>100%</b>

**Foundation courses (FC):**

<b>Course code</b>	<b>Name of the Course</b>	<b>Category</b>	<b>No. of Hours / Week</b>			<b>credits</b>
			<b>L</b>	<b>T</b>	<b>P</b>	
<b>THEORY</b>						
20CA110	Mathematical Foundations of Computer Science	FC –BS	3	1	-	4
20CA310	Operations Research	FC-BS	4	0	-	4

**Professional Core (PC):**

Course code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
20CA120	Programming in C	PC	3	1	-	4
20CA130	Computer Organization and Architecture	PC	4	-	-	4
20CA140	Data Structures	PC	3	1	-	4
20CA150	Database Management Systems	PC	4	-	-	4
20CA170	Data Structures using C Programming Laboratory	PC	-	-	4	2
20CA180	RDBMS Laboratory	PC	-	1	4	3
20CA210	Software Engineering	PC	4	-	-	4
20CA220	Object Oriented Programming using C++	PC	3	1	-	4
20CA230	Design and Analysis of Algorithms	PC	3	1	-	4
20CA240	Operating Systems	PC	4	-	-	4
20CA250	Data Warehousing and Data Mining	PC	3	1	-	4
20CA270	C++ Programming Laboratory	PC	-	-	4	2
20CA280	Algorithms Implementation and Analysis using C Laboratory	PC	-	-	4	2
20CA320	Internet and Java Programming	PC	4	-	-	4
20CA330	Object Oriented Modeling and Design Patterns	PC	3	-	-	3
20CA370	Java Programming Laboratory	PC	-	-	4	2
20CA380	Applications Development Laboratory	PC	-	2	4	3
20CA390	Data warehousing and Data Mining Laboratory	PC	-	-	4	2



**Professional Electives (PE):**

Course code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
20CAPA0	Electronic Commerce and Electronic Business	PE	3	-	-	3
20CAPB0	Business Analytics	PE	3	-	-	3
20CAPC0	Software Project Management	PE	3	-	-	3
20CAPD0	Machine Learning	PE	3	-	-	3
20CAPE0	Data Analytics	PE	3	-	-	3
20CAPF0	Programming In C# Using .Net	PE	3	-	-	3
20CAPG0	Cloud Computing	PE	3	-	-	3
20CAPI0	Internet of Things	PE	3	-	-	3
20CAPJ0	Cyber Security and law	PE	3	-	-	3
20CAPK0	Software Quality and Testing	PE	3	-	-	3

**Employability Enhancement Courses (EEC):**

Course code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
20CA290	Professional Communication	EEC-HSS	-	1	1	2
20CA410	Project	EEC	-	-	24	12

BS- Basic Sciences; HSS – Humanities and Social Sciences

<b>20CA110</b>	<b>MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE</b>	Category	L	T	P	Credit
		BS	3	1	0	4

### Preamble

A Computer Application student needs to have some basic mathematical tools and techniques to understand various designing concepts, storage methods, concepts in digital principles, managing databases etc. The main objective of this course is to introduce the basic terminology used in advanced courses in Computer application. This emphasizes the development of rigorous logical thinking for solving different kinds of problems that occur in computer applications. Based on this the course aims at giving adequate exposure in the theory and applications of Set theory, Propositional logic, Predicate logic, Lattices and Boolean Algebra, Automata theory which helps the learner to use them eventually in practical applications of computer science. These topics support the advanced courses in computer science such as digital principles, artificial intelligence, compiler and design, DBMS, Design of Software etc.

### Prerequisite

Higher Secondary Level, Degree Level –Set Theory, Logic Theory

### Course Outcomes

On the successful completion of the course, students will be able to	
CO1: Prove implication problems using truth table method, replacement process, Analyzation method, truth table technique, rules of inference method	Apply
CO2: Obtain PCNF and PDNF of given logical expression	Apply
CO3: Check the validity of the verbal or symbolic arguments using rules of inference	Apply
CO4: Construct verbal arguments with predicates in symbolic form and also to validate them	Apply
CO5: Represent the given relation in matrix, digraph and vice versa	Apply
CO6: Verify a given function is bijective or not, and also to find composition of functions	Apply
CO7: Design Karnaugh map to get simplified form of a Boolean function	Apply
CO8: Check whether the given grammar is regular or not using pumping lemma	Apply

### Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	S	S	M	L	M	S	S	-	-	-
CO2	S	S	S	M	M	L	M	M	S	-	-	-
CO3	S	S	S	M	M	L	L	M	S	-	-	-
CO4	S	S	S	M	M	L	L	M	S	-	-	-
CO5	S	S	S	M	M	L	S	M	L	-	-	-
CO6	S	S	S	S	M	L	L	M	S	-	-	-
CO7	S	S	S	S	M	L	L	M	S	-	-	-
CO8	S	S	S	S	M	L	L	M	S	-	-	-

S- Strong; M-Medium; L-Low

### Assessment Pattern

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	10	10	10	-
Understand	30	30	20	30
Apply	60	60	70	70
Analyse	0	0	0	0
Evaluate	0	0	0	0
Create	0	0	0	0

### Course Level Assessment Questions

**Prove implication problems using truth table method, replacement process, Analyzation method, truth table technique, rules of inference method (CO1):**

1. Prove the following implication by analyzation method.  
 $(PVQ) \wedge (P \rightarrow R) \wedge (Q \rightarrow S) \Rightarrow S \vee R$
2. Show that  $(QV(P \wedge \neg q) \vee (\neg P \wedge \neg Q))$  is a tautology using replacement process.
3. Verify the following implication by truth table.  
 $(P \rightarrow (Q \rightarrow R)) \Rightarrow ((P \rightarrow Q) \rightarrow (P \rightarrow R))$
4. Show the implication using CP rule if necessary.  $\neg PVQ, \neg QVR, R \rightarrow S \Rightarrow P \rightarrow S$ .

**Obtain PCNF and PDNF of given logical expression (CO2):**

1. Obtain CNF and DNF of  $\neg(PVQ) \leftrightarrow (P \wedge Q)$
2. Obtain PCNF and PDNF of  $(\neg P \rightarrow R) \wedge (Q \leftrightarrow P)$
3. Obtain principal disjunctive normal form of  $P \rightarrow ((P \rightarrow Q) \wedge \neg(\neg Q \vee \neg P))$  and hence obtain principal conjunctive normal form.

**Check the validity of the verbal or symbolic arguments using rules of inference(CO3):**

1. Show that the following system of premises is inconsistent.  
 If war is near, then the army would be mobilized. If the army has mobilized then labour costs are high. However the war is near and yet labour costs are not high.

- Pick out free and bound variables, scope of the quantifier from the following expression  $(\forall x)[P(x) \rightarrow (\exists y)Q(x, y)]$  and also get free and bound occurrence of the variables involved in the expression.
- Prove by indirect method the following implication:  
 $(\forall x)(P(x) \rightarrow Q(x)); (\exists y)P(y) \Rightarrow (\exists z)Q(z)$

**Construct verbal arguments with predicates in symbolic form and also to validate them (CO4):**

- Verify the validity of the following arguments:  
 Everyone chooses between good and evil. Rishi has chosen not to do evil. If anyone chooses to do good or if he is forced to obey the laws then he has an excellent chance for happiness. Therefore, Rishi chances for happiness are excellent.
- Show that from (i)  $(\exists x)(F(x) \wedge S(x)) \rightarrow (\forall y)(M(y) \rightarrow W(y))$  ;  
 (ii)  $(\exists y)(M(y) \wedge \neg W(y))$  the conclusion  $(\forall x)(F(x) \rightarrow \neg S(x))$

**Represent the given relation in matrix, digraph and vice versa. (CO5):**

- Let R denote a relation on the set of ordered pairs of integers such that  $\langle x, y \rangle R \langle u, v \rangle$  iff  $xv=yu$ . Show that R is an equivalence relation .
- Given  $A = \{1,2,3,4\}$  and  $R = \{(1,2), (1,1), (1,3), (2,4)\}$ ,  $S = \{(1,4), (1,3), (2,3), (3,1), (4,1)\}$  are relations on A. Find  $S \circ R, R \circ S, M_R, M_S, M_{(R \circ S)^{-1}}$  and graph of R,S.
- Discuss about all types of relations on the set  $A = \{1,2,3,4\}$  where R is given by  $R = \{(1,1), (2,2), (2,3), (3,2), (3,3)\}$ .

**Verify a given function is bijective or not, and also to find composition of functions.(CO6):**

- Verify which of the following functions are bijective where  $f, g : R \rightarrow R$ ,  $f(x) = -5x$ ,  $g(x) = x^2 - 8$ , where  $f, g : R \rightarrow R$ , hence find  $f \circ g, g \circ f, f \circ f, g \circ g$
- Let  $X = \{1,2,3,4\}$  and a mapping  $f : X \rightarrow X$  be given by  $f = \{(1,2), (2,3), (3,4), (4,1)\}$ . Execute  $f^3, f^4$ .
- 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  $f \circ g \circ h$ .

**Design Karnaugh map to get simplified form of a Boolean function (CO7):**

- Prove that  $D_{70}$  is a Lattice and also verify  $D_{70}$  as Distributive Lattice.
- Simplify the Boolean function  $f(a,b,c,d) = \sum(0,2,6,7,8,9,13,15)$  using Karnaugh map.
- Discuss algebraically the equality  $\overline{ab} + \overline{bc} + \overline{ca} = \overline{ab} + \overline{bc} + \overline{ca}$  using basic laws of Boolean algebra.

**Check whether the given grammar is regular or not using pumping lemma. (CO8):**

- Construct a DFA that accepts all the strings on  $\{0,1\}$  except those containing the substring 101.
- Define context free grammar and ambiguous grammar and hence identify the language generated by the grammar  $S \rightarrow aSb/ab$
- Convert the following NFA to DFA.

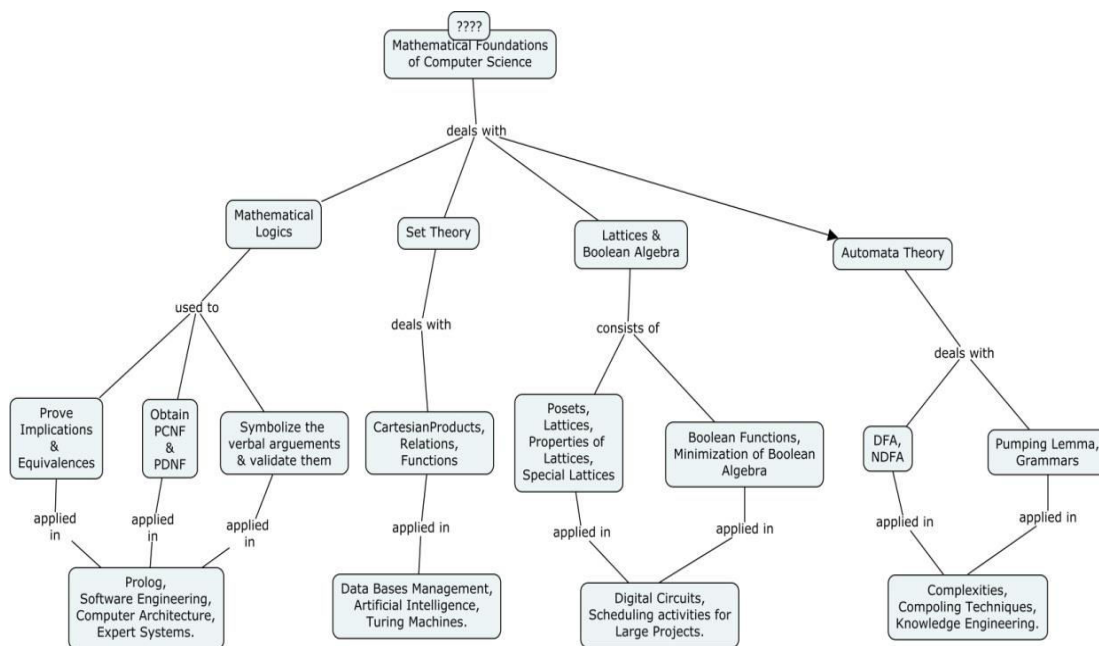
$\delta$	a	b
$S_0$	$\{S_0, S_1\}$	$\phi$
$S_1$	$\phi$	$\{S_2\}$
$S_2$	$\phi$	$\{S_2\}$

- (i) Draw transition diagram of N DFA.  
(ii) Draw transition diagram of DFA with its state table.

4. Find whether the following languages are regular or not using pumping lemma:

(i)  $L = \{0^{2n} / n \geq 1\}$       (ii)  $L = \{1^{n^2} / n \geq 1\}$

### Concept Map



### Syllabus

#### Mathematical logic:

##### Propositional Logic:

Introduction – statements and notation, connectives- Conjunction, Disjunction, Negation, Conditional and biconditional, -Implications and Equivalence, Tautology and Contradictions, Normal forms: Conjunctive Normal Form, Disjunctive Normal Form - Principal Conjunctive Normal Forms - Principal Disjunctive Normal Form, Rules of Inference: P, T, CP, AP rules – Consistency of premises. Validity by truth table technique. **Predicate Logic:** Predicates- Statement Function, Variables and Quantifiers, Predicate formulas – Free and Bound Variables, -Theory of inferences on one place predicate using P, T, CP rules.

##### Set Theory :

Basic Concepts of set theory and Cartesian products , Relations, Binary relations, Equivalence relations and Partitions, Composition of relations. **Functions:** Types of functions, Inverse of a function, Composition of functions, Recursive functions.

##### Lattices & Boolean Algebra:

**Partially ordered set :** Definition of Partially ordered set (PO Set), Hasse Diagram, LUB, GLB, Meet and Join of elements of PO set.

**Lattices as partially ordered sets :** Definition and basic properties of lattices, Sub lattices, Special lattices.

**Boolean Algebra:** – Definition and examples – Boolean functions — Minimization of Boolean functions.

**Automata Theory and Grammar :**

Deterministic and Non-Deterministic finite Automaton, NFA to DFA, NFA with  $\epsilon$ -moves, Regular language and Regular Expression, NFA and Regular Expressions, Pushdown Automaton, Introduction to Turing Machine.

Pumping Lemma (without proof) and its applications, Grammar, Types of Grammars – Language to Grammar – Grammar to Language.

**Reference Books**

1. Trembly and Manohar, "Discrete mathematical structures with applications to Computer Science", Tata McGrawHill, 2002.
2. Kenneth H. Rosen, "Discrete mathematics and its applications", McGrawHill International Editions 1999.
3. Dr. M.K.Venkataraman., Dr.N.Sridharan and N.Chandrasekaran, Discrete Mathematics, National Publishing Company, Chennai. of India (2004)
4. John E.Hopcraft, Rajeev Motwani, Jeffery D.Ullman, "Introduction to Automata Theory, Languages and Computation", Pearson Education, Asia, 2001.
5. John C.Martin, "Introduction to Languages and the theory of Computation", Tata McGraw-Hill Publishing Company Limited, New Delhi.
6. [http://www.research.ibm.com/haifa/dept/svt/papers/Mathematical\\_Logic.pdf](http://www.research.ibm.com/haifa/dept/svt/papers/Mathematical_Logic.pdf)
7. Mathematical Logic and its Application to Computer Science - Lecture Notes EitanFarchi, Ben-Chaim, March 3, 2010

**Course Contents and Lecture Schedule**

Module No	Topics	No. of Lectures
<b>Module 1: Propositional and Predicate Calculus</b>		
1.1	Introduction – Statements and Notations, Basic Connectives, Conditional, Biconditional	1
1.2	Implications and Equivalence, Tautology and Contradiction	1
1.3	Normal forms: Conjunctive Normal Forms - Disjunctive Normal Form	1
1.4	Principal Conjunctive Normal Forms - Principal Disjunctive Normal Form	1
1.5	The theory of inference for Propositional Calculus using Rule P, T	2
1.6	The theory of inference for Propositional Calculus using Rule CP, AP, Inconsistent	2
1.7	Predicates- Statement Function, Variables and Quantifiers	2
1.8	Predicate formulas – Free and Bound Variables	1
1.9	Theory of inferences on one place predicate using P, T, CP rules.	3
<b>Module II- Set Theory</b>		
2.1	Basic Concepts of set theory and Cartesian products	1
2.2	Relations, Binary Relations	1
2.3	Equivalence Relations and Partitions, Composition of Relations	2
2.3	Functions and types of functions	1
2.4	Inverse of a function and Composition of functions, Recursive functions	2

<b>Module III :Lattices and Boolean Algebra</b>		
3.1	Definition of Partially ordered set(PO Set), Hasse Diagram, LUB, GLB, Meet and Join	1
3.21	Lattice as PO	1
3.3	Properties of lattices	1
3.4	Special Lattice	2
3.5	Boolean Algebra and its properties	1
3.6	Boolean Function	1
3.7	Value of Boolean Expression	1
3.8	Minimization of Boolean functions	2
<b>Module IV : Automata Theory</b>		
4.1	Deterministic and Non-Deterministic finite automaton	1
4.2	NFA to DFA	1
4.3	NFA with $\epsilon$ -moves	1
4.4	Regular expressions and regular language	1
4.5	NFA and Regular Expressions	1
4.6	Pushdown automaton	1
4.7	Introduction to Turing machine	1
4.8	Pumping Lemma (without proof) and its applications	1
4.9	Grammars and its types	1
4.10	Grammar to Language	1
4.11	Language to Automata	1
	<b>Total</b>	<b>42</b>

**Course Designer:**

Dr. M. Kameswari [mkmat@tce.edu](mailto:mkmat@tce.edu)

**20CA120****PROGRAMMING IN C**

Category	L	T	P	Credit
PC	3	1	-	4

**Preamble**

This is a course offered in first semester for the students of Computer Applications. This course has four credits dedicated to provide the students a strong foundation on programming concepts and its application. It also enables the students to solve problems using programmable logic.

**Prerequisite**

Fundamentals of Programming languages

**Course Outcomes**

On the successful completion of the course, students will be able to

- CO1: Describe the reason why different constructs are available for iteration, such as "for" loops, "do...while" loops Understand
- CO2: Demonstrate the difference between iteration and recursion in terms of C programming Analyze
- CO3: Develop C programs for arrays and linked lists Apply
- CO4: Develop C programs for Data structure concept with functions Apply
- CO5: Summarize the Hardware interaction using Port I/O Understand
- CO6: Develop C programs for File Management concept Apply

**Mapping with Programme Outcomes**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	P11	PO12
CO1	L											
CO2	S	S	S	M	S		M			M		
CO3	S	S	S	M	S				M			
CO4	S	S	S	M	S		M					
CO5	M						M			M		
CO6	S	S	S	M	S							

S- Strong; M-Medium; L-Low

**Assessment Pattern**

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	20	20	10	10
Understand	20	20	20	20
Apply	50	50	50	50
Analyse	10	10	20	20
Evaluate	0	0	0	0
Create	0	0	0	0



**Course Level Assessment Questions**

**Describe the reason why different constructs are available for iteration, such as "for" loops, "do...while" loops (CO1):**

1. Differentiate between Structure and Union in C.
2. Explain how dynamic arrays are efficient compared to Static with example?
3. How is memory managed in C?
4. How garbage collection is done in C?

**Demonstrate the difference between iteration and recursion in terms of C programming (CO2):**

1. Explain how recursive functions affect the run time efficiency?
2. Is there any advantage of using recursion over looping control structures? Give a suitable example?
3. Analyze the factors that influence the execution times of a program?
4. Illustrate the limitation of array of pointers to strings using a sample?
5. Differentiate keywords BREAK and CONTINUE with an example?

**Develop C programs for arrays and linked lists concepts (CO3) :**

1. Write C code to implement linked list.
2. Compare the scenarios of usage of single dimension array and two dimensional array.
3. Implement the search of an element in array and linked list.

**Develop C programs for Data structure concepts with functions (CO4) :**

1. Write a program to perform stack operation using pointers?
2. Write a program to perform linked list operation using pointers?
3. Write a program to compute Matrix Multiplication using Pointers?

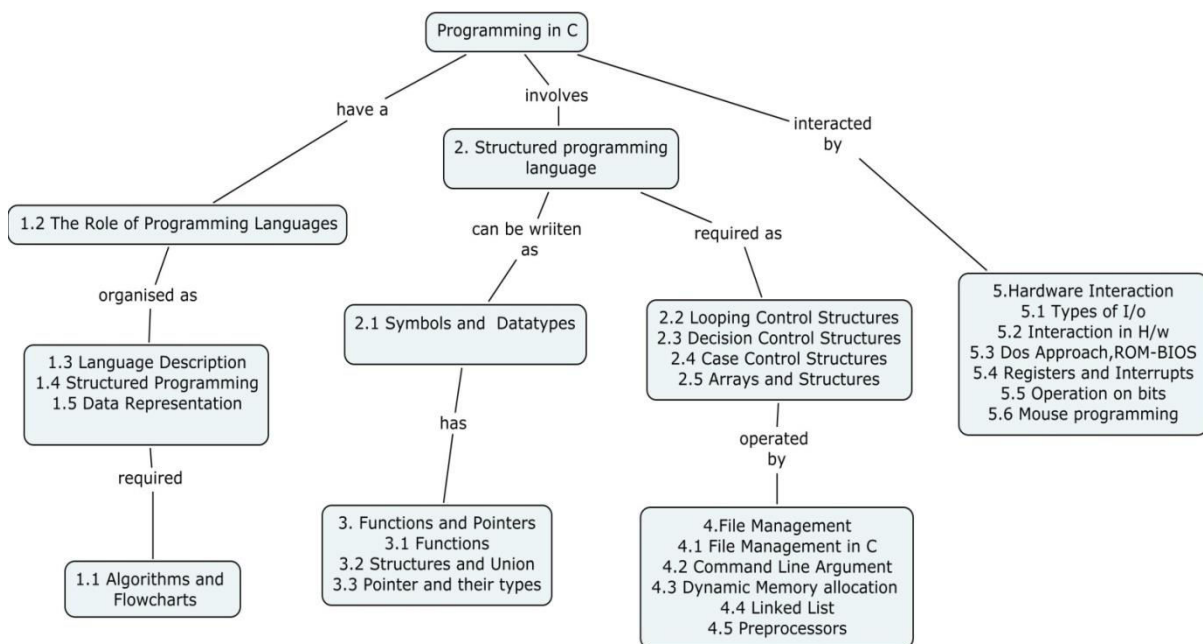
**Summarize the Hardware interaction using Port I/O (CO5) :**

1. Explain how interrupts can work in Hardware programming
2. Illustrate the usage of mouse programming
3. Illustrate the usage of init86( )

**Develop C programs for File Management concept (CO6):**

1. Write a program to add the numbers present in two files.
2. Write a program to Search the elements in the file
3. Write a program to remove the record in a file.

## Concept Map



## Syllabus

**Introduction to Programming Language** : Algorithms, Flowcharts, The Role of Programming Languages, Language Description, Structured Programming, Data Representation, Procedure Activations **Structured Programming Language**: Symbols and data types, Looping control structures, Decision control structures, Case control structures, Arrays and Strings **Functions and Pointers**: Functions, Structures, Union, Pointers, Type of Pointer **File Management**: File Management in C, Command Line Argument, Dynamic Memory allocation, Linked List and Preprocessors **Hardware Interface** : Types of I/O, Interaction with H/W in C, CPU Registers, Interrupts, DOS Function Requests, Interaction with HW using Port I/O, Operation on bits, Mouse Programming.

## Reference books

1. Brian W Kernighan & Dennis Ritchie, "The C programming language", 2nd Edition, Prentice Hall ,2015
2. Yashavant Kanetkar," Let us C", BPB Publications 8th Edition, 2014
3. Darnell and Margolis, "ANSI C- A Systematic programming Approach", Narosa publications, 2010.
4. Ravi Sethi, Viswanatha. K.V "Programming Languages – Concepts & Constructs" , Pearson Education, Second Edition, 2007
5. E. Balagurusamy, "Programming in ANSI C", Tata McGraw Hill, 2012
6. Reema. Thareja, "Programming in C", Oxford University Press, 2nd Edition,2016

### Course Contents and Lecture Schedule

Module No.	Topic	No of Lectures
1	<b>Introduction to Programming Language</b>	
1.1	Algorithms, Flowcharts	1
1.2	The Role of Programming Languages	1
1.3	Language Description	2
1.4	Structured Programming	2
1.5	Data Representation	2
2	<b>Structured Programming Language</b>	
2.1	Symbols and data types	1
2.2	Looping control structures	2
2.3	Decision control structures	2
2.4	Case control structures	2
2.5	Arrays and Strings	2
3	<b>Functions and Pointers</b>	
3.1	Functions	3
3.2	Structures, Union	1
3.3	Pointers, Type of Pointer	3
4	<b>File Management</b>	
4.1	File Management in C	2
4.2	Command Line Argument	1
4.3	Dynamic Memory allocation	2
4.4	Linked List	2
4.5	Preprocessors	1
5	<b>Hardware Interface</b>	
5.1	Types of I/O	1
5.2	Interaction with H/W in C	1
5.3	DOS Approach, ROM – BIOS	1
5.4	Registers, Interrupts	1
5.5	Operation on bits	2
5.6	Mouse Programming	2
	Total	<b>40</b>

### Course Designer

Dr.D.Anitha      anithad@tce.edu

<b>20CA130</b>	<b>COMPUTER ORGANIZATION AND ARCHITECTURE</b>	Category	L	T	P	Credit
		PC	4	0	0	4

**Preamble**

This course is dedicated to number system, logic design, and memory and processing. This is the only course that is concerned with the hardware of a computer, its logic design and organization. It aims at making the student familiar with digital logic and functional design of arithmetic and logic unit that is capable of performing floating point arithmetic operations. .

**Prerequisite**

None

**Course Outcomes**

On the successful completion of the course, students will be able to

CO1: Explain about computer architecture	Understand
CO2: Compute simple arithmetic operations for fixed-point and floating-point addition, subtraction, multiplication & division.	Apply
CO3: Design combinational and sequential digital functions	Apply
CO4: Construct an instruction set capable of performing a specified set of operations	Analyze
CO5: Demonstrate a memory system for a given set of specifications	Analyze
CO6: Explain pipelining concepts	Understand
CO7: Compare the different ways of communicating with I/O devices and standard I/O interfaces.	Analyze

**Mapping with Programme Outcomes**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S		S		M							
CO2		S				M	L					
CO3	M							S				
CO4			S		M		M					
CO5	S					S			L			
CO6		M				M						
CO7				S				M				

S-Strong; M-Medium; L-Low

### Assessment Pattern

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	10	10	10	15
Understand	20	20	20	15
Apply	40	40	30	30
Analyse	30	30	40	40
Evaluate	0	0	0	0
Create	0	0	0	0

### Course Level Assessment Questions

#### Explain about computer architecture (CO1):

1. What are the different units in a computer system?
2. What is an addressing mode?
3. What is an assembly language instruction?
4. What is the difference between machine language and assembly language?

#### Compute simple arithmetic operations for fixed-point and floating-point addition, subtraction, multiplication & division. (CO2):

1. Convert the decimal number 92.00625 into a binary number?
2. Multiply the number  $-13 * 11$  using (a) sequential circuit binary multiplier (b) Booth Algorithm
3. Multiply the binary numbers  $1010 \times 0111$  using any algorithm you know? Show all steps in calculation
4. Convert 1.7 to IEEE 32 bit floating point format.
5. Predict the Multiplier bit-pair recording table and give example of bit pair recording of 11010

#### Design combinational and sequential digital functions (CO3):

1. Obtain an algorithm to find all allowable weights for a weighted BCD code? Assume that all weights are positive integers.
2. Give a combinational circuit to multiply two numbers each of which is 3 bits long and has 1 bit sign? The output should have the right sign and magnitude.

#### Construct instruction set capable of performing a specified set of operations (CO4):

1. Analyze the characteristics, organization and operation modes of Asynchronous and Synchronous DRAMs in Detail?
2. Write a program for adding n numbers using straight line sequencing and branching statements.

#### Demonstrate a memory system for a given set of specifications (CO5):

1. Explain the connection between the processor and memory with neat diagram.

2. Explain in detail about Cache memory.
3. Define RAM and explain its types.

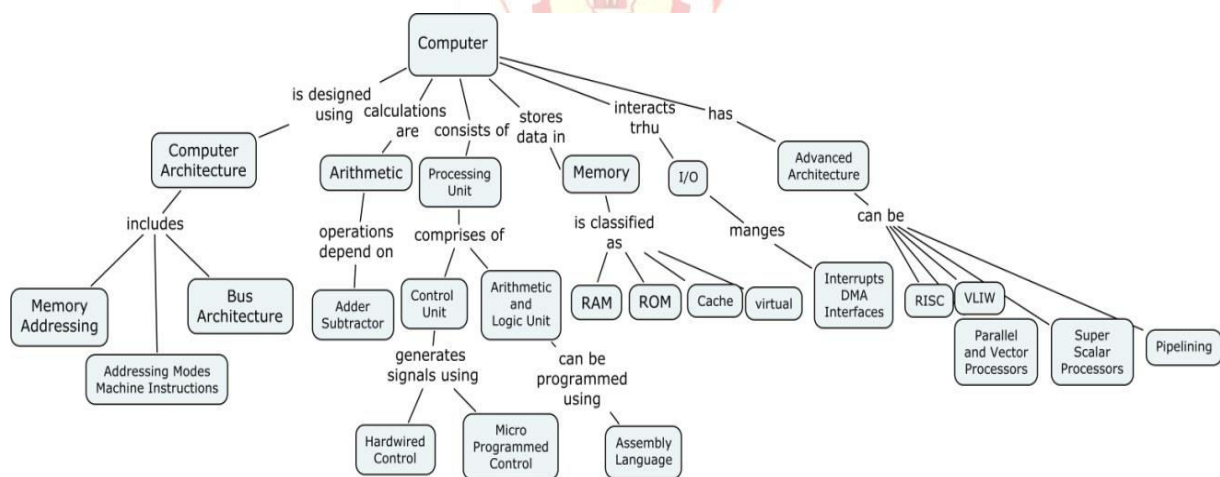
### Explain pipelining concepts (CO6):

1. Classify Hazards and analyze their impact on Pipelining
2. A pipelined processor has two branch delay slots. An optimizing compiler can fill one of these slots 85 % of the time and can fill the second slot only 20% of the time. What is the percentage improvement in performance achieved by this optimization, assuming that 20% of the instructions executed are branch instructions?
3. Classify Hazards and analyze their impact on Pipelining

### Compare the different ways of communicating with I/O devices and standard I/O interfaces. (CO7):

1. Draw a neat diagram of I/O Interface for input device and describe it.
2. Illustrate the Input and output interface Circuit in parallel port
3. What are the needs for input-output interface? Explain the functions of a typical 8-bit parallel interface in detail.

### Concept Map



### Syllabus

**Functional Units:** Basic operational concepts, Bus structures, Machine instructions, memory locations, addressing modes, assembly language **Arithmetic:** Number representations, addition and subtraction of signed numbers, Design of fast adders, Multiplication of signed numbers, Fast multiplication and Integer division **Processing Unit:** Concepts, Execution of complete instruction, Multi bus organization, ALU; Control Unit: Hardwired Control, Micro programmed Control; Micro Instructions, Micro program sequencing, Micro instructions with next address field and pre-fetching **Memory:** RAM, ROM, Cache Memories, and Virtual memory **Input and output organization:** Accessing I/O devices, Interrupts, DMA, and Interface circuits **Advanced Processor Architecture:** RISC, Pipelining, Super Scalar Processors, VLIW, Parallel and Vector Processors.

### Reference books

1. Carl Hamacher, Zvonko Vranesic, safwat Zaky, "Computer Organization and Embedded Systems", Sixth Edition, Tata McGraw Hill, 2011.
2. William Stallings, "Computer Organization and Architecture", Tenth Edition, Pearson Education, 2015.
3. David A. Patterson, John L.Hennessy, "Computer Organization and Design", Fourth Edition, Morgan Kauffmann Publishers, 2011.

### Course Contents and Lecture Schedule

Module No	Content	No of Periods
	<b>Functional Units</b>	
1.1	Basic structure of Computers: Functional Units, Operational Concepts	1
1.2	Bus Structures	1
1.3	Machine instructions and programs	2
1.4	Memory locations and Addresses, Memory operations	1
1.5	Instructions and instruction sequencing	1
1.6	Addressing Modes	1
	<b>Arithmetic</b>	
2.1	Number Representation	1
2.2	Addition and Subtraction of signed numbers	1
2.3	Design of Fast Adders	2
2.4	Multiplication of numbers	1
2.5	Operand multiplication	1
2.6	Fast multiplication	1
2.7	Integer division	1
2.8	Floating point numbers and operations	1
	<b>Processing Unit</b>	
3.1	Processing Unit: Fundamental Concepts(Seminar)	1
3.2	Execution of complete Instruction	1
3.3	Multiple bus organization	2
3.4	Hardwired Control	1
3.5	Micro programmed control	2
3.6	Micro instructions with next address field	1
	<b>Memory</b>	
4.1	Main memory: Concepts	1

4.2	Semiconductor RAM memories	2
4.3	Read Only Memory	1
4.4	Cache Memories	1
4.5	Virtual Memories	2
4.6	Memory Management Requirements(Seminar)	2
	<b>Input and output organization</b>	
5.1	Accessing I/O devices	1
5.2	Interrupts	2
5.3	Direct Memory Access & interface circuits	2
	<b>Advanced Processor Architecture</b>	
6.1	RISC	1
6.2	Pipelining	2
6.3	Super Scalar Processors	1
6.4	Parallel and Vector Processors	1
6.5	VLIW	1
	<b>Total</b>	<b>44</b>

**Course Designer:**

Dr.T.Chandrakumar

tckcse@tce.edu



**20CA140****DATA STRUCTURES**

Category	L	T	P	Credit
PC	3	1	0	4

**Preamble**

This course aims at facilitating the students to understand the various data structures, their organization and operations. The course helps the students to assess the applicability of different data structures and associated algorithms to real world problems.

**Prerequisite**

None

**Course Outcomes**

On the successful completion of the course, students will be able to

- CO1: Explain the organization and operations of data structures Stack, Queues, Trees, Graphs, Heaps and Hash tables. Understand
- CO2: Compare and contrast the functionalities and applications of different data structures Understand
- CO3: Demonstrate specific search and sort algorithms using data structures given specific user requirements. Apply
- CO4: Apply the operations of data structures in designing software procedures based on specific requirements Apply
- CO5: Assess the applicability of given data structures and associated operations to real time computer applications Evaluate
- CO6: Identify suitable algorithms with appropriate data structures for real time software requirements Apply
- CO7: Modify the existing operations of data structures for changing needs of the software requirements Analyze

**Mapping with Programme Outcomes**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	P12
CO1	M	-	-	-	-	-	-	-	-	--	-	-
CO2	M	L	-	-	-	-	-	-	-	-	-	-
CO3	S	L	-	-	-	-	-	-	-	-	-	-
CO4	S	L	M	L	L	-	M	-	-	-	-	-
CO5	S	L	M	M	L	-	S	-	-	-	-	-
CO6	S	L	M	M	-	-	S	-	-	-	-	-
CO7	S	L	M	M	L	-	S	-	-	-	-	-

S- Strong; M-Medium; L-Low

### Assessment Pattern

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	30	20	10	10
Understand	30	20	20	20
Apply	30	30	40	40
Analyse	10	20	20	20
Evaluate	0	10	10	10
Create	0	0	0	0

### Course Level Assessment Questions

**Explain the organization and operations of data structures Stack, Queues, Trees, Graphs, Heaps and Hash tables. (CO1):**

1. State the necessity of data structures.
2. List out the areas in which data structures are applied extensively.
3. List out few of the applications of tree data-structure.
4. Write down the pseudocode to implement stack using arrays..
5. Explain the purpose of hash tables.

**Compare and contrast the functionalities and applications of different data structures (CO2):**

1. Explain why B trees are used in RDBMS.
2. Analyze whether a queue can be constructed with stack.
3. Justify the use of adjacency list representation in representing sparse graph.
4. How many different trees are possible with 10 nodes?
5. If the balance factor of a node in AVL tree is -2, what is the next sequence of steps?

**Demonstrate specific search and sort algorithms using data structures (CO3):**

1. A hash table with hash function  $H_1(k) = k \bmod 13$  is shown below.  

0	1	2	3	4	5	6	7	8	9
10	11	12	26	38	17	33	48	35	25

Collision is resolved using the hash function  $H_2(k) = (k \bmod 11) + 1$  (a) Analyze how many key comparisons occur in searching for key 35 in the given hash table? (b) If a new key 67 is inserted into the given hash table, what will be its address?
2. Sort the given values using Quick Sort. 65 70 75 80 85 60 55 50 45
3. The way a card game player arranges his cards as he picks them up one by one, is an example of 1. bubble sort 2. selection sort 3. insertion sort 4. merge sort
4. Given a string of characters (let us say there are about 100 characters or more in the string), what is the most efficient method to use for finding out the character that repeats itself the most?

**Apply the operations of data structures in designing software procedures (CO4):**

1. State the traversals a given tree using Inorder, Preorder and Postorder traversals.
2. Use linked list to design software procedures for a music player.
3. Convert the expression  $((A + B) * C - (D - E) ^ (F + G))$  to equivalent Prefix and Postfix notations.  
 Draw a binary Tree for the expression :  $A * B - (C + D) * (P / Q)$

**Assess the applicability of given data structures and associated operations to real time computer applications (CO5):**

1. If you have one million named objects and you want to store them in a data structure that lets you insert new objects quickly and search for an object by name quickly, what data structure should you use?

- Ram uses a binary search tree for storing values mostly in increasing order. Is it a right data structure to implement? Justify your answer.
- Find the applicability of stack in finding balancing paranthesis in an expression.

### Identify suitable algorithms with appropriate data structures for real time software requirements (CO6)

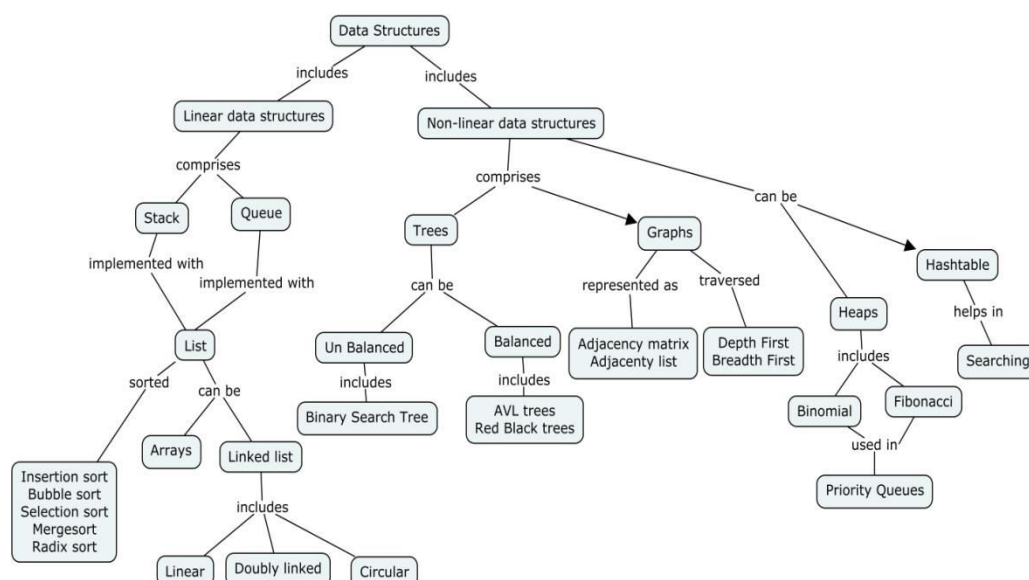
- Given a graph G which connects 8 routers at different locations in a network. The cost of connecting the routers is given as weights in the graph G. Apply Prim's algorithm to find a way of connecting the 8 routers so that the cost of connecting them is low.
- A word processor has unique ids for all its functionalities (eg. Copy- c03, paste – c04 etc) and it wants to maintain the list of activities done on a document for undoing the activities. Choose a data structures for the purpose.
- An online shopping organization likes to store its product details in Binary Search Tree having product id as unique key. Frequent addition, deletion and search of products are mandatory in the process. Design algorithms for the given requirement.

### Modify the existing operations of data structures for changing needs of the software requirements (CO7)

- Consider a game in which players are arranged in a circular linked list. The game has to be played in a way demonstrated as follows.
  - The game starts from first player as current player. Until there is one player
  - Each time a random number (n) from 1 to 10 is generated
  - The count goes from the current player and deletes the n<sup>th</sup> player. After deleting the player, now the game starts from the player next to the deleted player.

The game is to be stopped if there is only one player and the player is declared as a winner. Design an algorithm for the game.
- Merge two linked list of records as a single doubly linked list
- Analyze how a circular queue be changed to store elements in decreasing order

### Concept Map



## Syllabus

**Introduction:** Introduction to Data Structures, Abstract data types

**Linear Data Structures:** Stacks and Queues, Sparse matrices and their representation

**Linked list:** Linear Linked Lists, Circular & Doubly linked list

**Trees:** Binary Search Trees, Red-Black Trees, AVL Trees, B-Trees, Huffman algorithm

**Sorting:** Insertion Sort, Bubble sort, Selection Sort, Mergesort, Quick sort, Counting sort, Radix sort.

**Heaps:** Binomial heaps, Fibonacci Heaps, Priority Queues using heaps

**Searching:** Linear Search, Binary search, Hash tables & hashing techniques.

**Graphs:** Representation, Graph traversals, Minimum Spanning Trees, Shortest Path

**Case study:** Usage of Data structures in any current real time projects.

## Reference Books

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", MIT Press, Third Edition 2009.
2. Tanaenbaum A.S.,Langram Y. Augestein M.J " Data Structures using C" Pearson Education , 2004
3. Mark Allen Weiss,"Data Structures and Algorithm Analysis in C", Pearson Education, 2011.
4. Robert Kruse & Clovis L. Tondo " Data Structures and Program Design in C",Prentice Hall , 2012.
5. Ellis Horowitz et al.," Fundamentals of Data Structures in C", Silicon press, Second edition, 2007.

## Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures
1	<b>Introduction</b>	
1.1	Introduction to Data structures	1
	Abstract data types	1
2	<b>Linear Data Structures</b>	
2.1	Stacks	2
2.2	Queues	2
2.3	Sparse matrices and their implementation	1
3	<b>Linked lists</b>	
3.1	Linear Linked Lists	2
3.2	Doubly Linked Lists	2
3.3	Circular Linked Lists	1
4	<b>Trees</b>	
4.1	Binary Search Trees	2
4.2	Red-Black Trees	2

4.3	AVL Trees	2
4.4	B-Trees	2
4.5	Huffman algorithm	1
5	<b>Heaps</b>	
5.1	Binomial Heap	1
5.2	Fibonacci Heap	1
5.3	Priority Queues using heaps	1
6	<b>Data Search</b>	
6.1	Linear Search and Binary search	1
6.2	Hash Table	1
6.3	Hashing techniques (Open, Closed, Collision)	1
7	<b>Data Sorting</b>	
7.1	Insertion Sort , Bubble sort	1
7.2	Selection Sort	1
7.3	Quick sorting	1
7.4	Merge sorting	1
7.5	Counting sort, Radix sort	2
8	<b>Graphs</b>	
8.1	Representation	1
8.2	Traversals	2
8.3	Minimal Spanning Trees	2
8.4	Shortest Path algorithm	2
	Case study	1
	Total	41

**Course Designer:**

Dr. D. Anitha

[anithad@tce.edu](mailto:anithad@tce.edu)

<b>20CA150</b>	<b>DATABASE MANAGEMENT SYSTEMS</b>	Category	L	T	P	Credit
		PC	4	0	0	4

**Preamble**

This course aims at facilitating the student to understand the various functionalities of DBMS software and perform many operations related to creating, manipulating and maintaining databases for Real-world applications and student to understand the various designing concepts, storage methods, querying and managing databases.

**Prerequisite**

None

**Course Outcomes**

On the successful completion of the course, students will be able to

CO1: Explain the structure and model of the relational database system	Understand
CO2: Design multiple tables, and using group functions, sub queries	Apply
CO3: Design a database based on a data model considering the normalization to a specified level	Analyze
CO4: Estimate the storage size of the database and design appropriate storage techniques	Apply
CO5: Analyze the requirements of transaction processing, concurrency control	Analyze
CO6: Explain the basic requirements for Backup and recovery	Understand

**Mapping with Programme Outcomes**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	L	M	M	M	L	-	-	-	-	-	-	-
CO2	M	M	S	M	L	-	-	-	-	-	-	-
CO3	S	S	S	-	-	-	-	-	-	-	-	-
CO4	S	S	L	M	S	-	-	-	-	-	-	-
CO5	M	S	S	M	-	-	-	-	-	-	-	-
CO6	L	M	M	M	L							

S- Strong; M-Medium; L-Low

### Assessment Pattern

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	30	30	20	20
Understand	30	20	20	20
Apply	30	30	20	20
Analyse	10	20	40	40
Evaluate	0	0	0	0
Create	0	0	0	0

### Course Level Assessment Questions

#### Explain the structure and model of the relational database system (CO1):

1. What is database management system?
2. Define schema?
3. Explain the structure of a DBMS.
4. Mention the types of constraint with an example for each type.
5. What is the use of having clause?
6. Mention the use of Commit and Rollback commands.
7. Define strong entity set with an example.
8. Define the term ACID properties.
9. How the database system is advantageous than file system?

#### Design multiple tables, and using group functions, sub queries (CO2):

1. For the following employee database  
employee(employee-name, street, city)  
works(employee-name, company-name, salary)  
company(company-name, city)  
manages(employee-name, manager-name)  
Construct the appropriate tables along with the required constraints.
2. If you want to remove the primary constraints that you have created for the employee table, how will you do that?
3. If you want to change the city name of the company "TCS", what will you do?
4. Find those companies whose employees earn a higher salary, on average than the average salary at TCS.
5. Delete all the employees in TCS who earn less than Rs.10,000 .
6. If you want to get back the employees in TCS who earn less than Rs.10,000, what will you do?
7. You allow the user "A" to access the information regarding the employee's address along with the company in which they are working.
8. Consider a relational database with two relations  
Course(course\_name,room,instructor)  
Entrollment(course\_name, student\_name, grade)  
Create the instances of these relations for three courses, each of which enrolls five students.
9. Consider the following account relation and construct a bitmap index on the attributes

branch\_name and balance, dividing balance values into 4 ranges – < 250, 250 ..<500, 500..<750 and >750.

Account_No	Branch_Name	Balance
A-217	Madurai	200
A-219	Chennai	600
A-117	Coimbatore	350
A-207	Madurai	800
A-317	Chennai	700

**Design a database based on a data model considering the normalization to a specified level (CO3):**

- Using the functional dependencies given  $A \rightarrow BC$ ,  $CD \rightarrow E$ ,  $B \rightarrow D$ ,  $E \rightarrow A$  Compute  $B^+$ .
- Let relations  $r_1(A,B,C)$  and  $r_2(C,D,E)$  have the following properties:  $r_1$  has 20,000 tuples,  $r_2$  has 45,000 tuples, 25 tuples of  $r_1$  fit on one block, and 30 tuples of  $r_2$  fit on one block. Estimate the number of block transfers and seeks required using Hash join strategy for  $r_1$  natural joined with  $r_2$ .
- List out all the functional dependencies satisfied by the relation. Explain how they are satisfied.

A	B	C
a1	b1	c2
a1	b1	c2
a2	b4	c1
a3	b2	c3

Design a database for the Banking environment by following the various design phases including normalization.

**Estimate the storage size of the database and design appropriate storage techniques (CO4):**

- How does the remapping of bad sectors by disk controllers affect data retrieval rates?
- Draw the various levels of Redundant Arrays of Independent Disks and explain.
- Illustrate the Physical characteristics of Magnetic Disks?
- Illustrate Multitable Clustering File organization with suitable example

**Analyze the requirements of transaction processing, concurrency control(CO5):**

- Consider the following two transactions:

T1: read(A); read(B); if A=0 then B:=B+1; write(B)	T2: read(B); read(A); if B=0 then A:=A+1; write(A)
---	---

Let the consistency requirement be  $A=0$  or  $B=0$ , with  $A=B=0$  the initial values. Show that every serial execution involving these two transactions preserve the consistency of the database.



### Explain the basic requirements for Backup and recovery (CO6)

1. Consider the following transactions

T1: read(A); read(B); if A=0 then B:=B+1; write(B)	T2: read(B); read(A); if B=0 then A:=A+1; write(A)
---	---

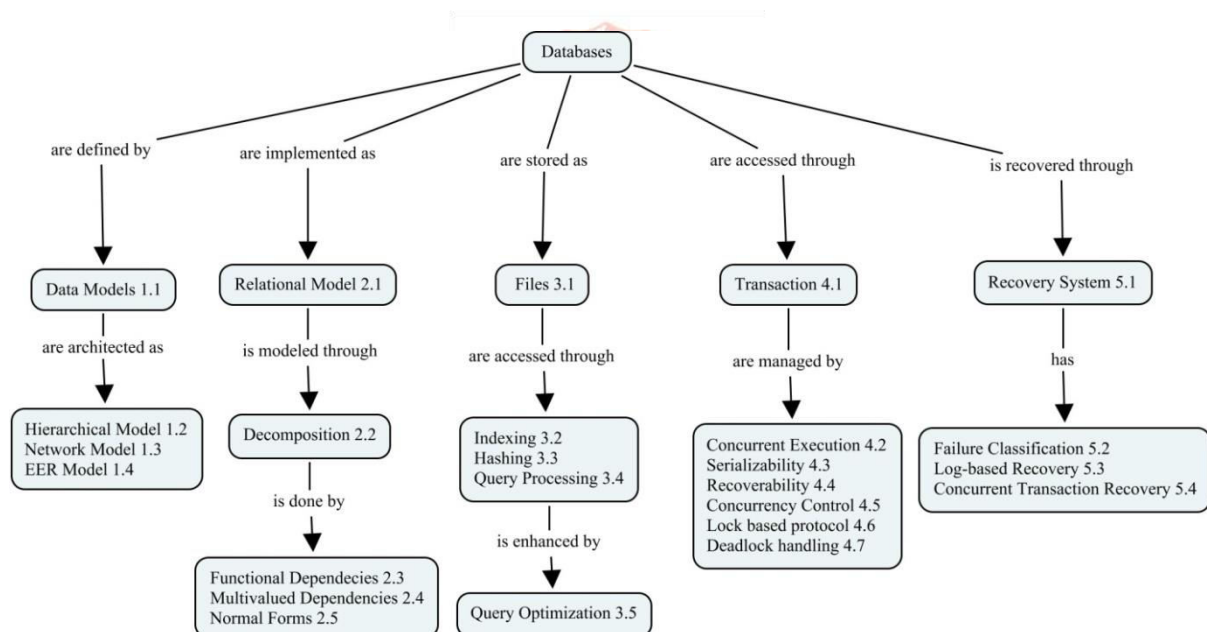
Add lock and unlock instructions to the above transactions and show that they observe two-phase locking protocol.

2. Compute the closure of the following set F of functional dependencies for relation schema  $R=(A,B,C,D,E)$ .

$A \rightarrow BC$ ,  $CD \rightarrow E$ ,  $B \rightarrow D$ ,  $E \rightarrow A$

List the candidate keys for R.

### Concept Map



### Syllabus

**Introduction** - Concepts, Relationships, System Architecture, Data Models, Relational languages **Data Definition Language:** Tables creation, Constraints, Displaying table information, Altering and Renaming an existing table, Truncating and dropping the table. **Data management and Retrieval** - DML operations – Insert, Delete & Update, Basic queries – select, Arithmetic operations, where clause, Sorting, CASE structure, DEFINE command, Complex queries – Functions and Grouping. **Database Design** – Relational Model, Decomposition, Functional Dependencies, Multivalued Dependencies, Normal forms **Storage Structure** – Storage and File Structure, Indexing and Hashing, **Query Processing** – Measures of query cost, Evaluation of Expressions, Query Optimization – Estimation statistics of Expression Results, Evaluation Plans, **Transaction and Concurrency control** - Transaction concepts, Concurrent Execution, Serializability, Recoverability, Concurrency Control, Lock based protocol, Deadlock handling.

### Reference Books

1. Henry F. Korth, Abraham Silberchatz, S.Sudarshan , Database System Concepts, McGraw-Hill – 2010
2. Ramez Elmasri, Shamkant B. Navathe - Fundamentals of Database Systems –Fifth Edition - Addison Wesley Higher Education – 2010
3. Raghu Ramakrishnan, Johannes Gehrke - Database Management Systems - Third Edition - McGraw-Hill - 2014
4. C.J.Date, Longman, Dr.S.Swamynathan, Introduction to Database Systems, Pearson Education - 2010
5. Hoffer, Prescott & McFadden - Modern Database Management – Eighth Edition - Prentice Hall – 2010
6. Kifer, Bernstein & Lewis - Database Systems: An Application Oriented Approach, Complete Version – Second Edition - Addison Wesley Higher Education - 2010

### Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures
<b>1</b>	<b>Introduction</b>	
1.1	Concepts	1
1.2	Relationships	1
1.3	System Architecture	1
1.4	Data Models	2
1.5	Relational languages	1
<b>2.</b>	<b>Data Definition Language</b>	
2.1	Tables creation	1
2.2	Constraints	1
2.3	Displaying table information	1
2.4	Altering and Renaming an existing table	1
2.5	Truncating and dropping the table	1
<b>3.</b>	<b>Data management and Retrieval</b>	
3.1	DML operations – Insert , Delete & Update	1
3.2	Basic queries – select	1
3.2.1	Arithmetic operations , where clause , Sorting	1
3.2.2	CASE structure , DEFINE command	1
3.3	Complex queries – Functions and Grouping	1
3.3.1	Multiple tables : JOINS and SET operators	1
3.3.2	Sub queries : Nested queries	1
<b>4.</b>	<b>Database Design</b>	
4.1	Relational model	1
4.2	Decomposition	1
4.3	Functional dependencies	2

Module No.	Topic	No. of Lectures
4.4	Multi valued dependencies	2
4.5	Normal Forms	2
<b>5.</b>	<b>Data Storage and Querying</b>	
5.1	Storage and File Structure	2
5.2	Indexing and Hashing	2
5.3	<b>Query Processing</b> – Measures of query cost	2
5.4	Evaluation of Expressions	1
5.5	Query Optimization – Estimation statistics of Expression Results	2
5.6	Evaluation Plans	1
<b>6</b>	<b>Transaction and Concurrency control</b>	
6.1	Transactions - concepts	1
6.2	Concurrent Execution	1
6.3	Serializability	1
6.4	Recoverability	1
6.5	Concurrency Control	1
6.6	Lock based protocol	1
6.7	Deadlock handling	1
	Total	42

**Course Designers:**

Dr. S. Vijayalakshmi

svlcse@tce.edu

<b>20CA170</b>	<b>DATA STRUCTURES USING C PROGRAMMING LABORATORY</b>	Category	L	T	P	Credit
		PC	0	0	2	2

### Preamble

This Laboratory course will enable students to identify, formulate and solve real world engineering problems that require usage of algorithms in C. The course serves as a foundation laboratory for improving the problem solving skills of students.

### Prerequisite

None

### Course Outcomes

On the successful completion of the course, students will be able to

CO1: Design algorithms for the given problem specifications.	Analyze
CO2: Write C programs for the designed algorithm specification.	Apply
CO3: Write C programs to implement linear data structures : Stack and Queue using arrays and linked list in an application context	Apply
CO4: Implement Non linear data structures: Graph, Trees, Hashtable in an application context	Apply
CO5: Implement specific sort algorithms in application context.	Apply
CO6: Generate different test cases for testing the validity of the developed programs	Analyze
CO7: Write technical report on the observations from the experiments	Apply

### Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	S	L	L	M	M	-	-	-	-	-
CO2	S	S	S	L	M	M	M	-	-	-	-	-
CO3	S	S	S	L	M	M	M	-	-	-	-	-
CO4	S	S	S	L	M	M	M	-	-	-	-	-
CO5	S	S	S	L	M	M	M	-	-	-	-	-
CO6	S	S	S	M	L	M	M	-	-	-	-	-
CO7	S	M	M	L	L	M	L	-	M	-	-	-

S- Strong; M-Medium; L-Low

### Lab experiments

Develop C programs for

1. Conditional and Iterative Structures
2. Arrays, Functions and Strings
3. Structures and Unions
4. Pointers
5. File Handling
6. Stack ADT implementation – Array implementation
7. Queue ADT implementation – Linked list implementation
8. Binary Search tree implementation
9. Hash table implementation
10. Graph representation and traversals
11. Sorting Algorithms: A) Sorting algorithm of  $O(n^2)$   
B) Sorting algorithm of  $O(n \log n)$

#### Mini Project:

The course instructor shall provide real time problems/specifications to the students for mini project. The project shall be completed before the commencement of 2<sup>nd</sup> semester and a report shall be submitted.

#### Sample specification for mini project:

- Develop a C processor that performs stemming on the words in a document.
- Simulate a client server process in which the server process clients on a priority basis.

#### Note:

For Exercise 1 to 5, a collection of program specifications shall be designed by the course instructor and assigned to the students.

For Exercise 6 to 11, students shall develop C programs in any one application domain (e.g. Banking, Online shopping, Conference management etc.)

### Reference Books

1. Brian W Kernighan & Dennis Ritchie, "The C programming language", 2nd Edition, Prentice Hall ,2015
2. Yashavant Kanetkar," Let us C", BPB Publications 8th Edition, 2014
3. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Printice hall International, 2010.
4. Mark Allen Weiss,"Data Structures and Algorithm Analysis in C", Pearson Education, 2011.
5. Robert Kruse & Clovis L. Tondo " Data Structures and Program Design in C",Prentice Hall , 2012.

### Course Designers

Dr. D. Anitha

anithad@tce.edu

<b>20CA180</b>	<b>RDBMS LABORATORY</b>	Category	L	T	P	Credit
		PC	0	1	2	3

**Preamble**

This course aims at giving adequate exposure to students on the Database design and E-R modelling. The course also facilitates students with hands on training on SQL and programming language extension to SQL within the RDBMS environment.

**Prerequisite**

None

**Course Outcomes**

On the successful completion of the course, students will be able to

CO1: Model Entity Relationship with E-R diagrams	Apply
CO2: Design database schema considering normalization and relationships within database	Apply
CO3: Write SQL queries to user specifications	Apply
CO4: Develop triggers, procedures, user defined functions and design accurate and PLSQL programs in Oracle and DB2.	Apply
CO5: Use the database from a front end application	Apply
CO6: Prepare technical report on the observations of the experiments	Apply

**Mapping with Programme Outcomes**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	S	-	S	L	L	-	-	-	-	-
CO2	S	S	S	L	S	L	L	-	-	-	-	-
CO3	S	S	S	-	S	L	L	-	-	-	-	-
CO4	S	S	S	-	S	L	L	-	-	-	-	-
CO5	S	S	S	-	S	L	L	-	-	-	-	-
CO6	L	-	-	-	M	M	L	-	S	-	-	-

S- Strong; M-Medium; L-Low

## List of Experiments

1. Basic SQL – DDL & DML, Views, Group operations, aggregate operations, System operations in Oracle
2. Intermediate SQL –Joins, Subqueries, DCL operations
3. Advanced SQL – Nested tables, V-arrays
4. ER Modeling
5. Database Design and Normalization
6. Stored procedures and using them in a client application
7. Triggers and their front end application
8. DBA mechanisms – Installation, Backup and recovery operations, Batch processing
9. **Mini Project**  
The course instructor shall provide real time problems / specifications to the students for mini project. The project shall be completed before the commencement of 2<sup>nd</sup> semester and a report shall be submitted.

### Sample Specifications

**Design a database for student mark entry system. Provide provisions for different queries, procedures and triggers.**

#### Sample Queries:

1. List the name of students under one particular staff.
2. Find the students who are below 50 marks in all subjects.

#### Procedure:

Write the procedure for calculating total marks (internal+ external) for a particular student.

#### Trigger:

Raise the trigger for entering the internal mark more than 30.

**Marks:** Algorithm = 10 E-R Diagram = 15. Table Design=15 Queries = 20.

Procedure = 15. Trigger = 15. Viva =10.

**Note: Experiments 1 to 8 are to be carried out in a single application domain preferably in Oracle/DB2.**

### Course Designer:

Dr. S. Vijayalakshmi

[svlcse@tce.edu](mailto:svlcse@tce.edu)

Dr. D. Anitha

[anithad@tce.edu](mailto:anithad@tce.edu)

**20CA210 SOFTWARE ENGINEERING**

Category	L	T	P	Credit
PC	4	0	0	4

**Preamble**

This course aims at introducing to the students about the product that is to be engineered and the process that provides a framework for the engineering technology. The course facilitates the students to analyze risk in software design and quality and to plan, design, develop and validate the software project.

**Prerequisite**

None

**Course Outcomes**

On the successful completion of the course, students will be able to

- |  |                   |
|--|-------------------|
| CO1: Explain a process model for a software project Development.   | <b>Understand</b> |
| CO2: Prepare the SRS, Design document, Project plan of a given software system                                   | <b>Analyze</b>    |
| CO3: Apply Project Management and Requirement analysis, Principles to S/W project development.                   | <b>Apply</b>      |
| CO4: Analyze the cost estimate and problem complexity using various estimation techniques                        | <b>Analyze</b>    |
| CO5: Generate test cases using the techniques involved in selecting: (a) White Box testing (b) Block Box testing | <b>Analyze</b>    |
| CO6: Explain the advantages of configuration management and risk management activities.                          | <b>Understand</b> |

**Mapping with Programme Outcomes**

S-Strong; M-Medium; L-Low

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	L	-	-	-	-	L	-				
CO2	S	M	-	-	-	-	-	-				
CO3	M		M	M		-	S	S				
CO4	L		L		S	-		L				
CO5	M	S		S	L	-						
CO6		S		M	L	-	M	S				



### Assessment Pattern

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	20	20	20	20
Understand	40	20	20	30
Apply	20	40	40	30
Analyse	20	20	20	20
Evaluate	0	0	0	0
Create	0	0	0	0

### Course Level Assessment Questions

#### Explain a process model for a software project Development. (CO1):

1. Define: Software Engineering.
2. Define: Metric, Measure and Indicator
3. Who are called as Stakeholders?
4. How learning is performed in ASD process model?
5. How extreme programming helps in modern software product development scenario? Explain the various activities involved in it with neat diagram
6. How requirements elicitation process is carried out during requirements analysis using QFD? Explain it in detail.
7. Consider a scenario of a customer's organization in need of a faster product development and the requirements are well understood. Apply RAD model helps in developing software product in this scenario? Explain it with a neat diagram.
8. Consider a system in which the developer has to add a new functionality based on changing customers' requirements. This has to be tested before and after by the tester. How Extreme Programming works in this situation? Explain it in detail.
9. Draw a Software process framework.
10. Quote an example of software projects that would be amenable to the waterfall model.

#### Prepare the SRS, Design document, Project plan of a given software system (CO2):

1. Consider an 'Online Vehicle Purchase System'. Apply requirements analysis and design to draw primary use case diagram, swimlane activity diagram and State Diagram for it.
2. Apply flow-oriented modelling to develop a DFD with level 0 and 1 and state diagram for a University Admission System.
3. Apply Control Flow based testing in White box testing to generate the test cases and independent paths in a sample piece of code.
4. Prepare a data dictionary by creating entries for all the data objects for an 'Online Vehicle Purchase System'
5. Construct a Structure chart from a DFD for your own application.
6. For a 'Students Attendance Monitoring System', create a primary use case diagram.
7. Apply the basic guidelines for conducting a collaborative requirement gathering meeting and prepare a SRS document for Online Shopping cart management system

**Apply Project Management and Requirement analysis, Principles to S/W project development. (CO3):**

1. Define QFD and show its different types of requirements.
2. Show the Seven distinct functions of Requirement Engineering Task.
3. Develop a complete use case for the following activities.
  - i. Making a withdrawal at an ATM,
  - ii. Searching for the books (on a specific topic) using an online book store.
4. Consider a system that has aspects A01, A02, A03 and A04. The system has been analyzed and it has been identified that there are requirements R01, R02, R03, R04 and R05. During the analysis it has been identified that, Req. R01 covered A01 and A03; R02 covered A04; R03 covered A01, A02 and A04; R04 covered A03 and A04; R05 covered A02 and A03. Draw a Requirements Traceability Table for the above and identify which requirements are sufficient to construct the system.
5. Discuss some of the problems that occur when requirements must be elicited from three of four different customers.

**Analyze the cost estimate and problem complexity using various estimation techniques (CO4):**

1. Analyze the effort calculated using COCOMO II model for semi detached and organic models for the same KLOC and provide your conclusion.
2. Evaluate the Cost estimate for a system having 10KLOC lines of code and is of 'Semi Detached Type' with average complexity metrics of all the attributes, using COCOMO II model.
3. Use the COCOMO-II model to estimate the effort required to build software for a simple E-Shopping application that provides 18 screens (simple), 15 reports (medium) and will require approximately 60 software components (difficult). Assume the developer's experience/capability is high and environment maturity/capability is very high. Use the application composition model with object points.
4. Evaluate the Complexity measure using FP based estimation for a system in which the following data exists:
  - i. No.of User Inputs – 20
  - ii. No.of User Outputs – 12
  - iii. No.of Enquiries - 8
  - iv. No.of Internal Logic Files – 4
  - v. No.of External Interfaces – 5
- b. Assume your own complexity level for each of the categories and you're your own values for the 14 questions raised to the customers.
5. Evaluate the developers' efficiency in Delphi Cost estimation model.
6. Using the risk projection table evaluate the highest priority risks in a given software.

**Generate test cases using the techniques involved in selecting: (a) White Box testing (b) Block Box testing (CO5):**

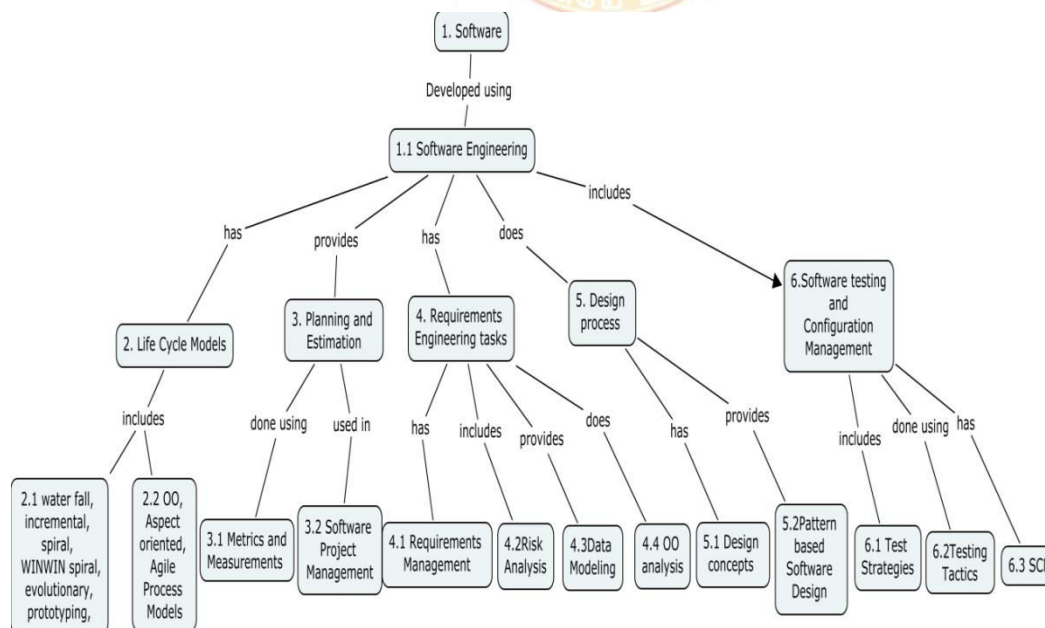
1. What is White Box testing? Draw the notations of a CFG and derive independent paths and test cases for a given source code using it.
2. Explain the following testing techniques to derive test cases (a) Equivalence Partitioning (b) Boundary Value Analysis (c) Random testing for OO classes.

3. Apply Control Flow based testing in White box testing to generate the test cases and independent paths in a sample piece of code.
4. Suppose a test group was testing a mission critical software system. The group has found 85 out of the 100 seeded defects. If you were the test manager, would you stop testing at this point?
5. Consider the “Withdraw” module in an ATM application. The module reads the amount the user wishes to withdraw from his/her account. The amount must be multiple of 100 and less than 10000. Develop black box test cases using Equivalence class partitioning and boundary value analysis. List out any assumptions that you make in deriving these test cases.

### Explain the advantages of configuration management and risk management activities (CO6):

1. Explain the role, contents and features of SCM repository in detail.
2. How risk mitigation, monitoring and management is done in software development process with RMMM plan? Explain.
3. How version control and change control are done in SCM process.
4. Using the risk projection table evaluate the highest priority risks in a given software.
5. Construct a sample risk projection table and assess the risk impact for any one risk.
6. Create a RMMM plan for any two risks identified during risk analysis.

### Concept Map



### Syllabus

**Software Process and Life Cycle Models:** Introduction to Software Engineering, Software Process, Life cycle models: water fall, incremental, spiral, WINWIN spiral, evolutionary, prototyping, object oriented, Aspect oriented, Agile Process Models. **Requirements Engineering tasks:** Software Requirements: Functional and Non-Functional, User requirements, System requirements, Software Requirements Document. Requirement

Engineering Process: Feasibility Studies, Requirements elicitation and analysis, requirements validation, requirements management, Data Modelling, OO analysis **Software Project Management:** Software Project Management: Estimation, LOC and FP Based Estimation, COCOMO Model, Project Scheduling: Scheduling, Earned Value Analysis - Risk Management. **Software Design:** Design process: Design concepts, Data design elements: Pattern based Software Design. **Software Testing:** Software testing strategies: fundamentals, Internal and external views of Testing-white box testing, basis path testing, control structure testing, black box testing, Regression Testing, Unit Testing, Integration Testing, Validation Testing, System Testing And Debugging.

### Reference Books

1. Roger Pressman, Software Engineering: A Practitioners Approach, (8th Edition), McGraw Hill, 2015
2. Eric J. Braude and Micheal E. Bernstein, Software Engineering Modern Approach, second edition, Wiley, 2011.
3. Ian Somerville, Software Engineering, 9th edition, Addison Wesley, 2011

### Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
<b>1</b>	<b>Software</b>	
1.1	Software Engineering	3
<b>2</b>	<b>Life Cycle Models</b>	
2.1	Water fall, incremental, spiral, WINWIN spiral, evolutionary, prototyping	4
2.2	Object oriented, Aspect oriented and Agile Process Models	3
<b>3</b>	<b>Planning and Estimation</b>	
3.1	Metrics in Software Project Management	3
<b>4</b>	<b>Requirements Engineering tasks</b>	
4.1	Requirements Management	3
4.2	Risk Management	3
4.3	Data Modeling	2
4.4	OO analysis	3
<b>5</b>	<b>Design process</b>	
5.1	Design concepts	3
5.2	Data design elements	3
5.3	Pattern based Software Design	3

Module No.	Topic	No. of Periods
	Tutorial	2
<b>6</b>	<b>Software testing and Configuration Management</b>	
6.1	Test Strategies for conventional and OO software	2
6.2	Testing Tactics	2
6.3	Software Configuration Management	1
	Total	40

**Course Designer:**

Dr.T.Chandrakumar

tckcse@tce.edu



<b>20CA220</b>	<b>OBJECT ORIENTED PROGRAMMING USING C++</b>	Category	L	T	P	Credit
		PC	3	1	0	4

### Preamble

The course facilitates the students to master all techniques of software development in C++ Programming Language and to demonstrate these techniques by implementing the solution for variety of problems spanning the breadth of the language.

### Prerequisite

- 20CA120 : Programming in C

### Course Outcomes

On the successful completion of the course, students will be able to

CO1: Distinguish between Structured and Object Oriented problem solving approaches and apply them based on the problem given	Understand
CO2: Identify classes and objects from the given problem description and able to create classes and objects using C++	Understand
CO3: Improve secured data processing by applying Abstraction, Encapsulation and Information hiding	Apply
CO4: Achieve code reusability and extensibility by means of Inheritance and Polymorphism.	Apply
CO5: Handle Exceptions and identify Run Time Type Information (RTTI)	Apply
CO6: Translate the informal description of an algorithm to solutions for problems in engineering, science and text processing using Object Oriented Programming.	Analyze

### Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	P12
CO1	S	S	S	L	L	L	M	-	-	-	-	-
CO2	S	S	S	L	L	L	M	-	-	-	-	-
CO3	S	S	S	L	M	M	S	-	-	-	-	-
CO4	S	S	S	L	M	M	S	-	-	-	-	-
CO5	S	S	S	L	M	M	S	-	-	-	-	-
CO6	S	S	S	M	L	M	S	S	-	-	-	-

S- Strong; M-Medium; L-Low

### Assessment Pattern

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	20	20	20	10
Understand	20	20	20	20
Apply	40	40	40	50
Analyse	20	20	20	20
Evaluate	0	0	0	0
Create	0	0	0	0

### Course Level Assessment Questions

#### Distinguish between Structured and Object Oriented problem solving approaches and apply them based on the problem given (CO1):

1. Suppose we have similar structured classes but with varying data types, how will you apply class templates to achieve the same.
2. Suppose that you have applied OO principle for your application. Analyze the drawbacks when compared to structured approach.
3. Differentiate between procedural paradigm and OO paradigm.
4. How top down approach is applied in conventional programming?

#### Identify classes and objects from the given problem description and able to create classes and objects using C++ (CO2):

1. Define: Class and Object.
2. What is a constructor? What are its types?
3. Write the purpose of a destructor?
4. Differentiate between procedural paradigm and OO paradigm.
5. What is the need for an explicit constructor in C++?
6. How namespaces reduces complexity? Explain.

#### Improve secured data processing by applying Abstraction, Encapsulation and Information hiding (CO3):

1. Differentiate static and dynamic binding?
2. Apply namespaces to achieve modularization in developing a complex application.
3. What is encapsulation?
4. Write a C++ code to do stack operations using interface and implementation separation.
5. What is information hiding?

#### Achieve code reusability and extensibility by means of Inheritance and Polymorphism. (CO4):

1. Consider a Client-Server application development – Banking Transaction System. Apply multiple inheritance to achieve the transaction and update the file for the same.
2. Consider a Stock Maintenance System. Do coding for Purchase and Sales and stock updating by applying virtual.
3. How C++ resolves the same method with different implementations in inheritance hierarchy?
4. Develop a program in C++ for calculating areas of different shapes by applying method overriding.
5. Apply method overloading to calculate incentives for different types of employees in an organization.

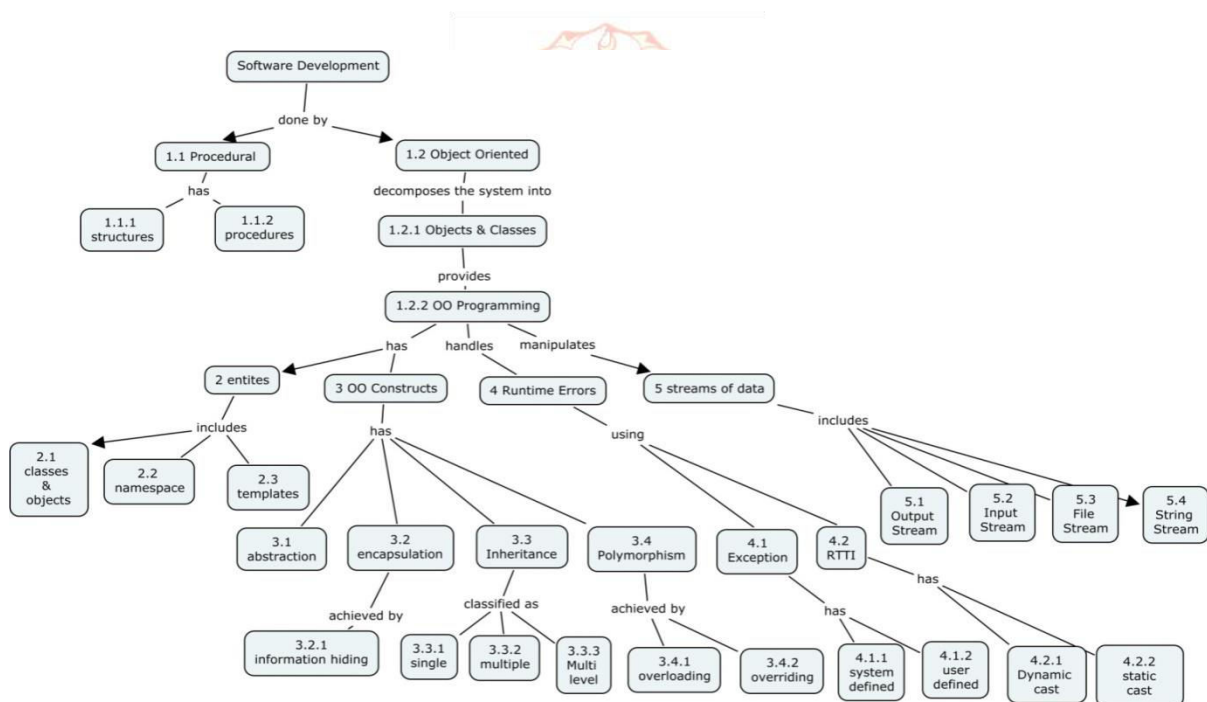
### Handle errors by means of Exception Handling and Run Time Type Identification (RTTI) (CO5)

1. Differentiate Error and an exception?
2. Consider an Employee payroll system. Apply multiple inheritance to calculate the payroll and also apply RTTI to find out the current active object in the calculation.
3. Analyze the usage of RTTI?
4. Suppose that, a method doesn't have the ability to handle an exception, which may be the suitable way of handling it? Give an example.

### Translate the informal description of an algorithm to solutions for problems in engineering, science and text processing using Object Oriented Programming (CO6)

1. Apply file stream classes to manage an Address Book.
2. What are File streams? Write a C++ program to create a file called "Item.txt" to have records being stored as objects of items.
3. What are string streams? Write a program of your own to manage streams of string data.

### Concept Map



### Syllabus

**Introduction to Programming Paradigms** : Procedural Programming Vs. Object-Oriented Programming – **Entities** : Namespaces – Access Specifiers - Classes and Objects – Constructors - Destructors - Class Objects as Members - **OO Constructs** : **Abstraction** – Function Templates - Class Templates - **Encapsulation** – Member data and Member function binding – Information hiding - **Inheritance** – Base Class and Derived Class – Types :Single, Multiple, Multi-level, Hybrid- Access Specifiers and Inheritance - **Polymorphism** - Overloading : Operator Overloading – Method Overloading - Templates Overloading: - Overriding : Method Overriding - Virtual Base Classes – Virtual Functions - Virtual Constructors - Friends - Abstract Classes - **Streams**: Output Streams - Input Streams- File Streams - String Streams – **Exception Handling**: System Defined Exceptions– User Defined Exceptions - try-catch-throw-rethrow **Run-Time Type Information (RTTI)** : Dynamic Cast - Typeid - Uses and Misuses of RTTI



## Reference Books

1. Bjarne Stroustrup, "The C++ Programming Language", 4<sup>th</sup> Edition, Addison-Wesley, 2015.
2. Scott Meyers, "Effective C++ 55 Specific Ways to Improve Your Programs and Designs", Third Edition, Addison-Wesley, 2011
3. Paul Deital & Harvey Deital, "C++ How to Program", 7<sup>th</sup> Edition, Pearson Education, 2010
4. Stanley Lippman, "C++ Primer", 4th Edition, Pearson Education, 2007.
5. Yashavant P. Kanetkar, "Let Us C++", BPB Publications, 2007.
6. Robert Laffore, "Object Oriented Programming using C++", 4<sup>th</sup> Edition, Sams Publishing, 2002.
7. Ashok Kamthane, "Object Oriented Programming with ANSI and Turbo C++", 4<sup>th</sup> Edition, Pearson Education, 2002.

## Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures
1	<b>Introduction to Programming Paradigms</b>	
1.1	Procedural Programming Vs. Object-Oriented Programming	2
2	<b>Entities</b>	
2.1	Namespaces	2
2.2	Classes and Objects	1
2.3	Constructors	1
2.4	Destructors	1
2.5	Class Objects as Members	1
3.1	<b>Abstraction</b>	
3.1.1	Function Templates	1
3.1.2	Class Templates	2
3.2	<b>Encapsulation</b>	
3.2.1	Member data and Member function binding	1
3.2.2	Information hiding	1
3.3	<b>Inheritance</b>	
3.3.1	Base Class and Derived Class	2
3.3.2	Single, Multiple, Multi-level, Hybrid	2
3.3.3	Access Specifiers and Inheritance	1
3.4	<b>Polymorphism</b>	
3.4.1	Overloading	1
3.4.2	Operator Overloading	2
3.4.3	Method Overloading	1
3.4.4	Templates Overloading	1
3.4.5	Overriding	1
3.4.6	Method Overriding	1
3.4.7	Virtual Base Classes	1
3.4.8	Virtual Functions	1
3.4.9	Virtual Constructors	1
3.4.10	Friends	1
3.4.11	Abstract Classes	1
4	<b>Streams</b>	
4.1	Output Streams	1

4.2	Input Streams	1
4.3	File Streams	1
4.4	String Streams	1
5	<b>Exception Handling</b>	
5.1	System Defined Exceptions	1
5.2	User Defined Exceptions	1
5.3	catch-throw-rethrow	1
6	<b>Runtime Type Information</b>	
6.1	Dynamic Cast	1
6.2	Typeid	1
6.3	Uses and Misuses of RTTI	1
	Total	40

**Course Designer:**

Prof. N. Murali

nmcse@tce.edu



<b>20CA230</b>	<b>DESIGN AND ANALYSIS OF ALGORITHMS</b>	Category	L	T	P	Credit
		PC	3	1	0	4

### Preamble

This course will enable students to solve a given problem using an algorithm. Also, it enables to mathematically analyze the algorithms for its efficiency and effectiveness.

### Prerequisite

- 20CA140: Data Structures

### Course Outcomes

On the successful completion of the course, students will be able to:

CO1: Summarize the relevance of algorithms for computational problems solving and real time applications.	Understand
CO2: Differentiate different algorithmic approaches, techniques and methods.	Understand
CO3: Apply design and analysis techniques for a given algorithm.	Apply
CO4: Apply optimization techniques for improving the efficiency of algorithms.	Apply
CO5: Analyze a given algorithm for its efficiency based on time and space it occupies.	Analyze
CO6: Evaluate any given problem with mathematical rigor to provide an algorithmic based solution.	Evaluate

### Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	S	S	L	-	M	-	-	-	-	-
CO2	S	M	M	S	S	-	L	-	-	-	-	-
CO3	S	S	M	M	M	-	L	-	-	-	-	-
CO4	M	S	M	S	S	-	M	-	-	-	-	-
CO5	S	L	S	S	M	-	L	-	-	-	-	-
CO6	S	M	S	S	M	-	M	-	-	-	-	L

S- Strong; M-Medium; L-Low

### Assessment Pattern

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	15	10	10	10
Understand	40	10	10	20
Apply	25	30	30	20
Analyse	20	30	30	30
Evaluate	0	20	20	20
Create	0	0	0	0

### Course Level Assessment Questions

#### Understand the relevance of algorithms for computational problems solving and real time applications (CO1):

1. What kind of problems is solved by algorithms?
2. Give an example of an application that requires algorithmic content at the application level, and discuss the function of the algorithms involved.
3. Is it possible to compare the running times of algorithms?
4. List a few applications of algorithms.
5. How will you connect algorithms for problem solving?

#### Differentiate different algorithmic approaches, techniques and methods (CO2):

1. Name a few algorithmic approaches/techniques for problem solving.
2. Discuss the procedure used in Dynamic programming to solve problems.
3. Mention the steps in Greedy Method to arrive at optimal solution.
4. Suppose that instead of always selecting the first activity to finish, we instead select the last activity to start that is compatible with all previously selected activities. Describe how this approach is a greedy algorithm, and prove that it yields an optimal solution.
5. Prove that the fractional knapsack problem has the greedy-choice property.

#### Apply design and analysis techniques for a given algorithm (CO3):

1. Describe the steps in solving knapsack problem using Greedy method and Dynamic programming.
2. How will you apply Divide and Conquer method to arrive at an optimal solution for a given problem?
3. Consider the problem of making change for  $n$  rupees using the fewest number of coins. Assume that each coin's value is an integer. Describe a greedy algorithm to make change consisting of denominations 5, 10, 50 and 100.
4. Analyze the greedy strategy to the traveling salesman problem yields the following algorithm: "At each stage visit the unvisited city nearest to the current city".
5. What is the method used to find the solution in  $n$ -queen problem by symmetry?

#### Apply optimization techniques for improving the efficiency of algorithms (CO4):

1. Describe the potential advantage of Notation  $\Omega$ ,  $\theta$ ,  $O$ .
2. Establish the worst-case running time of heap sort is  $\Omega(n \log n)$ .
3. Suppose we perform a sequence of  $n$  operations on a data structure in which the  $i^{\text{th}}$  operation costs  $i$  if  $i$  is an exact power of 2, and 1 otherwise. Use aggregate analysis to determine the amortized cost per operation.
4. How can we modify almost any algorithm to have a good best-case running time?
5. Determine the optimal parenthesization of a matrix-chain produce whose sequence of dimension is  $\{5, 10, 3, 12, 5, 50, 6\}$

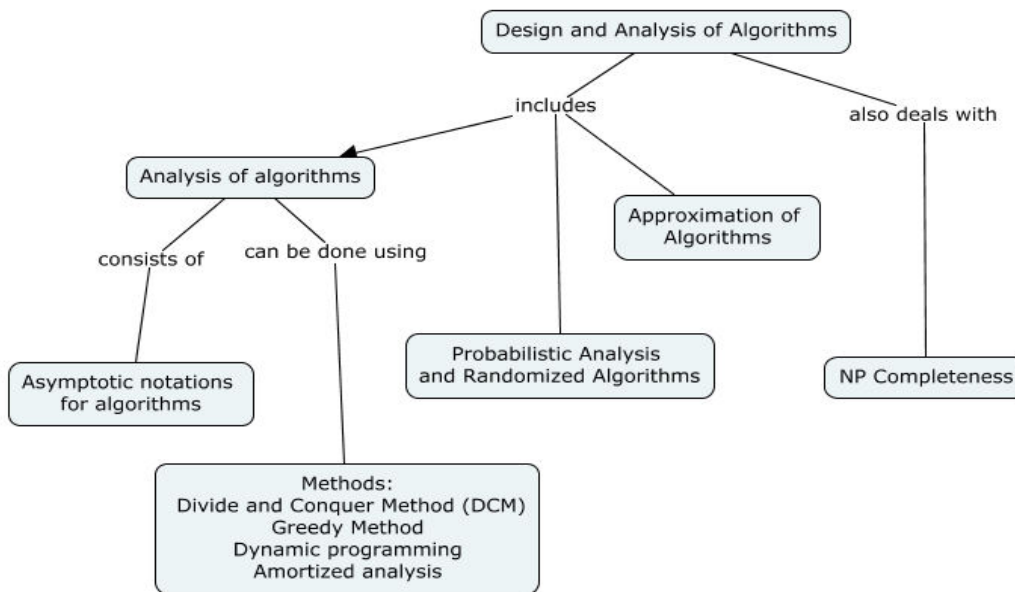
#### Analyze a given algorithm for its efficiency based on time and space it occupies (CO5):

1. Apply a recursive algorithm for solving Tower of Hanoi problem and analyze its efficiency for time complexity and space complexity.
2. How will you analyze a given algorithm for worst case, best case and average case analysis?
3. Write pseudocode for the brute-force method of solving the maximum sub-array problem.
4. Apply indicator random variables to compute the expected value of the sum of  $n$  dice.
5. Is it possible to improve the efficiency of an algorithm? Illustrate the steps involved in this process.

**To evaluate any given problem with mathematical rigor to provide an algorithmic based solution (CO6):**

1. Why do we analyze the expected running time of a randomized algorithm and not its worst-case running time?
2. Suppose that all element values are equal. What would be randomized quick-sort's running time in this case?
3. What is the smallest possible depth of a leaf in a decision tree for a comparison sort?
4. Evaluate the Reduction theorem  $MT \leq MQ$
5. Evaluate NP-Complete problems, indicating the reductions typically used to prove their NP-Completeness.

### Concept Map



### Syllabus

**Algorithms:** The Role of Algorithms in Computing - Algorithms as a technology; **Analyzing and Designing Algorithms;** Growth of Functions-Asymptotic notations - standard notations and common functions; Divide and Conquer Method-Solving recurrences; Probabilistic Analysis and Randomized Algorithms-Hiring problem-Randomized algorithm; **Dynamic programming**-Matrix Chain Multiplication-Elements of dynamic programming; **Greedy Algorithms**-Elements of Greedy Strategy; Amortized Analysis-Aggregate analysis-Dynamic tables; **NP-Completeness**-Polynomial time-NP-Complete problems; **Approximation algorithms**-The vertex-cover problem-travelling salesman problem-Randomization and linear programming.

### Reference Books

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to algorithms", Prentice-Hall, New Delhi, Third Edition, 2010. Chapters: 1 to 5, 15 to 17, 34 & 35.
2. M.A.Weiss, "Data Structures and Algorithm Analysis in C++", Pearson Education Asia, 2013.
- 3.. Anany Levitin, "Introduction to the Design and Analysis of Algorithms", Pearson Education, 2011.

### Course Contents and Lecture Schedule

Module. No	Topics	No. of Lectures
1	<b>Algorithms</b>	
1.1	Need for algorithms	1
1.2	Problem solving using algorithms	2
1.3	Role of algorithms in computing	2
1.4	Algorithms as a technology	1
1.5	Analysis of algorithms	1
1.6	Efficiency of algorithms	1
1.7	Algorithmic notations	2
2	<b>Divide and Conquer Method (DCM)</b>	
2.1	Introduction	1
2.2	Elements of DCM	1
2.3	Problem solving using DCM	2
3.	<b>Probabilistic Analysis and Randomized Algorithms</b>	
3.1	Introduction	2
3.2	Indicator random variables	1
3.3	Probabilistic analysis	1
3.4	Application of Indicator random variables	1
4.	<b>Dynamic programming</b>	
4.1	Elements of Dynamic programming	1
4.2	Matrix-chain multiplication	2
4.3	Optimization	1
5.	<b>Greedy algorithms</b>	
5.1	Elements of Greedy strategy	1
5.2	Matroids and greedy methods	2
5.3	Scheduling problem solving using Greedy method	2

6.	<b>Amortized analysis</b>	1
6.1	Aggregate analysis	1
6.2	Dynamic tables	1
7.	<b>NP-Completeness</b>	
7.1	Polynomial time	1
7.2	NP-Complete problems	2
8.	<b>Approximation Algorithms</b>	
8.1	The vertex-cover problem	1
8.2	The set-covering problem	2
8.3	Travelling-salesman problem	2
8.4	Randomization and linear programming	1
8.5	The subset-sum problem	2
Total		42

### Course Designer

Dr. S. Parthasarathy

spcse@tce.edu

<b>20CA240</b>	<b>OPERATING SYSTEMS</b>	Category	L	T	P	Credit
		PC	4	0	0	4

### Preamble

An operating system is an essential part of any computer system. The major objective of this course are to introduce basic concepts and principles of Windows and Linux operating systems which include memory management, device management, process management, file management and security & protection mechanisms.

### Prerequisites

- 20CA130: Computer Organization and Architecture

### Course Outcomes

On successful completion of the course, the students will be able to

CO1: Describe the evolution, types, structure and functions of operating systems	Understand
CO2: Explain techniques involved in process, memory, device and file management	Understand
CO3: Describe security and protection measures used in operating systems	Understand
CO4: Execute Linux basic commands and shell scripts	Apply
CO5: Implement processor scheduling, synchronization, deadlocks and disk allocation algorithms for a given scenario	Apply
CO6: Analyze the code for the resource allocation	Analyze

### Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	L	M	L	L	L							
CO2	L	M	M	L	M							
CO3	L	M	M	L	L							
CO4	S	S	S	S	S			L				
CO5	S	S	S	S	S			L				
CO6	L	S	M	L	M			L				

S- Strong; M-Medium; L-Low

### Assessment Pattern

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	20	20	20	20
Understand	50	30	30	30
Apply	30	40	40	40
Analyze	0	10	10	10
Evaluate	0	0	0	0
Create	0	0	0	0



### Course Level Assessment Questions

#### Describe the evolution, types, structure and functions of operating systems (CO1):

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.

#### Explain techniques involved in process, memory, device and file management (CO2):

1. Mention the three major activities of an operating system in regard to memory management.
2. 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.
3. Consider a system that supports the strategies of contiguous, linked, and indexed allocation. Which strategy is best utilized for a file? Justify your answer.
4. Suppose that a disk drive has 5000 cylinders, numbered 0 to 4999. The drive is currently serving a request at cylinder 143, and the previous request was at cylinder 125. The queue of pending requests in FIFO order is  
86, 1470, 913, 1774, 948, 1509, 1022, 1750, 150
5. 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)

#### Describe security and protection measures used in operating systems (CO3):

1. Explain the role of ACL in protection mechanism.
2. Compare and contrast the security models of various popular operating systems.
3. Explain the value of fault tolerance for disaster recovery.
4. Explain the implications of virtualization for disaster recovery.

#### Execute Linux basic commands & shell scripts (CO4):

1. Shell script to print the sum of all digits.
2. Shell script to print contents of a file from a given line number to the next given number of lines.
3. Shell script to say Good morning/Afternoon/Evening as you log in to the system.
4. Adding 2 nos. supplied as command line arguments.
5. Calculating the average of given numbers on command line arguments.
6. Finding out the biggest number from given three numbers supplied as command line arguments.

#### Implement processor scheduling, synchronization, deadlocks and disk allocation algorithms for a given scenario (CO5):

1. Consider the following set of processes, with the length of the CPU-burst time given in milliseconds:

Process Burst Time Priority

1 P1 10 3

. P2 1 1

P3 2 3

. P4 1 4

. P5 5 2

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

- Draw four Gantt charts illustrating the execution of these processes using FCFS, SJF, A non preemptive priority (a smaller priority number implies a higher priority), and RR (quantum = 1) scheduling.
  - What is the turnaround time of each process for each of the scheduling algorithms in part a?
  - What is the waiting time of each process for each of the scheduling algorithms in Part a?
  - Which of the schedules in part a results in the minimal average waiting time (over all processes)?
- Suppose that a disk drive has 5000 cylinders, numbered 0 to 4999. The drive is currently serving a request at cylinder 143, and the previous request was at cylinder 125. The queue of pending requests, in FIFO order, is 86, 1470, 913, 1774, 948, 1509, 1022, 1750, 130. Starting from the current head position, what is the total distance ((in cylinders) that the disk arm moves to satisfy all the pending requests, for each of the following disk scheduling a. FCFS b. SSTF c. SCAN d. LOOK e. C-SCAN
  - Analyze operating system requirements and recommend an appropriate operating system to meet the requirements.
  - Install several current operating systems and validate that the installations were successful.
  - Install a system with at least two virtual machines.

### Analyze the code for resource allocation (CO6):

- Would it be sensible to have a language that statically checks mutex correctness?

```
var m
var x guarded_by(m)

func f1() {
  lock(m)
  x = 42
  unlock(m)
}

func f2() {
  x = 42 // error, accessing x w/o holding its mutex
}

func f3() assumes_locked(m) {
  x = 42
}

func b1() {
  f3() // error
}

func b2() {
  lock(m)
  f3()
  unlock(m)
}
```

- Two threads are trying to share a memory area. You are given a wrong code for establishing mutual exclusion between threads as below:

**System:**

```
2
3 int favoredThread = 1;
4 boolean t1WantsToEnter = true;
5 boolean t2WantsToEnter = true;
6 startThreads();
```

<i>Thread T1:</i>	<i>Thread T2:</i>
-------------------	-------------------

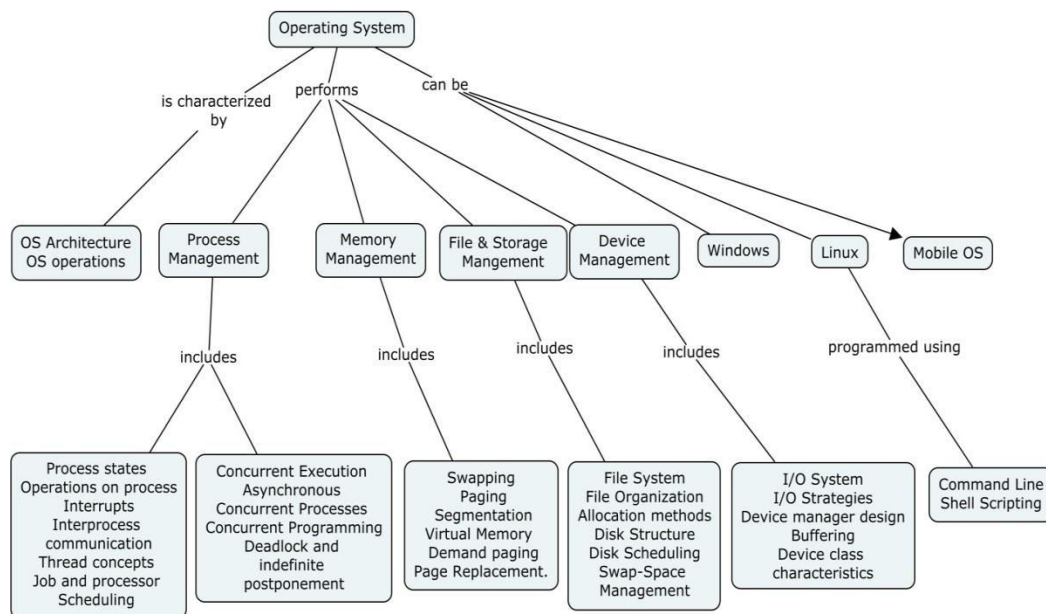
<pre> 11 void main() 12 { 13 while ( !done ) 14 { 15 t1WantsToEnter = false; 16 favoredThread = 2; 17 18 while ( t2WantsToEnter or favoredThread == 2 ); 19 20 // critical section code 21 22 t1WantsToEnter = true; 23 24 // code outside critical section 25 26 } // end while 27 28 } // end Thread T1 29                 </pre>	<pre> 32 void main() 33 { 34 while ( !done ) 35 { 36 t2WantsToEnter = false; 37 favoredThread = 1; 38 39 while ( t1WantsToEnter or favoredThread == 1 ); 40 41 // critical section code 42 43 t2WantsToEnter = true; 44 45 // code outside c r i t i c a l section 46 47 } // end while 48 49 } // end Thread T2                 </pre>
---	---

Scan each line of code and make necessary modifications. Explain the need for modifying the code by providing proper arguments in the given format. (10)

Modified Line number	Reason

Enhance the code so that indefinite postponement and deadlock are avoided and provide at least two case studies for proving the same. (10).

### Concept Map



### Syllabus

**Operating System Introduction** : Basics, OS Architecture, OS Operations. **Process Management:** Process states – Operations on process–Interrupts-Interprocess communication-Thread concepts -Job and processor Scheduling **Concurrent Execution:** Asynchronous Concurrent Processes- Concurrent Programming-Deadlock and indefinite postponement. **Memory Management:** Swapping, Paging, Segmentation, Virtual Memory – Demand paging, Page Replacement. **File & Storage Management:** File System, File Organization, Allocation methods, free space management, Disk Structure, Disk Scheduling, Swap-Space Management. **Linux Programming** - Command Line and Shell Scripting Basics. **Case Studies: Linux, Windows, Mobile Operating System.**

### Reference Books & web resources

1. William Stallings, "Operating systems Internal and Design Principles", Eighth Edition, Pearson Education, Global edition. 2015.
2. Andrew Tanenbaum, "Modern Operating Systems", Fourth Edition, Pearson Education, Global edition, 2014.
3. Abraham Silberschatz, Greg Gagne, Peter B. Galvin, "Operating System Concepts", 9<sup>th</sup> edition, Wiley, 2013.
4. H M Deital, P J Deital and D R Choffnes, "Operating Systems", Third Edition ,Pearson Education, (2004),Reprint 2012.
5. Jason Canon," Shell Scripting: How to Automate Command Line Tasks Using Bash Scripting and Shell Programming",Linux Training Academy,2015
6. Christine Bresnahan Richard Blum," Linux Command Line and Shell Scripting Bible", Third Edition, 2015,Wiley.
7. [http://nptel.ac.in/courses/Webcourse-contents/IIScBANG/Operating%20Systems/New\\_index1.html](http://nptel.ac.in/courses/Webcourse-contents/IIScBANG/Operating%20Systems/New_index1.html)
8. <https://www.shellscript.sh/first.html>
9. <http://www.freeos.com/guides/lst/>

### Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures
<b>1</b>	<b>Operating System Introduction</b>	
1.1	Basics, OS Architecture	2
1.2	OS Operations	1
<b>2</b>	<b>Process Management</b>	
2.1	Process states	1
2.2	Operations on Process	1
2.3	Interrupts	1
2.4	Inter process Communication	2
2.5	Thread concepts	2
2.6	Job and processor Scheduling	2
<b>3</b>	<b>Concurrent Execution</b>	

3.1	Asynchronous Concurrent Processes	2
3.2	Concurrent Programming	3
3.3	Deadlock and indefinite postponement	2
<b>4</b>	<b>Memory Management</b>	
4.1	Swapping	1
4.2	Paging	2
4.3	Segmentation	1
4.4	Virtual Memory – Demand paging	1
4.5	Page Replacement	3
<b>5</b>	<b>File &amp; Storage Management</b>	
5.1	File System	1
5.2	File Organization	1
5.3	Allocation methods and free space	1
5.4	Disk Structure & Disk scheduling	1
5.5	Swap-Space Management	1
<b>6</b>	<b>Linux Programming</b>	
6.1	Command Line	2
6.2	Shell Scripting Basics	4
<b>7</b>	<b>Case Studies</b>	
7.1	Linux	2
7.2	Windows	2
7.3	Mobile OS	2
	Total	44

**Course Designer:**

Dr. D. Anitha

anithad@tce.edu

<b>20CA250</b>	<b>DATA WAREHOUSING AND DATA MINING</b>	Category	L	T	P	Credit
		PC	3	1	0	4

### Preamble

This course aims at facilitating the student to understand the concepts of data warehousing and data mining. Students to understand the various techniques involved in mining the data from the databases.

### Prerequisite

- 20CA150 : Database management systems
- 20CA180 : RDBMS Laboratory

### Course Outcomes

On the successful completion of the course, students will be able to

CO1: Identify data mining tools and techniques in building intelligent machines	Understand
CO2: Analyze various data mining algorithms in applying in real time applications.	Analyze
CO3: Analyze unsupervised and supervised naive algorithms in real world applications	Analyze
CO4: Demonstrate the data mining algorithms to combinatorial optimization problems	Apply
CO5: Illustrate the mining techniques like association, classification and clustering on transactional databases.	Apply
CO6: Compare various supervised and unsupervised learning techniques in data mining	Analyze

### Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	L	S	L	M	-	-	-	-	-	-	-	-
CO2	M	-	-	M	-	-	-	-	-	-	-	-
CO3	M	S	M	-	-	-	-	-	-	-	-	-
CO4	S	M	M	-	-	-	-	-	-	-	-	-
CO5	S	S	-	M	-	-	-	-	-	-	-	-
CO6	M	M	S	-	-	-	-	-	-	-	-	-

S- Strong; M-Medium; L-Low

### Assessment Pattern

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	20	20	10	10
Understand	40	20	30	30
Apply	40	40	40	40
Analyse	0	20	20	20
Evaluate	0	0	0	0
Create	0	0	0	0

### Course Level Assessment Questions

#### Identify data mining tools and techniques in building intelligent machines (CO1):

1. List the various data sources for the data warehouse.
2. Distinguish between Data Mart and Data Warehouse?
3. What do you mean by strong association rule?
4. How to select an attribute for classification?
5. What is cluster analysis?
6. Mention the purpose of correlation analysis

#### Analyze various data mining algorithms in various logic and developing a real time applications (CO2):

1. What type of processing take place in a data warehouse? Describe.
2. Compare and contrast the clustering and the classification techniques.
3. In what way "Over Fitting" can be avoided?
4. Illustrate the significance of candidate set generation step of level wise algorithm.
5. Describe the importance of pruning in decision tree construction with an example.
6. Given the two vector objects  $X=(1,1,0,0)$  and  $Y=(0,1,1,0)$  identify the similarity between these objects.

#### Analyze unsupervised and supervised naive algorithms in real world applications (CO3):

1. What are the types of learning?
2. Compare Supervised neural networks with unsupervised neural networks.
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. Analyze the various Data mining techniques?

#### Apply data mining algorithms to combinatorial optimization problems (CO4):

1. Define: optimization
2. Suppose a group of 12 sales price records has been stored as follows:  
5, 10, 11, 13, 15, 35, 50, 55, 72, 92, 204, 215. Partition them into 3 bins by equal width binning.
3. Illustrate the apriori algorithm for the single dimensional transaction database.

**Apply the mining techniques like association, classification and clustering on transactional databases (CO5):**

1. For the given database find all the frequent item sets using Apriori method and list all the strong association rules that match the metarule

$$\forall x \in \text{transaction}, \text{buys}(X, \text{item1}) \wedge \text{buys}(X, \text{item2}) \Rightarrow \text{buys}(X, \text{item3}).$$

<i>TID</i>	<i>Items bought</i>	
100	{f, a, c, d, g, i, m, p}	Minimum Support = 30%
200	{a, b, c, f, l, m, o}	
300	{b, f, h, j, o, w}	
400	{b, c, k, s, p}	Minimum Confidence = 70%
500	{a, f, c, e, l, p, m, n}	

2. For the following Database use ID3 algorithm to construct the decision tree and partition the database based on the classification rules obtained from the decision tree.

Name	Rank	Years	Turned
Mike	Purchase Manager	3	No
Mary	Purchase Manager	7	Yes
Bill	Sales Manager	2	Yes
Jim	Production Manager	7	Yes
Dave	Purchase Manager	6	No
Anne	Production Manager	3	No

3. For the following Database, apply the entropy-based discretization for the numerical attribute and find the best split.

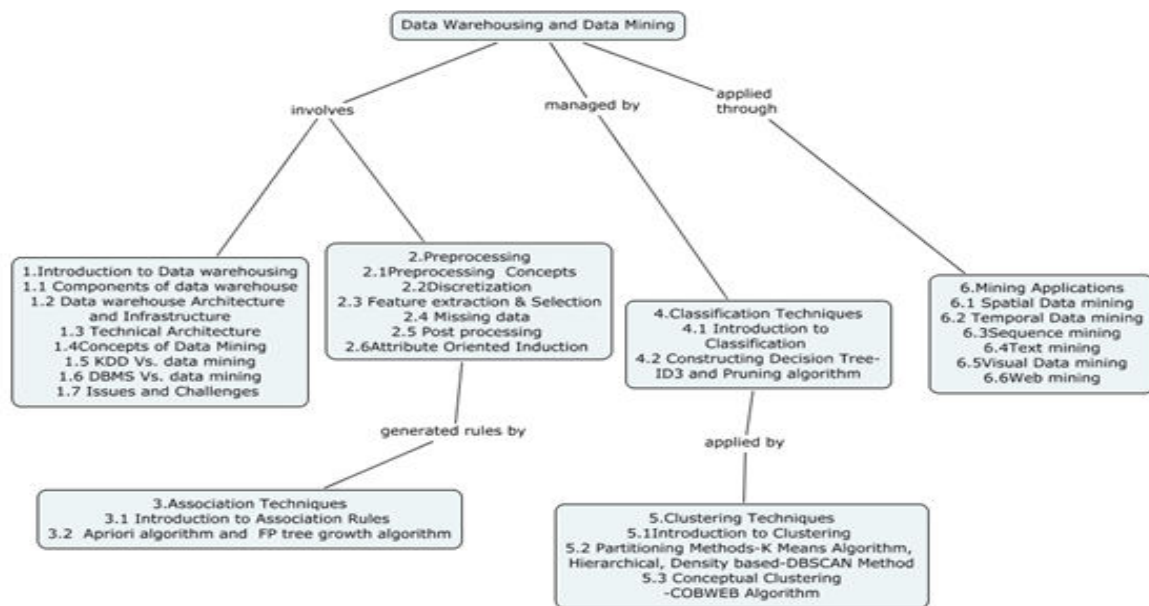
S.NO	Age	Credit rating
1.	25	Fair
2.	29	Excellent
3.	35	Fair
4.	42	Excellent
5.	47	Fair
6.	49	Excellent
7.	32	Fair
8.	34	Fair
9.	37	Excellent
10.	40	Fair
11.	44	Fair
12.	45	Excellent

**Analyze various supervised and unsupervised learning techniques in data mining (CO6)**



Given two objects  $A_1(22,1,42,10)$  and  $A_2(20,0,36,8)$  compute the distance by Euclidean measure. The data mining task wants to Cluster the following eight points (with  $(x,y)$  representing locations) into 3 clusters  $A_1(2,10)$ ,  $A_2(2,5)$ ,  $A_3(8,4)$ ,  $B_1(5,8)$ ,  $B_2(7,5)$ ,  $B_3(6,4)$ ,  $C_1(1,2)$ ,  $C_2(4,9)$ . The distance function is Euclidean distance. Initially assign  $A_1$ ,  $B_1$  and  $C_1$  as the center of each cluster respectively. Use K-Means algorithm to show the final three clusters.

## Concept Map



## Syllabus

**Introduction to Data Warehousing** – Components of data warehouse, Datawarehouse Architecture and Infrastructure. Technical Architecture and OLAP. **Data Mining** – Concepts , KDD vs Data mining , DBMS vs Data mining , Issues and Challenges, **Preprocessing** – Concepts , Discretization , Feature extraction & Selection , Missing data , Post processing , Attribute Oriented Induction, **Association Techniques** - Introduction Association Rules , Apriori algorithm , FP tree growth algorithm , Types of association rules, **Classification Techniques** - Introduction to Classification , Constructing decision tree – ID3 algorithm , Pruning. **Clustering Techniques** - Introduction to Clustering , Partitioning Method – K Means algorithm , Hierarchical Method , Density Based Method – DBSCAN method , Conceptual clustering – COBWEB algorithm, **Mining Applications** - Spatial data mining, Temporal data mining, Sequence mining, Text mining, Visual data mining, Web mining.

## Reference Books

1. Jiawei Han, Micheline Kamper, Data Mining: Concepts and Techniques Morgan Kaufman, 2007, ISBN: 1-55860-489-8. Chap1-3, 5-10.
2. K.P.Soman, Shyam Diwakar, V.Ajay, "Insight into Data Mining – Theory and Practice", Prentice Hall of India, 2009. (Modules II, III and VI)

3. Arun K.Pujari, "Data Mining Techniques", Universities Press, 2010. (Modules I, IV, V and VI )
4. M.H Dunham, "Data Mining: Introductory and advanced topics", Pearson Education, 2006.

### Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures
<b>1</b>	<b>Introduction to Data Warehousing</b>	
1.1	Components of data warehouse	2
1.2	Data warehouse Architecture and Infrastructure	2
1.3	Technical Architecture and OLAP	1
1.4	Data mining – basic concepts	1
1.5	Knowledge Discovery in Databases Vs. data mining	2
1.6	Database Management Systems Vs. data mining	1
1.7	Issues and Challenges	1
<b>2.</b>	<b>Processing</b>	
2.1	Preprocessing Basics	1
2.2	Discretization	1
2.3	Feature extraction & Selection	1
2.4	Missing data	1
2.5	Post processing Basics	2
2.6	Attribute Oriented Induction	1
<b>3.</b>	<b>Association Techniques</b>	
3.1	Introduction to Association Rules	1
3.2	Association Algorithms ( Apriori, FP tree )	3
<b>4</b>	<b>Classification Techniques</b>	
4.1	Introduction to Classification	1
4.2	Classifiers ( Decision tree, Pruning )	3
<b>5</b>	<b>Clustering Techniques</b>	
5.1	Introduction to Clustering	1
5.2	Algorithms ( Partitioning, Hierarchical, Density based)	4
<b>6</b>	<b>Mining Applications</b>	
6.1	Spatial Data mining	1
6.2	Temporal Data mining	2
6.3	Sequence mining	3
6.4	Text mining	1
6.5	Visual Data mining	2
6.6	Web mining	1
	Total	<b>40</b>

### Course Designer:

Dr. S. Vijayalakshmi

[svlcse@tce.edu](mailto:svlcse@tce.edu)

<b>20CA270</b>	<b>C++ PROGRAMMING LABORATORY</b>	Category	L	T	P	Credit
		PC	0	0	2	2

### Preamble

This Laboratory course will enable students to identify, formulate all techniques of software development in the C++ Programming Language and demonstrate these techniques by the solution of a variety of problems spanning the breadth of the language.

### Prerequisite

- 20CA120: Programming in C

### Course Outcomes

On the successful completion of the course, students will be able to

CO1: Explain the concepts of oops for building object based applications	Understand
CO2: Write a program in different logic with suitable validations for a given problem	Apply
CO3: Implement the techniques and features of the Object Oriented Programming constructs to construct an application	Apply
CO4: Implement method overloading and method overriding for different user specifications	Apply
CO5: Write programs implementing inheritance for an application domain	Apply
CO6: Write technical report on the observations from the experiments	Apply

### Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M	S	S	S	-	-	-	-	-	-
CO2	S	M	S	S	M	-	-	-	-	-	-	-
CO3	S	M	M	S	S	S	-	-	-	-	-	-
CO4	S	S	M	S	S	M	-	-	-	-	-	-
CO5	S	S	S	S	S	M	-	-	-	-	-	-
CO6	S	S	S	S	S	M	-	-	-	-	-	-

S-Strong;M-Medium;L-Low

## List of Experiments

Develop C++ programs for

1. Constructor and copy constructor.
2. Storage classes like auto, extern, register and static.
3. Static member data, static member function and bitwise operators.
4. Overloading and method overriding.
5. Inheritance
6. Pointer Arithmetic.
7. Inline Functions.
8. Functions & Recursion.
  - a. Recursion
  - b. Function with "this" pointer
9. Friend Function & Friend Class.
10. Exception handling methods.
11. Overload Unary & Binary Operators as Member Function & Non Member Function.
  - a. Unary operator as member function
  - b. Binary operator as non member function
12. Class Templates

- **Mini Project :**

The course instructor shall provide real time problems/specifications to the students for mini project. The project shall be completed before the commencement of 3<sup>rd</sup> semester and a report shall be submitted.

- **Sample specification for mini project:**

Develop a C++ processor that performs stemming on the words in a document.

Simulate a client server process in which the server process clients on a priority basis.

- **Note:**

For all exercises, students shall develop C++ programs in any one application domain (e.g. Banking, Online shopping, Employee management etc.)

## Reference Books

1. Bjarne Stroustrup , "The C++ Programming Language", 4<sup>th</sup> Edition, Addison-Wesley, 2015.
2. Scott Meyers, "Effective C++ 55 Specific Ways to Improve Your Programs and Designs", Third Edition, Addison-Wesley, 2011
3. Paul Deital & Harvey Deital, "C++ How to Program", 7<sup>th</sup> Edition, Pearson Education, 2010
4. Stanley Lippman, "C++ Primer", 4th Edition, Pearson Education, 2007.
5. Yashavant P. Kanetkar, "Let Us C++", BPB Publications, 2007.
6. Robert Laffore, "Object Oriented Programming using C++", 4<sup>th</sup> Edition, Sams Publishing, 2002.

## Course Designer:

Prof. N. Murali

nmcse@tce.edu

<b>20CA280</b>	<b>ALGORITHMS IMPLEMENTATION AND ANALYSIS USING C LABORATORY</b>	Category	L	T	P	Credit
		PC	0	0	2	2

### Preamble

This Laboratory will enable students to identify, formulate and solve real world engineering problems that require usage of algorithms.

### Prerequisite

- 20CA120: Programming in C
- 20CA140: Data Structures
- 20CA170: Data Structures using C Programming Laboratory

### Course Outcomes

On the successful completion of the course, students will be able to

CO1: .Implement any balanced tree in C	Apply
CO2: Develop C programs implementing graph algorithms in application context	Apply
CO3: Develop C programs for implementing greedy approach, dynamic programming and backtracking techniques	Apply
CO4: Write C programs by modifying existing programs/algorithms based on user specification	Analyze
CO5: Analyze time complexity and space complexity of algorithms	Analyze
CO6: Write technical reports on the observations in the experiments	Apply

### Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	S	L	S	S	M	-	-	-	-	-
CO2	S	S	S	L	S	S	M	-	-	-	-	-
CO3	S	S	S	L	S	S	M	-	-	-	-	-
CO4	S	S	S	L	S	S	M	-	-	-	-	-
CO5	S	S	S	L	S	S	M	-	-	-	-	-
CO6	S	L	L	L	L	M	L	-	M	-	-	-

S- Strong; M-Medium; L-Low

## Lab experiments

Develop C programs for

1. Linear Search and Binary search
2. Implementation of any one balanced tree structures
3. Graph Traversals (Breadth First and Depth First search)
4. Spanning Tree Implementation (Prims/ Kruskals)
5. Shortest Path Algorithms (Dijkstra's algorithm/ Bellmann Ford Algorithm)
6. Knapsack implementation (Greedy/Dynamic programming/Back tracking)
7. Huffman Coding Implementation.
8. Travelling Salesman problem
9. Euclids algorithm for finding GCD

## Mini Project :

The course instructor shall provide real time problems/specifications to the students for mini project. The project shall be completed before the commencement of 3<sup>rd</sup> semester and a report shall be submitted.

Sample specification for mini project:

1. Develop a C program that finds the run time efficiency of any C program given as input.
2. Develop a visualization tool that demonstrates the working of algorithms.

**Note:** Algorithm analysis report comprising time complexity, space complexity shall be submitted by the students and verified by the course instructor at the lab session for each experiment.

## Reference Books & web resources

1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Printice hall International, 2010.
2. Brian W Kernighan & Dennis Ritchie, "The C programming language", 2nd Edition, Prentice Hall ,2015
3. Yashavant Kanetkar," Let us C", BPB Publications 8th Edition, 2014
4. Fundamentals of Algorithmics, Gilles Brassard and Paul Bratley, Printice hall International, 2002.
5. Mark Allen Weiss,"Data Structures and Algorithm Analysis in C", Pearson Education, 2011.
6. Robert Kruse & Clovis L. Tondo " Data Structures and Program Design in C",Prentice Hall , 2012.

## Course Designers

Dr.D.Anitha                      anithad@tce.edu  
Dr.T.Chandrakumar            tckcse@tce.edu

<b>20CA290</b>	<b>PROFESSIONAL COMMUNICATION</b>	Category	L	T	P	Credit
		HSS	0	1	1	2

### Preamble

This course provides opportunities to students to develop and demonstrate basic communication skills in technical, professional and social contexts effectively.

### Prerequisite

- None

### Course Outcomes

On the successful completion of the course, students will be able to

<b>CO1:</b>	Listen, watch, understand and confidently respond to talks, conversations, etc by native and neutral speakers on science, general context, and from ETS samples.	<b>Apply</b>
<b>CO2:</b>	Present ideas, express opinions/comments, practice presentations, and converse in discussions on a variety of technical and non-technical domains without fear	<b>Apply</b>
<b>CO3:</b>	Comprehend texts / passages, solve verbal aptitude questions related to placement and higher studies	<b>Apply</b>
<b>CO4:</b>	Communicate in writing for journals and business correspondences with clarity, accuracy, intelligibility, and precision.	<b>Apply</b>

### Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1							M		S			
CO2							M		S			
CO3							M		S			
CO4							M		S			

S- Strong; M-Medium; L-Low

### Assessment Pattern

#### Internal

**No Common Continuous Assessment Test (CAT) will be conducted.**

Students' performance will be continuously assessed in various classroom activities on Listening, Speaking, Reading and Writing for 50 marks as detailed below

Listening Test	- 10
Speaking Test- Presentation (Technical / Review: Movie/Book) & GD	- 20
Written Test – Verbal Aptitude, E-mail / Blog	- 20

(The test will be conducted for 50 marks and reduced to 20)

**External (Practical)**

Test – Listening (20), Reading (20)	- 40
Group Discussion	- 25
Personal Interview / Situational Conversation	- 25
Resume Submission	- 10

**List of Experiments**

Sl. No.	Topic	No. of Hours	
		Tutorial	Practical
1	Characteristics of Technical Writing	2	
2	Development of Employability Skills	2	
3	Vocabulary Development	2	
4	Sentence Completion	1	
5	Error Spotting	1	
6	Interpretation of Verbal Analogy	1	
7	Interpretation of Reading (Comprehension - Conception)	1	
8	Interpretation of Reading (Comprehension - Reasoning)	1	
9	Practice for writing E-mails/Technical Blogs/Forums	1	
10	PPT Preparation / Demonstration of Technical Presentation		2
11	Preparation of Resume		1
12	Preparation for Job Interviews		2
13	Demonstration of Group Discussion Skills		2
14	Developing Listening Skill (Comprehension)		2
15	Practice for Short Speeches / Situational Conversation		1
16	Review : English Movies / Novels		2
Total Hours		12	12
*(Any two English movies and two novels shall be discussed based on students interest and relevance )			

**Course Designers:**

1	Dr. S. Rajaram	<a href="mailto:sreng@tce.edu">sreng@tce.edu</a>
2	Dr.A.Tamilselvi	<a href="mailto:tamilselvi@tce.edu">tamilselvi@tce.edu</a>
3	Mr. R. Vinoth	<a href="mailto:vino@tce.edu">vino@tce.edu</a>
4	Dr. G.Jeya jeevakani	<a href="mailto:gjjeng@tce.edu">gjjeng@tce.edu</a>
5	Ms. R.Manibala	<a href="mailto:rmaeng@tce.edu">rmaeng@tce.edu</a>



**20CA310****OPERATIONS RESEARCH**

Category	L	T	P	Credit
BS	4	0	-	4

**Preamble**

The course aims at exploring the various problems like linear programming, Integer programming, Transportation, assignment problem and the project network analysis. It covers the various inventory and queuing models

**Prerequisite**

None

**Course Outcomes**

On the successful completion of the course, students will be able to

CO1:	Identify the Transportation and Assignment problems and to optimize in engineering fields	Understand
CO2:	Apply the various sequencing techniques	Apply
CO3:	Solve linear programming techniques to optimization problems arising in all Computer fields	Apply
CO4:	Solve Integer linear programming techniques to optimization problems arising in all Computer fields	Apply
CO5:	Summarize the inventory and queuing models	Understand
CO6:	Classify the various scheduling techniques in mini Project	Analyze

**Mapping with Programme Outcomes**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	L	-	-	-	-							
CO2	M	M	M	M	L		M			M		
CO3	M	M	M	M	L				M			
CO4	M	M	M	M	L		M					
CO5	M	M	M	M	I		M			M		
CO6	-	S	M	S	M							

S- Strong; M-Medium; L-Low

### Assessment Pattern

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	10	10	10	-
Understand	30	30	20	30
Apply	60	60	70	70
Analyse	0	0	0	0
Evaluate	0	0	0	0
Create	0	0	0	0

### Course Level Assessment Questions

**Identify the Transportation and Assignment problems and to optimize in engineering fields (CO1) :**

- At the end of a cycle of schedules, a transport company has a surplus of one truck in each of the cities 1,2,3,4,5 and a deficit of one truck in each of the cities A,B,C,D,E and F. the distance between the cities with a surplus and cities with a deficit are given below.

		To city					
		A	B	C	D	E	F
From city	1	80	140	80	100	56	98
	2	48	64	94	126	170	100
	3	56	80	120	100	70	64
	4	99	100	100	104	80	90
	5	64	80	90	60	60	70

How should the truck be dispatched so as to minimize the total distance traveled ? which city will not receive a truck?

- Solve the assignment problem for maximization given the profit matrix,

	P	Q	R	S
A	51	53	54	50
B	47	50	48	50
C	49	50	60	61
D	63	64	60	60

- Define non existing feasible solution?
- State all the constraints in a transportation problem and how they are different from linear programming problem?
- What are assignment problems? What is the difference between transportation and assignment problem.

**Apply the various sequencing techniques (CO2) :**

- Calculate the total float, free float and independent float for the project whose activities are given below.

Activity	1-2	1-3	1-5	2-3	2-4	3-4	3-5	3-6	4-6	5-6
Duration	8	7	12	4	10	3	5	10	7	4

**Solve linear programming techniques to optimization problems arising in all Computer fields (CO3):**

1. Solve by simplex method

$$\text{Max } Z = X_1 + 2X_2 + 3X_3 - X_4$$

Subject to,

$$X_1 + 2X_2 + 3X_3 = 15$$

$$2X_1 + X_2 + 5X_3 = 20$$

$$X_1 + 2X_2 + X_3 + X_4 = 10 \text{ and}$$

$$X_1, X_2, X_3, X_4 \geq 0$$

2. Solve the LPP.

$$\text{Min } Z = X_1 + X_2 + X_3$$

Subject to,

$$X_1 - 3X_2 + 4X_3 = 5$$

$$X_1 - 2X_2 \leq 3$$

$$2X_2 - X_3 \geq 4$$

$$X_1, X_2 \geq 0 \text{ and } X_3 \text{ is unrestricted.}$$

**Solve Integer programming techniques to optimization problems arising in all Computer fields (CO4):**

1. Solve the following mixed integer problem.

$$\text{Max } Z = -3X_1 + X_2 + 3X_3$$

Subject to,

$$-X_1 + 2X_2 + X_3 \leq 4$$

$$2X_2 - 3/2 X_3 \leq 1$$

$$-X_1 - 3X_2 + 2X_3 \leq 3$$

$$X_1, X_2 \geq 0 \text{ and } X_3 \text{ is non negative integer.}$$

**Summarize the inventory and queuing models (CO5):**

1. A branch of a National bank has only one typist. Since the typing work varies in length. The typing rate is randomly distributed. Approximating Poisson distribution with mean rate of 8 letters per hour, the letters arrive at the rate of 5 per hour. During the entire 8 hour work day if the PC is valued at Rs. 15.0/- per hour. Determine
- Equipment utilization.
  - The percent time an arriving letters has to wait.
  - Average system time.
  - Average idle time cost of the type writer per day.

**Classify the various scheduling techniques in mini Project (CO6) :**

1. Construct the network for the project whose activities are given below and calculate the total float, free float and independent float for the project.

Activity	0-1	1-2	1-3	2-4	2-5	3-4	3-6	4-7	5-7	6-7	
Duration		3	8	12	6	3	3	8	5	3	8

Determine the critical path and project duration of the project.

2. A maintenance foreman has given the following estimate of times and cost of jobs in a maintenance project.

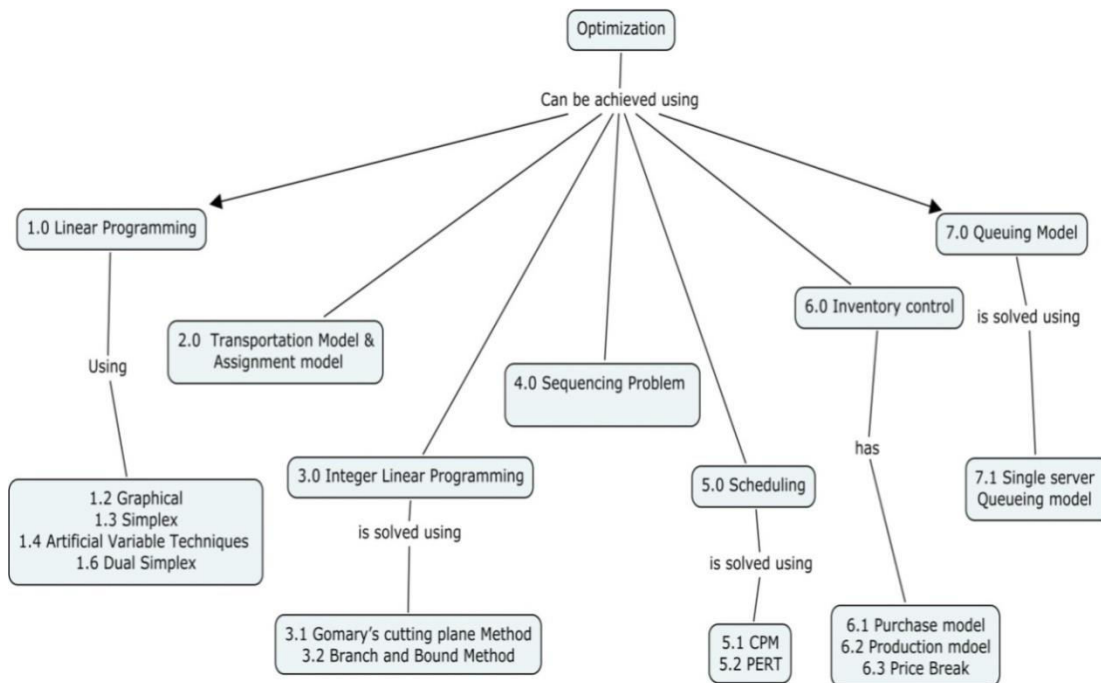
Job	Predecessor	Time	Normal		Crash	
			Cost	Time	Cost	Time
A	-	8	80	6	100	

B	A	7	40	4	94
C	A	12	100	5	184
D	A	9	70	5	102
E	B,C,D	6	50	6	50

Overhead cost is Rs. 23 per hour. Find

- the normal duration of the project and the associated cost
  - the minimum duration of the project and associated cost
  - the least duration of the project and its cost
  - if all the activities are crashed what will be the project duration and the correspond cost
3. Analyze the practical steps involved in solving PERT problem.

**Concept Map**



**Syllabus**

**Linear Programming**-Graphical Solution- The Simplex algorithm, Artificial Variable Technique - Duality-Dual Simplex - Variants of the Simplex Method **Transportation Model**- Initial Basic Feasible Solution methods Test for optimality-Variants of the Transportation problem **Assignment Model**- Hungarian algorithm Variants of the Assignment problem, Travelling Salesman Problem **Integer Linear Programming**- Gomary's cutting plane method Branch and Bound method **Sequencing Problem** - N jobs through 2 machines, N Jobs through 3 machines, N jobs through m machines **Scheduling** - Critical path Method, Project Evaluation and Review Techniques **Inventory control** - Purchase and production model with and without shortage , price break **Queuing Model**- Single channel model, Multichannel model.

**Reference Books**

1. Sharma J.K.: "Operations Research Theory and applications", Macmillan India Ltd., V Edition, 2015.
2. Hamdy A. Taha: Operations Research – An Introduction", Prentice Hall of India Pvt Ltd., EIGHT Edition, 2014.
3. Chandrasekara Rao, K. Shanti Lata Misra "Operation Research", Alpha science international Ltd-2015.
4. Kanti Swarup, P.K.Gupta and Man Mohan "Operations Research", Sultan Chand,2014

### Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures
1.0	<b>Linear Programming</b>	
1.1	Formulation	1
1.2	Graphical Solution	1
1.3	The Simplex algorithm	2
1.4	Artificial Variable Techniques	2
1.5	Variants of the simplex methods	2
1.6	Duality-Dual Simplex	2
2.0	<b>Transportation Model</b>	
2.1	Initial Basic Feasible Solution methods	2
2.2	Test for optimality-Variants of the Transportation problem	2
	<b>Assignment Model</b>	
2.3	Hungarian Algorithm	2
2.4	Variants of the Assignment Problem	2
2.5	Travelling Salesman problem	1
3.0	<b>Integer Linear Programming</b>	
3.1	Gomary's cutting plane method	3
3.2	Branch and Bound method	2
4.0	<b>Sequencing Problem</b>	
4.1	N jobs through 2 machines, N Jobs through 3 machines, N jobs through m machines	2
4.2	Processing Two jobs through m machines	2
5.0	<b>Scheduling</b>	
5.1	Critical path Method	2
5.2	Project Evaluation and Review Techniques	2
6.0	<b>Inventory control</b>	

6.1	Purchase model with and without shortage	2
6.2	Production model with and without shortage	1
6.3	Price Break	1
7.0	<b>Queuing Model</b>	
7.1	Single server Queueing model	3
7.2	Multi channel model	2
	Total	42

**Course Designer**

Dr. M.Kameswari

mkmat@tce.edu



<b>20CA320</b>	<b>INTERNETAND JAVA PROGRAMMING</b>	Category	L	T	P	Credit
		PC	4	0	0	4

**Preamble**

This course aims at facilitating the student to understand the basic internet programming concepts and the programming concepts of JAVA towards developing Java based applications and hands on practices by applying the concepts for implementing internet applications.

**Prerequisite**

- 20CA220: Object Oriented Programming using C++

**Course Outcomes**

On the successful completion of the course, students will be able to

CO1: Use the object oriented concepts of java for the given problem.	Understand
CO2: Use exceptions, threads, collections, logs of Java for the given problem.	Understand
CO3: Apply events through swing, RMI, JAR operations for the given application	Apply
CO4: Select the proper library classes in Java based on the need of a problem	Apply
CO5: Apply different Java technologies to solve Internet applications.	Apply
CO6: Design an application using DHTML and JSP	Apply

**Mapping with Programme Outcomes**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	L	M	L	L	L							
CO2	L	M	L	L	L							
CO3	S	L	L	L	L			L				
CO4	S	S	S	L	S			L				
CO5	S	M	L	M	S			L				
CO6	S	M	S	M	S			L				

S- Strong; M-Medium; L-Low

**Assessment Pattern**

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	30	20	20	20
Understand	30	30	30	30
Apply	40	50	50	50
Analyze	0	0	0	0
Evaluate	0	0	0	0
Create	0	0	0	0

**Course Level Assessment Questions****Use the object oriented concepts of java for the given problem (CO1):**

1. Write a Java Program to implement multilevel inheritance for a university-college database management system.
2. Implement method overloading for calculating the area of different shapes.
3. Write a program to define a class for a student result processing system.

**Use exceptions, threads, collections, logs of Java for the given problem (CO2):**

1. Write a Java Program to print the numbers 1 to 50 alternatively by Threads (Use Multithreading).
2. Illustrate with an example how collections are manipulated using array list

**Apply events through swing, RMI, JAR operations for the given application (CO3):**

1. Write a simple registration form for any event using swing.
2. Illustrate action event using button.
3. Write a code segment to give feedback for a course using swing.
4. Write a sample Java program using adjustment event.
5. List the listeners in java.
6. Write an RMI based Application Program for addition of two complex numbers.

**Select the proper library classes in Java based on the need of a problem (CO4):**

1. Design an application for Library Management System
2. Implement Railway Reservation system
3. Implement a Quiz application

**Apply different Java technologies to solve Internet applications (CO5):**

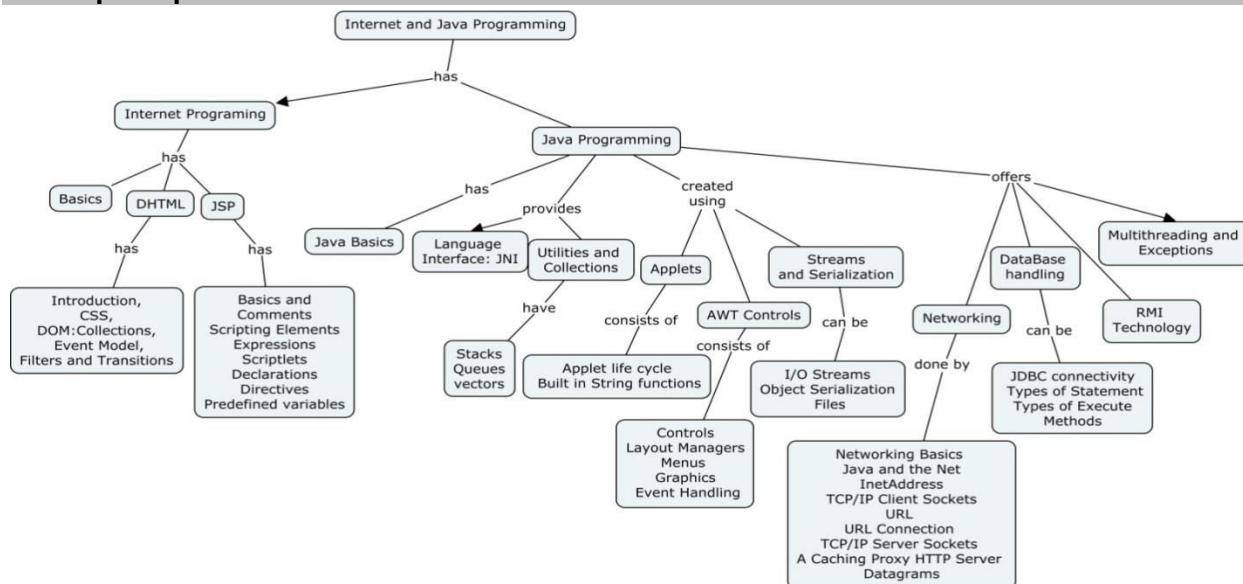
1. Develop a sample client/server application using sockets and datagram packets.
2. Develop an Applet for 'Online Job Portal' by applying the event handlers to handle the events.
3. Develop a user defined exception for handling a negative number or zero during age validation in a Voter Management System using Exception base class.

**Design an application using HTML/DHTML and JSP (CO6):**

1. Develop an application for online shopping
2. Develop an application for online cab booking
3. Design an application for online Ticket reservation system for air bus.



## Concept Map



## Syllabus

**Internet Programming:** Internet Basics, DHTML : Introduction - CSS - DOM:Collections- Event Model - Filters and Transitions. **Java Programming:** Java Basics:Java features, Array and Strings, Object Oriented Features, Language Interface: JNI, Utilities and Collections, Stacks, Queues and vectors **Multithreading and Exceptions:** Multithreaded Programming, Exception Handling, **Applets:** Applet life cycle, Built in String functions, Streams and serialization: I/O Streams, Object Serialization, Files, AWT Controls: Controls, Layout Managers, Menus, Graphics, Event Handling. **JSP :** Basics and Comments - Scripting Elements - Expressions - Scriptlets - Declarations- Directives - Predefined variables. **Database Handling:** JDBC connectivity, Types of Statement, Types of Execute Methods. **Networking:** Networking Basics - Java and the Net – InetAddress – TCP/IP Client Sockets – URL –URL Connection – TCP/IP Server Sockets – A Caching Proxy HTTP Server – Datagrams, RMI Technology.

## Reference Books & web resources

1. Herbert Schildt, "Java the Complete Reference", 9th Edition, McGraw Hill, 2014.
2. Margaret Levine Young, "The Internet - Complete Reference" , 2nd Edition, Tata McGraw Hill, 2002, (Reprint 2016).
3. Paul Deitel, Harvey Deitel, Abbey Deitel, "Internet and WWW How to Program", 5th Edition, Tata McGraw Hill, 2011.
4. [nptel.ac.in/courses/106105084/](http://nptel.ac.in/courses/106105084/)
5. [nptel.ac.in/courses/106105084/28](http://nptel.ac.in/courses/106105084/28)
6. [nptel.ac.in/courses/106105084/30](http://nptel.ac.in/courses/106105084/30)
7. [www.nptelvideos.in/2012/11/internet-technologies.html](http://www.nptelvideos.in/2012/11/internet-technologies.html)
8. [www.nptelvideos.com/java/java\\_video\\_lectures\\_tutorials.php](http://www.nptelvideos.com/java/java_video_lectures_tutorials.php)

**Course Contents and Lecture Schedule**

<b>Module No.</b>	<b>Topic</b>	<b>No. of Lectures</b>
<b>1</b>	<b>Internet Programming</b>	
1.1	Internet Basics,DHTML,CSS	1
1.2	DOM:Collections	1
1.3	Event Model	1
1.4	Filters and Transitions	1
<b>2</b>	<b>Java Programming</b>	
2.1	Java Basics:Java features	1
2.2	Array and Strings	1
2.3	Object Oriented Features	1
<b>3</b>	<b>Language Interface</b>	
3.1	JNI	2
<b>4</b>	<b>Utilities and Collections</b>	
4.1.	Stacks, Queues and vectors	3
<b>5</b>	<b>Multithreading and Exceptions</b>	
5.1	Multithreaded Programming	2
5.2	Exception Handling	2
<b>6</b>	<b>Applets</b>	
6.1	Applet life cycle	1
6.2	Built in String functions	1
<b>7</b>	<b>Streams and serialization</b>	
7.1	I/O Streams	1
7.2	Object Serialization	1
7.3	Files	1
<b>8</b>	<b>AWT Controls</b>	
8.1	Controls	1
8.2	Layout Managers	1
8.3	Menus	1
8.4	Graphics	1
8.5	Event Handling	2
<b>9</b>	<b>Database Handling</b>	
9.1	JDBC connectivity	1
9.2	Types of Statement	1
9.3	Types of Execute Methods	1
<b>10</b>	<b>JSP</b>	
10.1	Basics and Comments	1
10.2	Scripting Elements	1
10.3	Expressions - Scriptlets	1
10.4	Declarations- Directives & predefined variables	2
<b>11</b>	<b>Networking</b>	
11.1	Networking basics	1
11.2	Socket Programming	3
11.3	RMI Technology	1
	Total	40

**Course Designer:**

Dr. D. Anitha                      anithad@tce.edu

<b>20CA330</b>	<b>OBJECT ORIENTED MODELING AND DESIGN PATTERNS</b>	Category	L	T	P	Credit
		PC	3	0	0	3

### Preamble

This course aims at facilitating the student to learn the object orientation on real world problems; analyze and design the problem domain using the principles and practices followed in object oriented problem solving by applying object technology with UML modelling.

### Prerequisite

- 20CA220 : Object Oriented Programming using C++

### Course Outcomes

On the successful completion of the course, students will be able to	
CO1: Understand the complexity of Industrial Strength Software and the application of Unified Process Model.	Understand
CO2: Identify classes and objects using Object Oriented Analysis techniques	Understand
CO3: Develop new classes from the problem domain by applying object oriented design principles	Apply
CO4: Incorporate design patterns to create classes with reusability and extensibility.	Apply
CO5: Design different design documents to show logical view using UML for any given application	Analyze
CO6: Develop different design documents to show physical view using UML for any given application .	Analyze

### Mapping with Programme Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	P12
CO1	S	S	S	M	M	L	L	L	L	-	-	-
CO2	S	S	S	M	M	L	L	L	L	-	-	-
CO3	S	S	S	M	M	S	S	L	L	-	-	-
CO4	S	S	S	M	L	S	S	L	L	-	-	-
CO5	S	S	S	M	L	S	S	L	L	-	-	-
CO6	S	S	S	M	L	S	S	L	L	-	-	-

S- Strong; M-Medium; L-Low

### Assessment Pattern

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	20	20	20	20
Understand	20	20	20	30
Apply	40	40	20	30
Analyse	20	20	40	20
Evaluate	0	0	0	0
Create	0	0	0	0

### Course Level Assessment Questions

**Understand the complexity of Industrial Strength software and the application of Unified Process Model. (CO1):**

1. Write the reasons of Software Crisis? How to avoid it?
2. Why software is inherently complex? List the reasons.
3. How OO based decomposition differs from algorithmic decomposition.
4. What is Unified Process Model?
5. Explain the different life cycle phases of Rational Unified Process Model.

**Identify classes and objects from the given problem domain by applying the different Object Oriented Analysis (OOA) based classification techniques (CO2):**

1. What is OOA?
2. Which tasks are involved in the OOA process?
3. Explain the different ways of classifying the objects and classes in OOA?
4. How CRC card based analysis helps in identifying classes and objects?
5. Companies may employ many people, and people may work for many companies. Every employee in a company has a manager to manage many subordinate employees. One work role is assigned to each employee and each job requires and desired credentials. The credentials are composed of qualifications. Identify the classes from this scenario

**Develop new classes from the problem domain by applying object oriented design principles. (CO3)**

1. What are design principles? How the various types of it are used in OOD? Explain it with an example for each.
2. What is LSP?
3. What is OCP?
4. When will you apply Common Closure Principle (CCP)?
5. What is Release Reuse Principle?

**Incorporate design patterns to create classes with reusability and extensibility. (CO4)**

1. What are design patterns? Analyze the creational based design patterns in classes and objects representation in OOD.
2. What are structural design patterns? Analyze how the different types of it are applied in OOD?
3. How behavioral patterns are used in representing the functionality in OOD?
4. What is the purpose of Memento? Give an example.
5. What is a framework? Distinguish it from design patterns.

**Design different UML design documents to show logical view by applying Object Oriented Design (OOD) in various software domains.(CO5)**

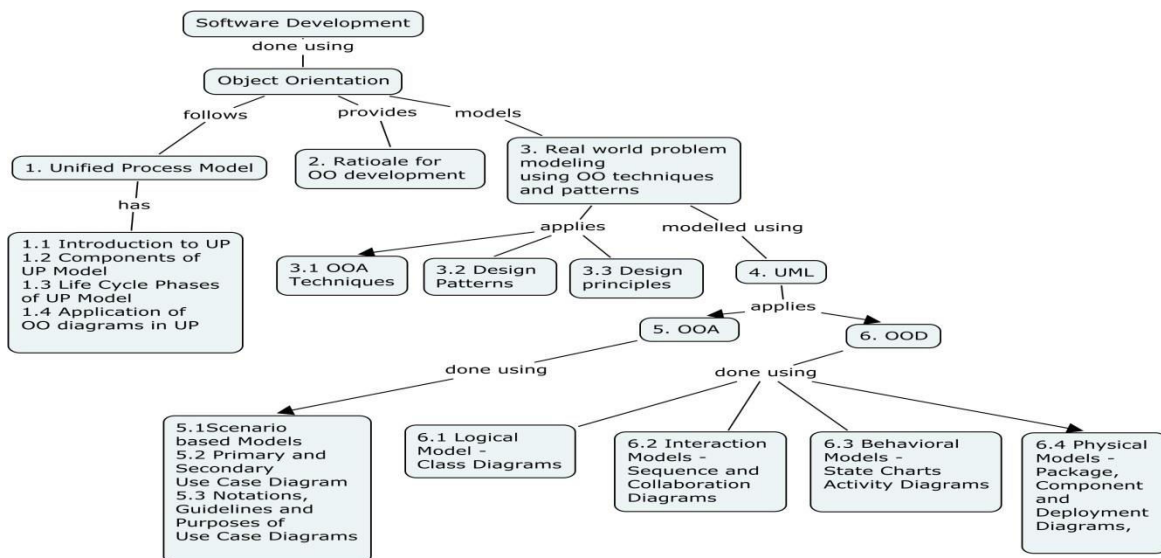
1. What is the difference between use case diagram and use cases?
2. What are the standard class stereotypes available in UML 2.0?
3. List the purposes of sequence diagrams.
4. How object creation, deletion, life line of an object and life time of a method are represented in a sequence diagram? Draw its notations.
5. What are the types of links supported by collaboration diagrams?

## Develop different design documents to show physical view using UML for any given application. (CO5)

Consider a ‘Transport Management System’. This system proposes an efficient method for handling the transport operations. It provides an efficient record maintenance system. The main feature is it provides automatic intimation of expired date of driving license, complaints and suggestions to the RTO office, fix appointment for LLR test and get bus information. This information will maintain details of all employees, contacts, bus details etc. and provides an option for reports generation.

1. Draw a package diagram by grouping the classes identified from the above application.
2. Draw a component diagram to show the components interaction for the above system.
3. How a deployment diagram shows the layered architecture implementation of the system given above.

### Concept Map



### Syllabus

**Unified Process in Object Oriented Software Development** - Basics of Software Development Process, Introduction to UP, Components of Unified Process Model, Life Cycle Phases of Unified Process Model, Application of OO diagrams in UP, Rationale for OO development - Object Orientation in Software development process, Flavors of Object Orientation, Basic Entities and Constructs of Object Orientation, Structured Approach Vs. Object Orientated Approach, **Modelling the real world problems using OO techniques and Design Patterns** - Object Oriented Analysis (OOA) of problem domain, OOA Techniques for Objects Identification, Object Oriented Design (OOD) of problem domain, Design Principles in Class Design, Design Patterns in Classes and Objects Identification and Refinement, Modeling with UML - Problem Domain Understanding, Traditional Analysis Methods and Models, Characteristics of Good Analysis, Deficiency with the traditional approaches,

**UML** – Introduction, UML diagrams for OOA and OOD, Object Oriented Analysis using UML- Scenario based Models - Use Case Analysis - Primary and Secondary Use Case Diagram - Notations, Guidelines, Purposes of Use Case Diagrams, Object Oriented Design using UML-

Logical Model -UML Class Diagram basic and advanced concepts - Interaction Models – Sequence and Collaboration Diagrams, Behavioral Models, State Charts and Activity Diagrams, Physical Models – Package, Component and Deployment Diagrams, Case Study – Tutorial

### Reference Books

1. Grady Booch, Robert A.Maksmichuk, Michael W.Engle, Bobbi J.Young, Jim Conallen, Kelli A. Houston, “Object-oriented analysis and design with applications”, Third edition, Pearson Education, 2011.
2. Martin Fowler, “UML Distilled”, Third edition, Pearson Education, 2011.
3. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, “Design Patterns: Elements of Reusable Object-Oriented Software”, First Edition, Addison-Wesley Professional, 2010 (Reprint).
4. Ali Bahrami, “Object-oriented system development”, First Edition, Tata McGraw Hill, 2010(Reprint).

### Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures
1	<b>Unified Process in Object Oriented Software Development</b>	
1.1	Basics of Software Development Process	1
1.2	Introduction to UP	1
1.3	Components of Unified Process Model	1
1.4	Life Cycle Phases of Unified Process Model	1
1.5	Application of OO diagrams in UP	1
	Tutorial	
2	<b>Rationale for OO development</b>	
2.1	Object Orientation in Software development process	1
2.2	Flavours of Object Orientation	2
2.3	Basic Entities and Constructs of Object Orientation	1
2.4	Structured Approach Vs. Object Orientated Approach	1
	Tutorial	
3	<b>Modelling the real world problems using OO techniques and Design Patterns</b>	
3.1	OOA Techniques for Classes and Objects Identification	2
	Tutorial	2
3.2	OOD Techniques - Design Principles in Class Design	2

Module No.	Topic	No. of Lectures
3.3	OOD Techniques - Design Patterns in Classes and Objects Identification and Refinement	2
	Tutorial	
4	<b>Modelling with UML</b>	
4.1	Problem Domain Understanding	1
4.2	Traditional Analysis Methods and Models	2
4.3	Characteristics of Good Analysis	1
4.4	Deficiency with the traditional approaches	1
4.5	UML - Introduction	1
4.6	UML diagrams for OOA and OOD	1
5	<b>Object Oriented Analysis</b>	
5.1	Scenario based Models - Use Case Analysis	1
5.2	Primary and Secondary Use Case Diagram	2
5.3	Notations, Guidelines, Purposes of Use Case Diagrams	2
	Tutorial	
6	<b>Object Oriented Design</b>	
6.1	Logical Model -UML Class Diagram basic and advanced concepts	2
6.2	Interaction Models – Sequence and Collaboration Diagrams	2
6.3	Behavioral Models – State Charts and Activity Diagrams	1
6.4	Physical Models – Package, Component and Deployment Diagrams	1
6.5	Case Study - Tutorial	
	Total	36

**Course Designer:**

Prof. N. Murali

nmcse@tce.edu

						Category	L	T	P	Credit
<b>20CA370</b>	<b>JAVA PROGRAMMING LABORATORY</b>					PC	0	0	2	2

### Preamble

To enable the students practice the concepts of java programming language and develop solutions for real world problems.

### Prerequisite

None

### Course Outcomes

On the successful completion of the course, students will be able to

CO1: Understand the enabling technologies for building internet applications. Understand

CO2: Write Java programs for techniques and features of the networking and remote method development to Construct a internet application. Apply

CO3: Implement packages, access specifiers and interfaces in a program Apply

CO4: Implement Program for Events and interactivity using Layout Manager. Apply

CO5: Generate program for network chatting Analyze

CO6: Write technical report on the observations from the experiments Apply

### Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M	S	S	S	-	-	-	-	-	-
CO2	S	M	S	S	M	M	-	-	-	-	-	-
CO3	S	S	M	S	S	S	-	-	-	-	-	-
CO4	S	S	M	S	S	M	-	-	-	-	-	-
CO5	S	S	S	S	S	M	-	-	-	-	-	-
CO6	S	S	S	S	S	M	-	-	-	-	-	-

S- Strong; M-Medium; L-Low



**List of Experiments**

Develop Java Programs for

1. Use of Objects
2. Using classes and inheritance
3. JNI concepts
4. Multithread applications
5. Exception handling
6. Implementing packages, access specifiers and interfaces
7. Streams
8. JDBC program using different statements
9. Applet program for Animation text, images and sounds
10. Events and interactivity using Layout Manager.
11. Socket program for network chatting
12. Client server application using RMI techniques

**Mini Project :**

The course instructor shall provide real time problems/specifications to the students for mini project. The project shall be completed before the commencement of 4<sup>th</sup> semester and a report shall be submitted.

**References:**

1. Herbert Schildt, "Java the Complete Reference", 9th Edition, McGraw Hill, 2014.
2. Margaret Levine Young, "The Internet - Complete Reference" , 2nd Edition, Tata McGraw Hill, 2002, (Reprint 2016).
3. Paul Deitel, Harvey Deitel, Abbey Deitel, "Internet and WWW How to Program", 5th Edition, Tata McGraw Hill, 2011.

**Course Designers:**

Dr. D. Anitha

[anithad@tce.edu](mailto:anithad@tce.edu)

<b>20CA380</b>	<b>APPLICATIONS LABORATORY</b>	<b>DEVELOPMENT</b>	Category	L	T	P	Credit
			PC	0	1	2	3

### Preamble

This course emphasizes on developing automated applications that increases data access, efficiency and visibility across the organization to achieve the goal of an organization.

### Prerequisite

- 20CA180: RDBMS Laboratory
- 20CA210 : Software Engineering

### Course Outcomes

On the successful completion of the course, students will be able to:

CO1:	Design a consistent enterprise application across all delivery projects in an Organisation.	Apply
CO2:	Provide reduction in costs and effort for the delivery projects.	Apply
CO3:	Achieve the business strategy, vision and target operating model.	Apply
CO4:	Communicate the project to the stakeholders.	Apply
CO5:	Provide the flexibility to include new ideas in the future.	Apply
CO6:	Develop quality Mobile Application by means of integration of various developed components.	Apply

### Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	S	M	M	-	S	-	-	-	-	-
CO2	S	S	S	S	L	-	S	-	-	-	-	-
CO3	S	S	S	S	L	-	S	-	-	-	-	-
CO4	S	S	S	-	-	-	S	-	-	-	-	-
CO5	S	S	S	S	S	-	S	-	-	-	-	-
CO6	S	S	S	M	L	-	S	-	-	-	-	-

S- Strong; M-Medium; L-Low

### Lab Experiments

a. Development of Enterprise Application using J2EE (Java 2 Enterprise Edition)/ .Net / Open Source Based development -Based on the discretion of the course faculty)

1. Application of Architectural Frameworks & Design Patterns
2. Client Tier development
3. Web Tier Development
4. Business Tier Development
5. System Integration With Enterprise Information System Tier
6. Packaging and Deployment

For any of the following enterprise applications:

1. HR Management
  - a. Recruitment System
  - b. Staff Appraisal System
2. Finance Management
  - a. Billing
  - b. Payroll Processing
3. Stock Management
  - a. Sales and Purchase
  - b. Marketing of Products
4. Production Management (Increase the QTY and QLTY)
  - a. Cost Management
  - b. Quality Assurance and Delivery of Finished Products



#### Reference Books & web resources

1. Martin Fowler, Patterns of Enterprise Application Architecture (Addison Wesley Signature Series), 1<sup>st</sup> Edition, 2010 (Reprint)
2. Inderjeet Singh, Beth Stearns, Mark Johnson, and the Enterprise Team, Designing Enterprise Applications with the J2EE™ Platform, Second Edition, Addison Wesley, 2011 (Reprint)
3. John Kanakalis, "Developing .NET Enterprise Applications, 1st Edition, Apress, 2003.
4. Yakov Fain, "Enterprise Development with Flex", O'Reilly Series, 1<sup>st</sup> Edition, 2010
5. vo Jansch, "PHP/Architect's Guide to Enterprise PHP Development", Musketeers.me, LLC, 2008
6. Steven Holzner, PHP: The Complete Reference, McGraw Hill Education; 1 edition (30 November 2007).
7. <http://docs.oracle.com/javaee/6/firstcup/doc/gcrky.html>
8. <http://www.oracle.com/technetwork/developer-tools/jdev/j2eedev-084379.html>
9. <http://j2eetutorials.50webs.com/>
10. <http://www.webagesolutions.com/knowledgebase/waskb/waskb017/>

#### Course Designers:

Dr. T. Chandrakumar

[tckcse@tce.edu](mailto:tckcse@tce.edu)

<b>20CA390</b>	<b>DATA WAREHOUSING AND DATA MINING LABORATORY</b>	Category	L	T	P	Credit
		PC	0	0	2	2

**Preamble**

In this laboratory, students will implement the various Data Warehousing and Data Mining concepts using Oracle and WEKA / R tool

**Prerequisite**

- 20CA180 : RDBMS Laboratory
- 20CA250: Data Warehousing and Data Mining

**Course Outcomes**

On the successful completion of the course, students will be able to

CO1: Develop various real time applications using data mining techniques	Understand
CO2: Test the developed code using VB.net and Weka/R tool	Apply
CO3: Apply text mining on the data warehouse	Apply
CO4: Perform multi-dimensional data model using Oracle	Analyze
CO5: Develop a program using a R Tool to solve a association rule	Apply
CO6: Develop a program to perform clustering and Classification using various algorithms.	Analyze

**Mapping with Programme Outcomes**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M	S	-	M	-	-	-	-	-	-
CO2	M	M	M	M	M	-	-	-	-	-	-	-
CO3	M	S	S	M	M	L	-	-	-	-	-	-
CO4	L	S	M	M	S	L	-	-	-	-	-	-
CO5	S	L	S	S	M	L	-	-	-	-	-	-
CO6	M	M	S	M	M	L	-	-	-	-	-	-

S- Strong; M-Medium; L-Low

### List of Experiments

Perform the given experiments:

1. Execute Queries and PL/SQL
2. Multi-dimensional data model using SQL queries. E.g. Star, snowflake and Fact constellation schemas
3. OLAP operations such slice, dice, roll up, drill up, pivot etc.
4. Text mining on the given data warehouse
5. Correlationship analysis between for the given data set
6. Attribute relevance analysis on the given data
7. Information gain for a particular attribute in the given data
8. Data pre-processing for data mining in Weka/R tool
9. Clustering in Weka/R tool.
10. Association rule analysis in Weka / R tool

#### 11. Mini Project:

The course instructor shall provide real time problems / specifications to the students for mini project. The project shall be completed before the Commencement of 4<sup>th</sup> semester and a report shall be submitted.

#### Data Mining for Financial Data Analysis

Design and construction of data warehouses for multidimensional data analysis and data mining .View the debt and revenue changes by month, by region, by sector, and by other factors, Access statistical information such as max, min, total, average, trend, etc.

#### Course Designers:

Dr. S. Vijayalakshmi

[svlcse@tce.edu](mailto:svlcse@tce.edu)

**20CAPA0****ELECTRONIC COMMERCE AND  
ELECTRONIC BUSINESS**

Category	L	T	P	Credit
PE	3	0	0	3

**Preamble**

To enable the students to gain knowledge in information systems such as Electronic Commerce and provide a customer specific software solution.

**Prerequisite**

- None

**Course Outcomes**

On the successful completion of the course, students will be able to

CO1:	Identify current management issues associated with electronic commerce strategies	Understand
CO2:	Analyze the Internet's role in the decision process	Analyze
CO3:	Demonstrate the functionality of Electronic Markets and different threats	Analyze
CO4:	Develop Business models for Intranet and Extranet e-commerce	Apply
CO5:	Explain the electronic payment systems	Understand
CO6:	Summarize public policy, Taxation, Infrastructure for E-Commerce	Analyze

**Mapping with Programme Outcomes**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1			M						M			
CO2				L		M		M				
CO3	M				S							
CO4				M								
CO5	S		L				S					S
CO6						L				S		

S- Strong; M-Medium; L-Low

## Assessment Pattern

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	20	20	20	10
Understand	20	20	20	20
Apply	40	40	40	40
Analyse	20	20	20	30
Evaluate	0	0	0	0
Create	0	0	0	0

## Course Level Assessment Questions

### Identify current management issues associated with electronic commerce strategies (CO1):

1. Define E- Commerce.
2. State the definition of EDI.
3. What is Credit Transaction Trade Cycle?
4. State any two advantages of Electronic Markets.
5. What is Business to Consumer electronic commerce?

### Analyze the Internet's role in the decision process (CO2):

1. Explain the following terms:
  - (i) Supply Chain
  - (ii) Value Chain
  - (iii) Competitive Advantage
  - (iv) Business Strategy
2. Explain in detail Electronic Data Interchange and how it is useful in implementation of E-Commerce.
3. Explain Business to Business E-Commerce in detail with one example.
4. State the advantages and disadvantages of Electronic market.
5. Explain different kinds of Threats in E-Security

### Demonstrate the functionality of Electronic Markets and different threats (CO3):

1. Explain the Web Site Evaluation Model in detail and how do you apply it.
2. Explain in detail E- Security and how it will apply for an e-business.
4. Explain the following Concepts with one example: Internet Book Shop, Virtual Auction, Online Share Dealing and Electronic Newspaper
6. Explain in detail Strategy formulation, Implementation Planning, Implementation and evaluation of E-Commerce.

### Develop Business models for Intranet and Extranet (CO4):

1. Show the Architecture of Intranet, Internet and Extranet.
2. Give some applications of Internet.
3. Explain the basic concepts of Extranets.
4. How do you connect business customers to sell parts? Give a case study.
5. Develop a business model for tracking shipping status.

### Explain the electronic payment systems (CO5):

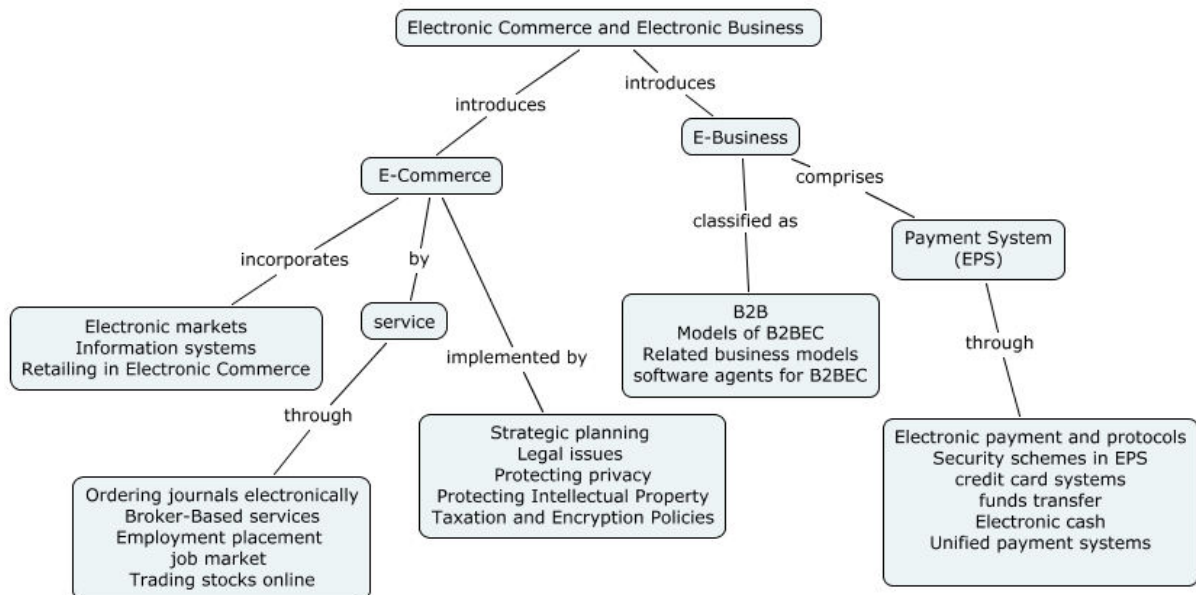
1. Elucidate secure electronic Transaction protocol for credit card payment.

2. Electronic Fund Transfer. Show the merits and Demerits.
3. Show the Security Schemes in Electronic Payment systems.
4. Who are all the Players in Credit card systems?
5. Differentiate Debit and Credit Card.
6. Define Smart card.

### Summarize public policy, Taxation, Infrastructure for EC (CO6):

1. What are the ethical issues in EC?
2. How is Private information collected?
3. Say something about FTC audit.
4. Define Gambling.
5. Show the uses of TCP / IP Protocol.
6. What is the purpose of using HTTP?
7. Define Webcasting.

### Concept Map



### Syllabus

**Foundations of Electronic Commerce (EC)** - The EC field-Electronic markets - Information systems - Benefits and limitation of EC - Driving forces of Electronic Commerce - Impact of EC. Retailing in Electronic Commerce – Overview - Forecast of the B2C Electronic markets. **Electronic Commerce for service industries**- Ordering journals electronically-services-Employment placement and job market-Trading stocks online-Cyber banking and personal finance-Electronic Auctions. **Business-to-Business Electronic Commerce (B2BEC)**- Models of B2BEC-Traditional EDI-software agents for B2BEC-solutions of B2BEC-Managerial issues. Intranet and Extranet.

**Electronic Payment Systems (EPS)**-Electronic payment and protocols-Security schemes in EPS-Electronic credit card systems-Electronic funds transfer-Electronic cash-Unified payment systems-Prospects of EPS – Case studies. **Electronic Commerce strategy and implementation**-Strategic planning for Electronic Commerce-Competitive intelligence on the internet. Legal issues to Privacy in Electronic Commerce (EC)-Internet protocols-Client/Server technology-Internet Security-Selling on the web-Multimedia delivery-Webcasting-Challenges and Opportunities – Case studies.



### Reference Books

1. "Electronic Commerce- A Managerial Perspective", Efraim Turban, Jae Lee and David King, Person Education, 2010. Chapters 1, 2, 5, 6, 7, 8, 9, 10 and 11.
2. "Electronic Commerce: A Managers Guide", Ravi Kalakotta and Andrew B. Whinston, Person Education, 2009.
3. "E-Business and IS Solutions: An Architectural Approach to business Problems and Opportunities", William J. Buffan, Person Education, 2009.

### Course Contents and Lecture Schedule

Module. No	Topics	No. of Lectures
1	<b>Foundations of Electronic Commerce</b>	
1.1	The EC field, Electronic markets	2
1.2	Information systems, Benefits and limitation of EC	1
1.3	Driving forces of Electronic Commerce, Impact of EC.	1
1.4	Retailing in Electronic Commerce	1
1.5	Overview-Forecast of the B2C Electronic markets, Business models of Electronic marketing	1
1.6	Online customer service, Procedure for internet shopping	1
1.7	Aiding comparison shopping, Impact of EC on traditional retailing systems.	1
2	<b>Electronic Commerce for service industries</b>	
2.1	Ordering journals electronically, Broker-Based services	1
2.2	Travel and tourism services, Employment placement	1
2.3	Job market , Trading stocks online	1
2.4	Cyber banking and personal finance	1
2.5	Electronic Auctions-Types of Auctions-Benefits and limitations	1
2.6	Business to business Auction-Managerial issues.	1
2.7	Case Studies	1
3	<b>Business-to-Business Electronic Commerce (B2BEC)</b>	
3.1	Overview, Characteristics of B2BEC	2
3.2	Models of B2BEC, Traditional EDI-Internet based EDI-Roll of software agents for B2BEC Electronic marketing in B2BEC-Solutions of B2BEC-Managerial issues.	2
3.3	Intranet and Extranet-Architecture of the Internet	1

3.4	Intranet, Extranet-Applications, Related business models.	1
3.5	Mobile Commerce in B2B and B2C	1
3.6	Technical aspects of M-Commerce and Case Studies	1
3.6	<b>Electronic Payment Systems (EPS)-Overview</b>	
3.7	Electronic payment and protocols, Security schemes in EPS-Authentication, Authorization and Access Rights	1
3.8	Electronic credit card systems, Electronic funds transfer, Prospects of EPS	1
3.9	Case Studies	2
4	<b>Electronic Commerce strategy and implementation-</b>	
4.1	Electronic Business's strategy, Strategic planning for Electronic Commerce	1
4.2	Competitive intelligence on the internet, Legal issues to Privacy in Electronic Commerce (EC)	1
4.3	Ethical issues-Protecting privacy, Protecting Intellectual Property-	1
4.4	Taxation and Encryption Policies, Consumer and Seller protection in EC.	1
4.5	Infrastructure for EC-Internet protocols, Client/Server technology-	1
4.6	Internet Security, Selling on the web-Multimedia delivery	1
4.7	Webcasting, Challenges and Opportunities	1
4.8	Case Studies	2
	Total	36

**Course Designer:**

Dr. S. Vijayalakshmi

svlcse@tce.edu

**20CAPB0****BUSINESS ANALYTICS**

Category	L	T	P	Credit
PE	3	0	0	3

**Preamble**

The course is designed to gain an understanding of how managers use business analytics to formulate and solve business problems and to support managerial decision making. The course familiarizes the students with the processes needed to develop, report, and analyze business data.

**Prerequisite**

None

**Course Outcomes**

On the successful completion of the course, students will be able to

- |  |            |
|--|------------|
| CO1: Explain the evolution of business analytics, business intelligence, operations research and management science, and decision support systems. | Understand |
| CO2: Apply relative and absolute addressing in Excel formulas.   | Apply      |
| CO3: Compute the range, interquartile range, variance, and standard deviation of a set of data.  | Apply      |
| CO4: Explain the purpose and procedures of hypothesis testing, ANOVA tool.   | Analyze    |
| CO5: Explain the purpose of regression analysis and provide examples in business.  | Apply      |
| CO6: Apply the four-step process to develop a mathematical model for an optimization problem.  | Analyze    |

**Mapping with Programme Outcomes**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S			M		L						
CO2		S	M			S						
CO3				S		L		M				
CO4	S		L		S				S			
CO5	M		S			L						
CO6		S		L			M	S				

Strong; M-Medium; L-Low

**Assessment Pattern**

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	20	20	20	20
Understand	20	20	20	20
Apply	40	40	20	30
Analyse	20	20	40	30
Evaluate	0	0	0	0
Create	0	0	0	0

**Course Level Assessment Questions**

**Explain the evolution of business analytics, business intelligence, operations research and management science, and decision support systems. (CO1):**

1. Discuss how business analytics can be used in sports, such as tennis, cricket, football, and so on. Identify as many opportunities as you can for each.
2. Suggest some metrics that a hotel might want to collect about their guests. How might these metrics be used with business analytics to support decisions at the hotel?
3. A bank developed a model for predicting the average checking and savings account balance as  $\text{Balance} = -17,732 + 367 * \text{age} + 1,300 * \text{years education} + 0.116 * \text{household wealth}$ .
  - a. Explain how to interpret the numbers in this model.
  - b. Suppose that a customer is 32 years old, is a college graduate (so that years education = 16), and has a household wealth of \$150,000. What is the predicted bank balance?
4. A survey handed out to individuals at a major shopping mall in a small Florida city in July asked the following:
  - gender
  - age
  - ethnicity
  - length of residency
  - overall satisfaction with city services (using a scale of 1–5, going from poor to excellent)
  - quality of schools (using a scale of 1–5, going from poor to excellent)

What types of data (categorical, ordinal, interval, or ratio) would each of the survey items represent and why?

5. A firm installs 1500 air conditioners which need to be serviced every six months. The firm can hire a team from its logistics department at a fixed cost of \$6,000. Each unit will be serviced by the team at \$15.00. The firm can also outsource this at a cost of \$17.00 inclusive of all charges.

- a. For the given number of units, compute the total cost of servicing for both options. Which is a better decision?
- b. Find the break-even volume and characterize the range of volumes for which it is

more economical to outsource.

**Apply relative and absolute addressing in Excel formulas. (CO2):**

1. The Excel file *Firm Data* shows the prices charged and different product sizes. Prepare a worksheet using VLOOKUP function that will compute the invoice to be sent to a customer when any product type, size, and order quantity are entered.
2. The Excel file *Store and Regional Sales Database* provides sales data for computers and peripherals showing the store identification number, sales region, item number, item description, unit price, units sold, and month when the sales were made during the fourth quarter of last year. Modify the spreadsheet to calculate the total sales revenue for each of the eight stores as well as each of the three sales regions.
3. The following exercises use the *Purchase Orders* database. Use MATCH and/or INDEX functions to find the following:
  - a. The row numbers corresponding to the first and last instance of item number 1369 in column C (be sure column C is sorted by order number).
  - b. The order cost associated with the first instance of item 1369 that you identified in part (a).
  - c. The total cost of all orders for item 1369. Use the answers to parts (a) and (b) along with the SUM function to do this. In other words, you should use the appropriate INDEX and MATCH functions within the SUM function to find the answer.

Validate your results by applying the SUM function directly to the data in column G.

4. Use INDEX and MATCH functions to fill in a table that extracts the amounts shipped between each pair of cities in the Excel file *General Appliance Corporation*. Your table should display as follows, and the formula for the amount should reference the names in the From and To columns:

From	To	Amount
Marietta	Cleveland	0
Marietta	Baltimore	350
Marietta	Chicago	0
Marietta	Phoenix	850
Minneapolis	Cleveland	150
Minneapolis	Baltimore	0
Minneapolis	Chicago	500
Minneapolis	Phoenix	150

**Compute the range, interquartile range, variance, and standard deviation of a set of data (CO3):**

1. In the Excel file Facebook Survey, find the average and median hours online/week and number of friends in the sample using the appropriate Excel functions. Compute the midrange and compare all measures of location.
2. Define Arithmetic Mean. Give example.
3. Show the difference between median and mode. Illustrate with example.
4. Explain the measures of Dispersion.
5. How do you apply the Coefficient of Variation?

6. How Empirical Rules are used to Measure the Capability of a Manufacturing Process.
7. Give the formula for finding Co-variance.
8. Computing the Correlation Coefficient. Discuss in detail.

**Explain the purpose and procedures of hypothesis testing, ANOVA tool. (CO4):**

1. Show the Hypothesis-Testing Procedure with example.
2. Exhibit one-sample hypothesis test.
3. How  $\beta$  depends on the True Population Mean.
4. Illustrate a Two-Tailed Hypothesis Test for the Mean.
5. The manager of a store claims that 60% of the shoppers entering the store leave without making a purchase. Out of a sample of 50, it is found that 35 shoppers left without buying. Is the result consistent with the claim?
6. The director of human resources for a large bank has compiled data on about 70 former employees at one of the bank's call centers. For each of the following, assume equal variances of the two populations.
  - a. Test the null hypothesis that the average length of service for males is the same as for females.
  - b. Test the null hypothesis that the average length of service for individuals without prior call center experience is the same as those with experience.
  - c. Test the null hypothesis that the average length of service for individuals with a college degree is the same as for individuals without a college degree.
  - d. Now conduct tests of hypotheses for equality of variances. Were your assumptions of equal variances valid? If not, repeat the test(s) for means using the unequal variance test.
7. A survey of college students determined the preference for cell phone providers. The following data were obtained:

Gender	Provider			
	T-Mobile	AT&T	Verizon	Other
Male	12	39	27	16
Female	8	22	24	12

Can we conclude that gender and cell phone provider are independent? If not, what implications does this have for marketing?

8. A college is trying to determine if there is a significant difference in the mean GMAT score of students from different undergraduate backgrounds who apply to the MBA program. The data Scores contain data from a sample of students. What conclusion can be reached using ANOVA?

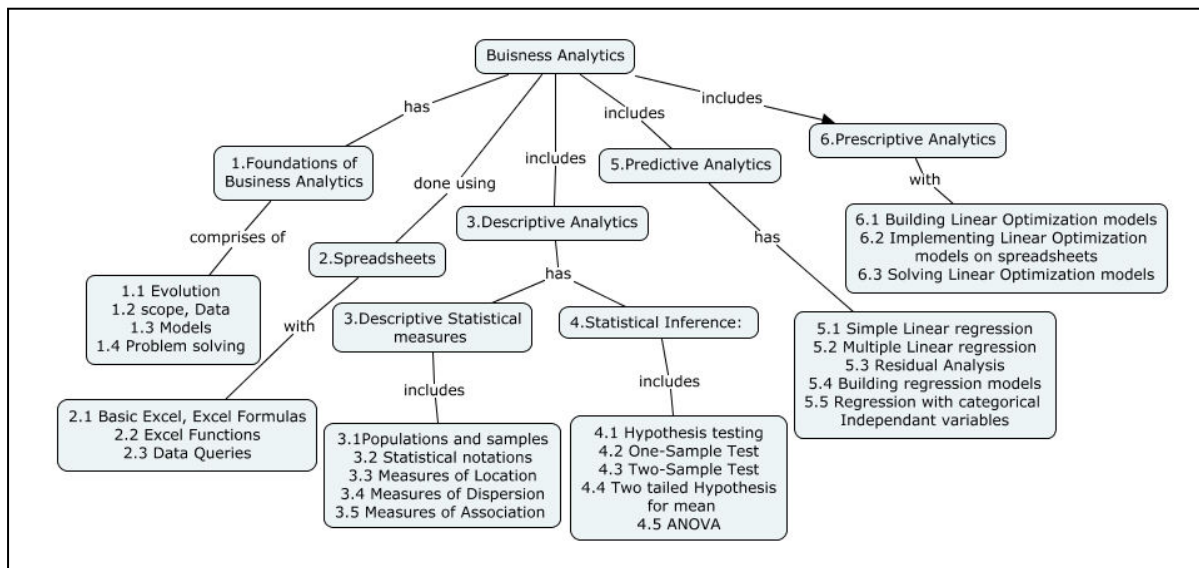
**Explain the purpose of regression analysis and provide examples in business (CO5):**

1. Show the Common types of mathematical functions used in predictive analytical models
2. Compute Using Excel to Find the Best Regression Line.
3. Outline the features of least-squares regression with example.
4. What is coefficient of determination?
5. Building Good Regression Models. Illustrate.

### Apply the four-step process to develop a mathematical model for an optimization problem. (CO6):

1. Apply the four-step process to develop a mathematical model for an optimization problem.
2. Show the Characteristics of Linear Optimization Models.
3. How do you Implement Linear Optimization Models on Spreadsheets.
4. Define feasible solution.
5. Illustrate Standard Solver for the SSC Problem.

### Concept Map



### Syllabus

**Foundations of Business Analytics:** Evolution of Business analytics, scope, Data for Business Analytics, Models in Business Analytics, problem solving with business analytics. **Analytics on Spreadsheets:** Basic Excel, Excel Formulas, Excel Functions, Data Queries. **Descriptive Analytics: Descriptive Statistical measures** - Populations and samples, Statistical notations, Measures of Location, Measures of Dispersion, and Measures of Association. **Statistical Inference:** Hypothesis testing, one-Sample Test, Two-Sample Test, Two-tailed Hypothesis for mean, ANOVA. **Predictive Analytics:** Simple Linear regression, Multiple Linear regression, Residual Analysis, Building regression models, Regression with categorical Independent variables. **Prescriptive Analytics:** Building Linear Optimization models, Implementing Linear Optimization models on spreadsheets, Solving Linear Optimization models.

### Reference Books

1. James Evan, Business Analytics- Methods, Models, and Decisions (2<sup>nd</sup> Edition), Pearson, 2013. (Chapters: 1, 2, 4, 7, 8, 13)
2. Gert H. N. Laursen, Business Analytics for Managers: Taking Business Intelligence Beyond Reporting, Wiley (2<sup>nd</sup> Edition), 2010.
3. S. Christian Albright and Wayne L. Winston, Analytics: Data Analysis and Decision Making, Sixth Edition, 2014.

**Course Contents and Lecture Schedule**

<b>Module No.</b>	<b>Topic</b>	<b>No. of Periods</b>
<b>1</b>	<b>Foundations of Business Analytics:</b>	
1.1	Evolution of Business analytics	1
1.2	scope, Data for Business Analytics	2
1.3	Models in Business Analytics	2
1.4	Problem solving with business analytics	2
<b>2</b>	<b>Analytics on Spreadsheets:</b>	
2.1	Basic Excel, Excel Formulas	1
2.2	Excel Functions	1
2.3	Data Queries	2
<b>3</b>	<b>Descriptive Analytics</b>	
3.1	Descriptive Statistical measures	2
3.2	Populations and samples	1
3.3	Statistical notations	1
3.4	Measures of Location	2
3.5	Measures of Dispersion	2
3.6	Measures of Association	2
<b>4</b>	<b>Statistical Inference:</b>	
4.1	Hypothesis testing	1
4.2	One-Sample Test	1
4.3	Two-Sample Test	1
4.4	Two tailed Hypothesis for mean	1
4.5	ANOVA	1
<b>5</b>	<b>Predictive Analytics</b>	
5.1	Simple Linear regression	1
5.2	Multiple Linear regression	1
5.3	Residual Analysis	1
5.4	Building regression models	1
5.5	Regression with categorical Independent variables	1



Module No.	Topic	No. of Periods
6	<b>Prescriptive Analytics</b>	
6.1	Building Linear Optimization models	1
6.2	Implementing Linear Optimization models on spreadsheets	2
6.3	Solving Linear Optimization models	2
		36

**Course Designer:**

Dr.T.Chandrakumar

[tckcse@tce.edu](mailto:tckcse@tce.edu)

<b>20CAPC0</b>	<b>SOFTWARE PROJECT MANAGEMENT</b>	Category	L	T	P	Credit
		PE	3	0	0	3

**Preamble**

To provide a sound understanding of the software project management concepts. Also, to help the students understand the challenges and issues in software projects from project managers perspectives. To learn as how to incorporate leadership and management qualities in software product development.

**Prerequisite**

- 20CA210 : Software Engineering

**Course Outcomes**

On the successful completion of the course, students will be able to:

CO1:	Identify suitable software process model for software projects.	Understand
CO2:	Differentiate different software product development techniques.	Understand
CO3:	Apply appropriate software cost estimation technique for a given project.	Apply
CO4:	Apply software project management principles for a software project.	Apply
CO5:	Develop software metrics for measuring and managing software processes.	Analyze
CO6:	To assess the software product for quality standards.	Evaluate

**Mapping with Programme Outcomes**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	S	M	S	-	S	M	L	-	-	L	-
CO2	M	M	-	S	S	-	S	L	-	-	L	-
CO3	M	S	S	M	M	L	S	M	-	-	L	-
CO4	M	S	M	S	S	L	M	S	L	L	M	L
CO5	S	L	L	S	M	M	S	S	L	L	S	L
CO6	S	M	L	S	M	M	M	M	-	-	S	-

S- Strong; M-Medium; L-Low

**Assessment Pattern**

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	25	20	10	10
Understand	25	20	20	10
Apply	40	30	30	30
Analyse	10	20	20	30
Evaluate	0	10	20	20
Create	0	0	0	0

### Course Level Assessment Questions

#### Identify suitable software process model for software projects (CO1):

1. Define the term: software process.
2. List the various software process models.
3. Mention the challenges and issues in software projects.
4. Outline the activities in software project planning.
5. How will you classify software projects?

#### Differentiate different software product development techniques (CO2):

1. How will you choose a software process model for a given project?
2. Distinguish between traditional and agile software development techniques.
3. How will measure the impact of formal verification and validation techniques?
4. What are the top 10 software management principles in modern project profile?
5. Outline the different levels of software process.

#### Apply appropriate software cost estimation technique for a given project (CO3):

1. How will you select a suitable cost estimation model for a given project?
2. Is it possible to measure the accuracy of software cost estimation?
3. How will you adjust software cost estimation drivers/parameters according to schedule?
4. Mention the risks associated with software cost estimation models.
5. How will you perform cost/benefit analysis for different cost models?

#### Apply software project management principles for a software project (CO4):

1. Analyze software project planning activities for a project with a neat diagram.
2. Elaborate the various software project management principles.
3. How is software project management viewed from a managerial perspective and a technical perspective?
4. How are resources optimization handled by project managers?
5. Discuss the various activities at the time of project completion.

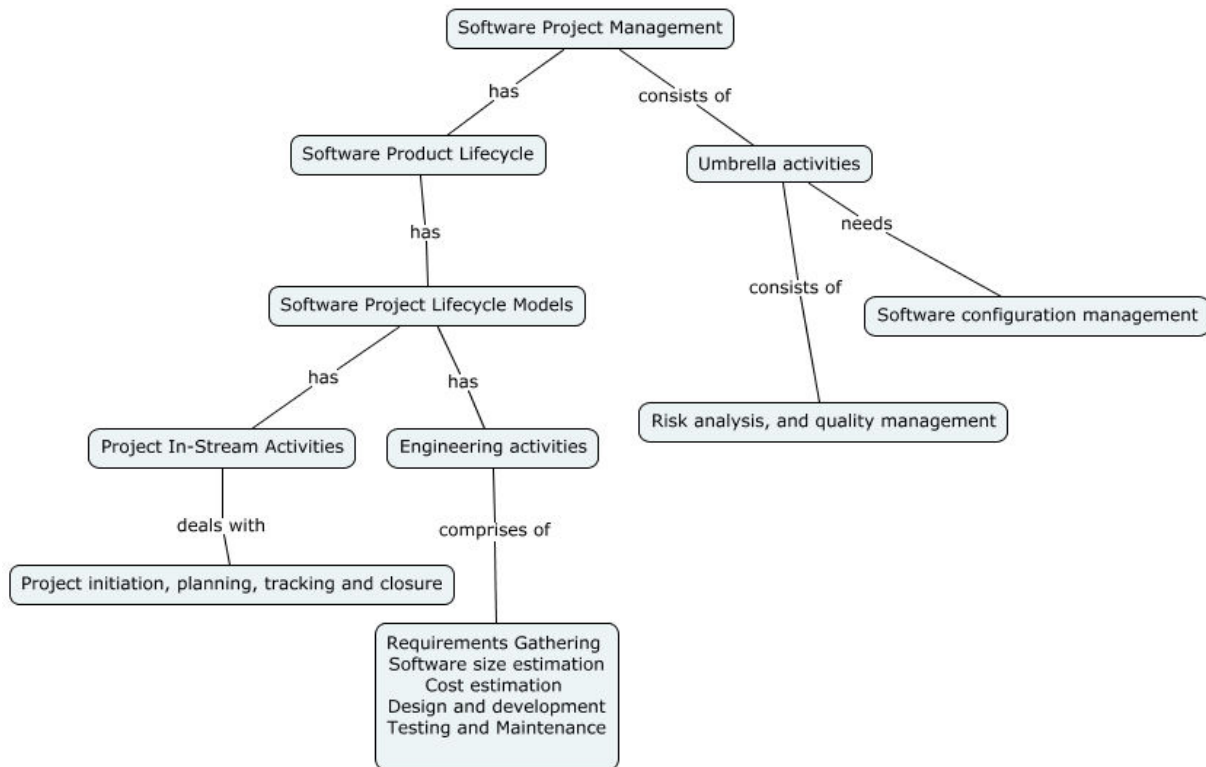
#### Develop software metrics for measuring and managing software processes (CO5):

1. Present various direct and indirect software metrics used in a software project.
2. How will you apply and analyze software processes using metrics?
3. How will you manage software risks using metrics?
4. Discuss the various quality management metrics.
5. Evaluate the cost of quality management in a software project.

#### To assess the software product for quality standards (CO6):

1. Present the various software quality attributes for a software project.
2. Why do you require Quality Function Deployment (QFD)?
3. How will you apply and analyze quality standards such as CMM or TQM for a software project?
4. Describe the features of various quality models and its significance to project management.
5. How will you evaluate the performance of one project with another?

## Concept Map



## Syllabus

**Project Management** - An Overview; **Software Product Lifecycle**; **Software Processes**-Process Models-software development-Modern practices and methods; **Software Project Lifecycle Models**; **Umbrella Activities** in software projects- Software metrics-Software configuration management-Software quality assurance -Risk analysis and management; **Project In-Stream Activities**-Project Initiation -Project Planning-Project Tracking-Project Closure; **Engineering Activities**-Requirements Gathering, Analysis and Management-Software size and cost estimation techniques-Design and development-Testing and Maintenance. **Case studies** on software process models, cost estimation and software design and deployment.

## Reference Books

1. Gopalswamy Ramesh, Managing Global Software Projects, Tata McGraw-Hill Publishing Company Ltd., New Delhi, 2012.
2. Walker Royce, "Software Project Management", Tata McGraw-Hill Publishing Company Ltd., New Delhi, 2011.
3. Bob Hughes, Mikecatterell, "Software Project Management", Third Edition, Tata McGraw Hill, 2012.
4. Robert T. Futrell, Donald F. Shefer and Linda I. Shefer, "Quality Software Project Management", Pearson Education, 2010.

**Course Contents and Lecture Schedule**

<b>Module. No</b>	<b>Topics</b>	<b>No. of Lectures</b>
<b>1</b>	<b>Project Management</b>	
1.1	Software project management - Issues and Challenges	2
1.2	Principles of Software Project Management	2
1.3	Role of Project Managers	1
1.4	Defining Project Scope	1
1.5	Software Product Lifecycle	2
1.6	Software Processes	1
1.7	Process Models and software development	1
1.8	Software Project Lifecycle Models	2
<b>2</b>	<b>Umbrella Activities</b>	
2.1	Software metrics	2
2.2	Software configuration management	1
2.3	Software quality assurance	2
2.4	Risk Identification	1
2.5	Risk analysis	1
2.6	Risk management	1
2.5	Case studies	1
<b>3</b>	<b>Project In-Stream Activities</b>	
3.1	Project Initiation	2
3.2	Project Planning	1
3.3	Project Tracking	1
3.4	Project Closure	1
3.5	Case studies	1
<b>4</b>	<b>Engineering Activities</b>	
4.1	Requirements Gathering	2
4.2	Software size and cost estimation techniques	2
4.3	Design and development	1
4.4	Testing	1
4.5	Maintenance	1
4.6	Case studies	2
	<b>Total</b>	<b>36</b>

**Course Designer**

Dr. S. Parthasarathy    spcse@tce.edu

<b>20CAPD0</b>	<b>MACHINE LEARNING</b>	<b>Category</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>Credit</b>
		<b>PE</b>	<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>

**Preamble**

The course aims to provide an understanding of machine learning's role in data-driven modeling, prediction, and decision-making.

**Prerequisite**

17CA230 - Design and Analysis of Algorithms

**Course Outcomes**

On the successful completion of the course, students will be able to

- CO1: Illustrate the steps involved in designing a machine learning algorithm. Understand
- CO2: Construct training and prediction algorithms for classification using decision trees, artificial neural networks and Support Vector Machines. Apply
- CO3: Illustrate various Practical applications of Reinforcement Learning. Understand
- CO4: Compare the available design options and apply supervised and unsupervised learning algorithms to solve complex problems with an understanding of the trade-offs involved. Analyze
- CO5: Design and implement various machine learning algorithms in a range of real-world applications. Analyze

**Mapping with Programme Outcomes**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1		M	M			S					
CO2		L	M		S						
CO3			S					M			
CO4			S		M		M				
CO5			S		M		M				

S- Strong; M-Medium; L-Low

**Assessment Pattern**

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	20	20	20	20
Understand	20	20	20	20
Apply	40	40	30	40
Analyse	20	20	30	20
Evaluate	0	0	0	0
Create	0	0	0	0

### Course Level Assessment Questions

#### Illustrate the steps involved in designing a machine learning algorithm (CO1):

1. What is Machine learning?
2. What are the five popular algorithms of Machine Learning?
3. What are the different Algorithm techniques in Machine Learning?
4. List down various approaches for machine learning?
5. Classify the various types of training experience suitable for learning tasks.

#### Construct training and prediction algorithms for classification using decision trees, artificial neural networks and Support Vector Machines (CO2):

1. What is algorithm independent machine learning?
2. What are the advantages of Naive Bayes?
3. Define Perceptron Training Rule.
4. Explain the two components of Bayesian logic program?
5. Derive the formula that calculates the margin given a set of linearly separable training examples
6. What are neural networks used for?

#### Illustrate various Practical applications of Reinforcement Learning (CO3):

1. What are the two classification methods that SVM (Support Vector Machine) can handle?
2. What is ensemble learning?
3. Assume that you have to build a classifier based on SVM to classify a set of images into one of the given 100 types of different flowers. Analyze and Illustrate the Pros and Cons of each of the following techniques to perform a multi-class classification using SVM: (i) One-against-One (ii) One-against-All (iii) Directed Acyclic Graph (DAG) SV
4. What linear function is used by a SVM for classification? How is an input vector  $x_i$  (instance) assigned to the positive or negative class?
5. Illustrate the constrained minimization problem that defines the SVM learning given a set of linearly separable training examples. What is the outcome of solving the problem?

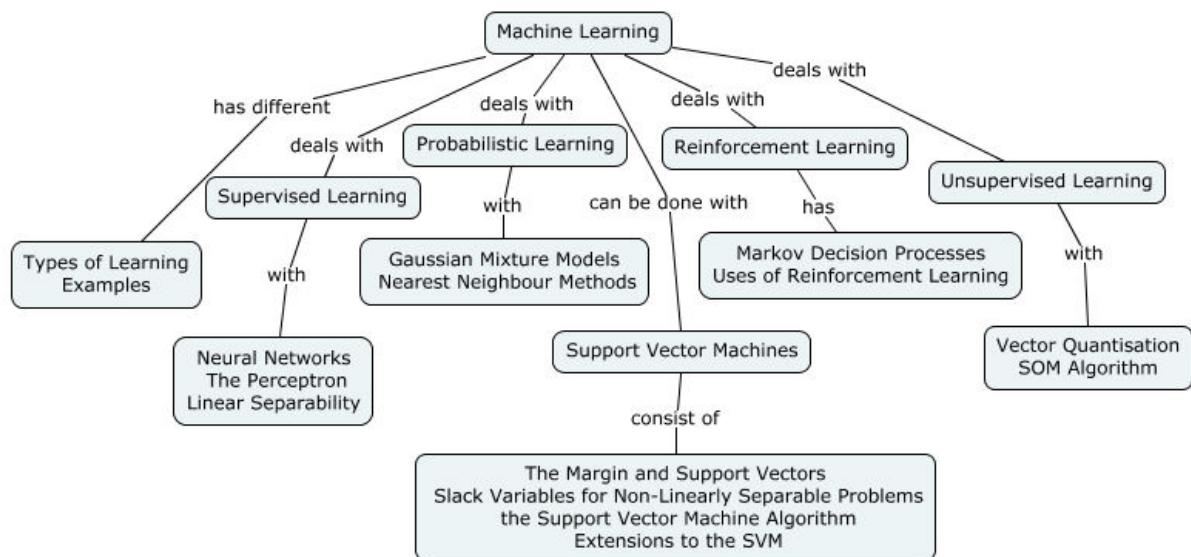
#### Compare the available design options and apply supervised and unsupervised learning algorithms to solve complex problems with an understanding of the trade-offs involved (CO4):

1. What are the different methods for Sequential Supervised Learning?
2. What is the difference between deep learning and reinforcement learning?
3. Is classification supervised learning?
4. Illustrate the different types of supervised learning in machine learning?
5. How does unsupervised learning work?

## Design and implement various machine learning algorithms in a range of real-world applications (CO5):

1. Examine the various strategies to be considered for implementing parallel machine learning algorithm in Hadoop framework.
2. Design and implement a parallel machine learning algorithm on top of the Hadoop framework for a research problem (Assignment)
3. Explain some applications of Machine Learning from Day-to-Day Life.
4. What is the best programming language for machine learning?

### Concept Map



### Syllabus

**Introduction:** Machine Learning, Types of Learning, Examples of Machine Learning Applications, **Supervised Learning:** Neural Networks, The Perceptron, Linear Reparability. **Probabilistic Learning:** Gaussian Mixture Models, Nearest Neighbour Methods. **Support Vector Machines:** The Margin and Support Vectors, Slack Variables for Non-Linearly Separable Problems, the Support Vector Machine Algorithm, Extensions to the SVM, **Reinforcement Learning:** Markov Decision Processes, Uses of Reinforcement Learning, **Unsupervised Learning:** Vector Quantization, the SOM Algorithm.

### Reference Books

1. Tom Mitchell, "Machine Learning", McGraw Hill, Indian edition, 2017.
2. Christopher M. Bishop, "Pattern Recognition and Machine Learning", Springer, 2013.
3. Alpaydin Ethem, "Introduction to Machine Learning", Prentice Hall, 3<sup>rd</sup> edition, 2015.
4. Stephen Marsland, "Machine learning: An algorithmic perspective", CRC, 2009.



### Course Contents and Lecture Schedule

Module .No	Topics	No.of Lectures
1	<b>Introduction:</b> Machine Learning	2
1.1	Types of Learning	1
1.2	Examples of Machine Learning Applications	2
2	<b>Supervised Learning:</b>	1
2.1	Neural Networks,	2
2.2	The Perceptron,	2
2.3	Linear Separability,	2
2.4	The Multi-Layer Perceptron Algorithm.	2
3	<b>Probabilistic Learning:</b>	1
3.1	Gaussian Mixture Models	2
3.2	Nearest Neighbour Methods	2
4	<b>Support Vector Machines:</b>	2
4.1	The Margin and Support Vectors,	1
4.2	Slack Variables for Non-Linearly Separable Problems,	2
4.3	the Support Vector Machine Algorithm,	2
4.4	Extensions to the SVM	2
5	<b>Reinforcement Learning:</b>	2
5.1	Markov Decision Processes	2
5.2	Uses of Reinforcement Learning	1
6	<b>Unsupervised Learning:</b>	1
6.1	The K-Means Algorithm	2
6.2	Vector Quantisation	2
6.3	the SOM Algorithm	2
	Total	40

### Course Designers

Dr. S. Vijayalakshmi

[svlcse@tce.edu](mailto:svlcse@tce.edu)

Dr.T. Chandrakumar

[tckcse@tce.edu](mailto:tckcse@tce.edu)

**20CAPE0****DATA ANALYTICS**

Category	L	T	P	Credit
PE	3	0	0	3

**Preamble**

This course is designed to introduce the basics of data analytics and apply them in the real time analysis of data. The course is intended to facilitate the students to apply statistical methods of exploring data and introduce machine learning methods for data analytics.

**Prerequisite**

- None

**Course Outcomes**

On the successful completion of the course, students will be able to:

CO1:	Explain the nature, source and uses of data	Understand
CO2:	Analyze the correlation between various parameters of a data set using suitable techniques through statistical study	Analyze
CO3:	Apply statistical learning using various regression techniques in analytics	Apply
CO4:	Apply supervised learning techniques for data analytics	Apply
CO5:	Apply Unsupervised learning techniques for data analytics	Apply
CO6:	Identify suitable technique for various stages of data analytics	Analyze

**Mapping with Programme Outcomes**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	L	L	M	M	-	-	-	-	-	-	-
CO2	S	S	S	M	L	-	M	-	-	-	-	-
CO3	S	S	S	M	-	-	-	-	-	-	-	-
CO4	S	S	S	M	L	-	M	-	-	-	-	-
CO5	S	S	S	M	L	-	M	-	-	-	-	-
CO6	S	S	S	M	L	-	S	-	-	-	-	-
CO7	S	L	L	-	S	L	L	-	-	-	-	-

S- Strong; M-Medium; L-Low

**Assessment Pattern**

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	20	20	20	20
Understand	30	20	30	20
Apply	30	30	30	40
Analyse	20	30	20	20
Evaluate	0	0	0	0
Create	0	0	0	0

**Course Level Assessment Questions**

**Explain the nature, source and uses of data (CO1):**

1. Write a brief note on the different sources of data in any Engineering application.

BOS meeting approved: 20-04-2020

Approved in 60<sup>th</sup> Academic Council meeting on 25-07-2020

2. List few modes of availability of data for a management information system.

**Analyze the correlation between various parameters of a data set using suitable techniques (CO2):**

1. An employee has a rating of 5.7 on a measure of job performance. The mean for employees at that company is 5.5 and the standard deviation is .4. What is that employee's percentile score for job performance? What is their T-Score?
2. An investigator predicts that individuals that fit the Type A Behavior Pattern (highly competitive and time conscious) will have higher scores on a questionnaire measure of need for achievement than individuals that fit the Type B Behavior pattern (a absence of Type qualities). The investigator collects need for achievement scores from 10 Type A subjects and 10 Type B subjects. Higher scores reflect greater levels of need for achievement. Write the null and alternative hypotheses for testing this prediction. Test the null hypothesis stated above. Please provide a sentence for a results section that states the conclusion the investigator is entitled to draw.

Type A Type B

Type A	Type B
12	8
10	10
8	5
11	7
15	8
12	5
9	4
16	7
11	8
8	10



3. If the null hypothesis for a study is that aerobic instructors do not have faster metabolisms than bodybuilders, choose the appropriate research hypothesis among the given list.
  - A. Bodybuilders do not have faster metabolisms than aerobic instructors.
  - B. Bodybuilders and aerobics instructors have the same metabolism rate.
  - C. Aerobics instructors have slower metabolisms than bodybuilders.
  - D. Aerobics instructors have faster metabolisms than bodybuilders

**Apply statistical learning using various regression techniques in analytics (CO3):**

1. Illustrate the sequential steps in Ridge regression.
2. Explain in detail about Lasso regression with an example.
3. Demonstrate Multiple linear regression with suitable illustrations.

**Apply supervised learning techniques for data analytics (CO4):**

1. What is the relation between Linear Discriminant analysis and Quadratic discriminant analysis?
2. Show that the difference of the two linear discriminant functions is

$$D(z) := \delta_1(z) - \delta_0(z) = (z - \mu)^T \Sigma^{-1/2} \delta - \frac{1}{2} \delta^T \delta$$

and that LDA classification is obtained by predicting G equal to 0 if  $D(z) < 0$  and 1 if  $D(z) > 0$ .

3. Assume everything else remains same, which of the following is the right statement about the predictions from decision tree in comparison with predictions from Random Forest?
  - A. Lower Variance, Lower Bias
  - B. Lower Variance, Higher Bias
  - C. Higher Variance, Higher Bias

## D. Lower Bias, Higher Variance

**Apply Unsupervised learning techniques for data analytics (CO5):**

1. Consider the following data set consisting of the scores of two variables on each of seven individuals:

3. Subject	4. A	5. B
6. 1	7. 1.0	8. 1.0
9. 2	10. 1.5	11. 2.0
12. 3	13. 3.0	14. 4.0
15. 4	16. 5.0	17. 7.0
18. 5	19. 3.5	20. 5.0
21. 6	22. 4.5	23. 5.0
24. 7	25. 3.5	26. 4.5

This data set is to be grouped into two clusters. Apply k-means algorithm.

2. Demonstrate Hierarchical clustering with an example scenario.
3. Explain Principal component analysis with an example.

**Identify suitable technique for various stages of data analytics (CO6):**

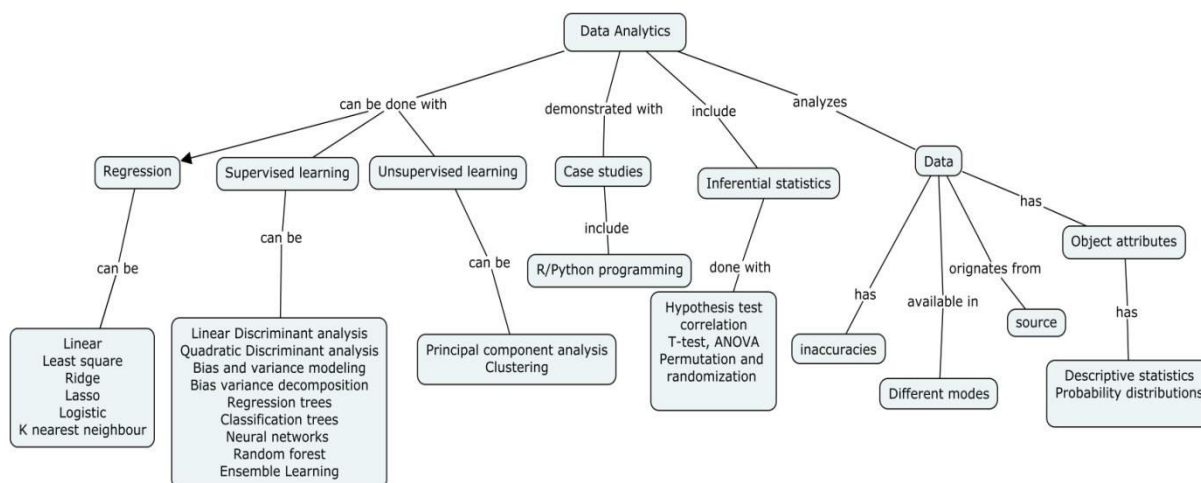
1. Can a random forest be used for feature selection in multiple linear regression? Analyze.
2. The data scientists at "BigMart Inc" have collected 2013 sales data for 1559 products across 10 stores in different cities. Also, certain attributes of each product based on these attributes and store have been defined. The aim is to build a predictive model and find out the sales of each product at a particular store during a defined period. Which learning problem does this belong to?
  1. Supervised learning
  2. Unsupervised learning
  3. Reinforcement learning
3. We want to find which feature would be better for splitting root node (where root node represents entire population). For this, we will set "Reduction in Variance" as our splitting method.

Outlet_Location_Type	Item_Fat_Content	Item_Outlet_Sales
Tier 1	Low Fat	3735.1380
Tier 3	Regular	443.4228
Tier 1	Low Fat	2097.2700
Tier 3	Regular	732.3800
Tier 3	Low Fat	994.7052

The split with lower variance is selected as the criteria to split the population. Among Between Outlet\_Location\_Type and Item\_Fat\_Content, which was a better feature to split?

- A. Outlet\_Location\_Type
- B. Item\_Fat\_Content
- C. will not split on both

## Concept Map



## Syllabus

**Introduction:** Intelligent data analysis – changing data - Nature of data – Modern data analytic tools

**Statistical concepts :** Probability – Probability distribution – Random sampling - Statistical inference – Prediction and prediction errors – Resampling.

**Statistical learning:** Introduction - Assessing model accuracy - Regression techniques: Simple linear – Multiple linear – Logistic regression – Case study .

**Supervised learning Techniques:** Linear Discriminant Analysis – Quadratic Discriminant Analysis - Comparison of classification methods - Subset selection – Shrinkage methods – Dimension reduction methods – Considerations in high dimensions – Basics of decision tree – Bagging – Random forests – Boosting. – Case study.

**UnSupervised Learning Techniques:** K-means clustering – Hierarchical clustering – case study.

## Reference Books & web resources

1. James, Gareth, Daniela Witten, and Trevor Hastie. "An Introduction to Statistical Learning: With Applications in R." , Springer, 1<sup>st</sup> edition , 6<sup>th</sup> printing, 2016. Chapters 1,2,3,4,5,6,8 & 10.
2. Michael R. Berthold, David J. Hand, "Intelligent Data analysis: An introduction", Springer, 2006. Chapters 1 & 2.
3. John W. Foreman, " Data smart: Using Data Science to Transform Information into Insight", Wiley publications, 2013.
4. Foster Provost, Tom Fawcett, "What You Need to Know about Data Mining and Data-Analytic Thinking", O'Reilly Media, 2013.
5. Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. "The Elements of Statistical Learning: Data Mining, Inference, and Prediction." *Springer Series in Statistics*, 2009. <http://statweb.stanford.edu/~tibs/ElemStatLearn/>
6. Montgomery, D. C., and G. C. Runger, Applied Statistics and Probability for Engineers. John Wiley & Sons, 2010.
7. Witten, I. H., E. Frank, and M. A. Hall, Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann, 2016.

**Course Contents and Lecture Schedule**

Module No.	Topic	No. of Lecture hours
1.	<b>Introduction</b>	
1.1	Intelligent data analysis – changing data	1
1.2	Nature of data – Modern data analytic tools	1
2.	<b>Statistical concepts</b>	
2.1	Probability – Probability distribution	1
2.2	Random sampling	1
2.3	Statistical inference	2
2.4	Prediction and prediction errors	1
2.5.	Resampling	1
3	<b>Statistical learning</b>	
3.1	Introduction - Assessing model accuracy	2
3.2	Regression techniques	
3.2.1	Simple linear regression	1
3.2.2	Multiple linear regression	2
3.2.3	Logistic regression	2
3.3	Case study	2
4	<b>Supervised learning Techniques</b>	
4.2	Linear Discriminant Analysis	2
4.3	Quadratic Discriminant Analysis	1
4.4	Comparison of classification methods	1
4.5	Subset selection	2
4.6	Shrinkage methods	2
4.7	Dimension reduction methods, Considerations in high dimensions	1
4.8	Basics of decision tree	2
4.9	Bagging – Random forests – Boosting	1
4.10	Case study	2
5	<b>Unsupervised Learning Techniques</b>	
5.2	Clustering	1
5.2.1	K-means clustering for analytics	1
5.2.2.	Hierarchical clustering – An analytical perspective	1
5.2.3.	Case study	2
	<b>Total</b>	<b>36</b>

**Course Designer**

Dr. D. Anitha

[anithad@tce.edu](mailto:anithad@tce.edu)

<b>20CAPF0</b>	<b>PROGRAMMING IN C# USING .NET</b>	Category	L	T	P	Credit
		PE	3	0	0	3

**Preamble**

The course facilitates the students to understand the syntactical features of C# language and use the design of the language to develop robust software.

**Prerequisite**

- 20CA120 : Programming in C
- 20CA220 : Object Oriented Programming in C++

**Course Outcomes**

On the successful completion of the course, students will be able to

CO1: Explain the .NET framework	Understand
CO2: Apply the general programming structure of C# in developing software solutions based on user requirements	Apply
CO3: Develop windows application and web applications in .NET framework analyzing user requirements	Analyze
CO4: Demonstrate working advanced features of .NET programming	Understand
CO5: Design a document for GUI applications.	Apply
CO6: Develop Assemblies and Deployment in .NET for Mobile Application Development based on requirements	Apply

**Mapping with Programme Outcomes**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	S	L	M	M	L	L	M	L	-	-	-	-
CO2	S	L	L	L	S	L	S	-	-	-	-	-
CO3	S	M	L	L	S	L	L	-	-	-	-	-
CO4	S	M	L	L	-	M	M	L	-	-	-	-
CO5	S	L	M	L	S	L	-	-	-	-	-	-
CO6	S	L	L	L	S	L	L	-	-	-	-	-

S- Strong; M-Medium; L-Low

**Assessment Pattern**

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	20	20	20	20
Understand	30	20	20	20
Apply	40	50	50	50
Analyse	10	10	10	10
Evaluate	0	0	0	0
Create	0	0	0	0

## Course Level Assessment Questions

### Explain the .NET framework (CO1):

1. List new features that are unique to c# language
2. Explain name any 4 applications that are supported by .Net framework.
3. Discuss string and string builder in C#
4. Outline the role of Common Language Runtime (CLR) in .NET
5. Explain how to use primitive data types in code and understand Boxing- Unboxing.

### Apply the general programming structure of C# in developing software solutions based on user requirements (CO2):

1. Structures and enumerations concepts to Create an array of 10 integers. Populate the array by having the user enter integers at the console (Console Readline) Output the integers sorted from the greatest to least.
2. The methods available for window based applications and build an application to accept the reservation details of a train ticket and to store the details in a database table. Use drop down list box to choose the train number and name. Accept source and destinations in text boxes. Allow the user to enter the date of journey one month in advance. Assume that in each train, there are thirty tickets and every booking should have a unique number.
3. Implement Guest Book with name, email and comment textboxes and necessary controls to validate these fields. It should be able to view the guests using datagridview

### Develop windows application and web applications in .NET framework analyzing user requirements (CO3):

1. Analyze array list differ from array
2. Analyze a program that assigns the value 25 to variable x and 5 to variable y. Output the sum, difference, quotient, product and modulus of x and y. What will be the output of the following method? Why?  

```
Static void Main ()
{
  intvarA = 5;
  intvarB = ++varA;
  intvarC = varB++;
  Console.WriteLine("A:{0}, B:{1}, C:{2}",varA, varB, varC); }

```
3. Analyze the building blocks of an XML web service.

### Demonstrate working advanced features of .NET programming (CO4):

1. Explain the methods to compare two objects in C#.
2. Discuss in detail about the activities of CLR
3. Discuss the distributed application in which marshalling and assembling finds their usage.
4. Explain about interfaces in C#

### Design a document for GUI applications (CO5):

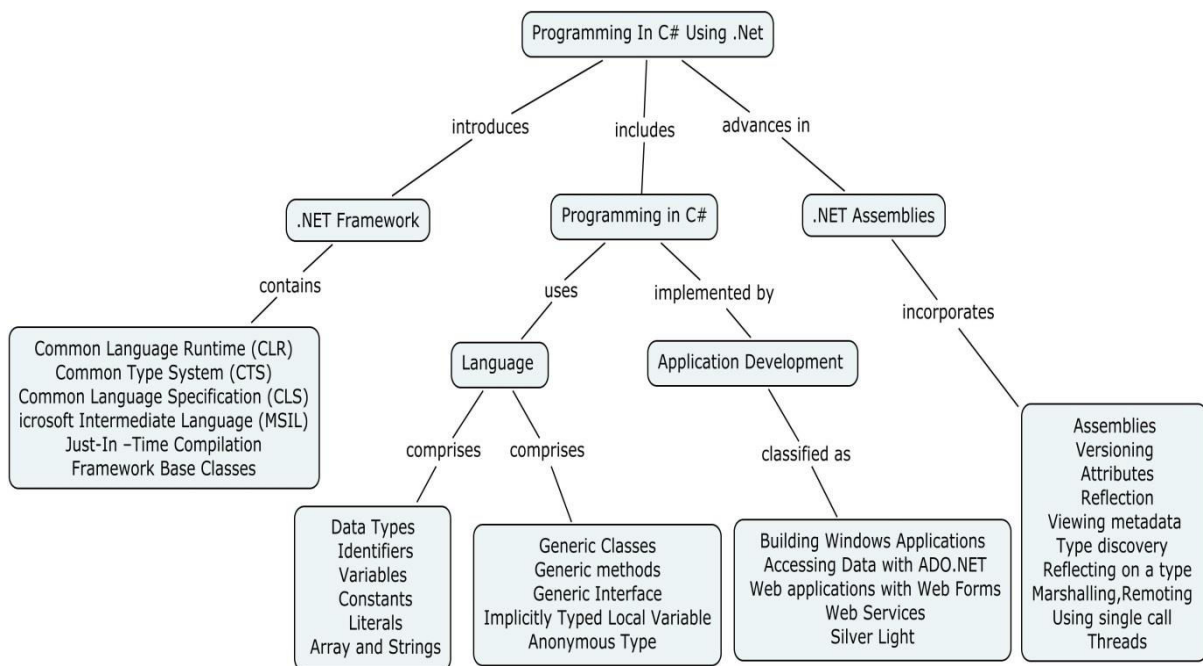
1. Develop a C# program to write student database in a flat file and read from the same file and print the 'marklist' of students.
2. Apply the features of .NET web forms and Web Pages and design an application
3. Concept of inheritance and polymorphism to find area of various shapes rectangle, circle and triangle



## Develop Assemblies and Deployment in .NET for Mobile Application Development based on requirements (CO6):

1. Justify your argument over using .NET frame work for a distributed application.
2. Write a C# code for Exception Handling.
3. Create an abstract 'Animal; class that has private members weight and name abstract methods Speak(),Move() and ToString().derive from Animal a 'Cat' and a 'dog' class that override the methods appropriately.Create an Animal array,Populate it with Dogs and cats and then call each member's overridden virtual methods.

### Concept Map



### Syllabus

**The .Net framework:** Introduction, The Origin of .Net Technology, Common Language Runtime (CLR), Common Type System (CTS), Common Language Specification (CLS), Microsoft Intermediate Language (MSIL), Just-In-Time Compilation, Framework Base Classes.

**C -Sharp Language:** Introduction-Boxing-unboxing, Interfaces, Properties, indexes, Namespace. Method parameter modifiers- Out-ref- Params. Decision Constructs and iteration constructs.

**Introduction-**Data Types, Identifiers, Variables, Constants, Literals, Array and Strings,- OOPS concepts-Delegates and Events- -Generic Classes-Generic methods-Generic Interface-Implicitly Typed Local Variable- Anonymous Type. Implicitly Typed Local Variable- Anonymous Type, Object Initializers. LINQ-Introduction

**Application Development on .NET:** Building Windows Applications. Accessing Data with ADO.NET. Web applications with Web Forms, Web Services

**.NET Assemblies:** Assemblies- Versioning- Attributes- Reflection- Viewing metadata - Type discovery - Reflecting on a type - Marshalling - Remoting - Using single call - Threads-Silver Light.

**Reference Books**

1. Andrew Troelsen, "Pro C# with .NET 3.0 4/E", APress, 2009.
2. Herbert Schildt, C# 3.0 The Complete Reference 4/E., McGraw-Hill, 2009.
3. Liberty, J., Donald Xie, "Programming C# 3.0", 5th Edition, O'Reilly, 2007
4. Robinson, "Professional C#", 3rd Edition, Wrox Press, 2004.

**Course Contents and Lecture Schedule**

Module No	Topic	No. of Lectures
1	<b>The .Net framework</b>	
1.1	Introduction, The Origin of .Net Technology,.	1
1.2	Common Language Runtime (CLR), Common Type System (CTS), Common Language Specification (CLS)	1
1.3	Microsoft Intermediate Language (MSIL), Just-In –Time Compilation	2
1.4	Framework Base Classes	2
2	<b>C –Sharp Basics</b>	
2.1	Data Types, Identifiers, Variables, Constants, Literals	2
2.2	Introduction-Boxing - Unboxing,	1
2.3	Interfaces – Properties – Indexes - Namespace	2
2.4	Method parameter modifiers- out - ref- params	2
2.5	Decision constructs - Iteration constructs	2
3	<b>C# Types Construction</b>	
3.1	Array and Strings, OOPS concepts	1
3.2	Delegates and Events	2
3.3	Generic Classes-Generic methods-Generic Interface	3
3.4	Implicitly Typed Local Variable- Anonymous Type, Object Initializers	1
3.5	LINQ-Introduction	1
4	<b>Application Development on .NET</b>	
4.1	Building Windows Applications	1
4.2	Accessing Data with ADO.NET	1
4.3	Web applications with Web Forms	2
4.4	Web Services	2
5	<b>C# Advanced Concepts</b>	
5.1	Assemblies-Introduction	1
5.2	Versioning, Attributes	2
5.3	Reflection- Viewing metadata Type discovery – Reflecting on a type	2
5.4	Marshalling – Remoting	1
5.5	Using single call – Threads	1
5.6	SilverLight	1
	Total	36

**Course Designer:**

Prof. N. Murali

nmcse@tce.edu

<b>20CAPG0</b>	<b>CLOUD COMPUTING</b>	Category	L	T	P	Credit
		PE	3	0	0	3

**Preamble**

Cloud computing paradigm covers a range of distributed computing, hosting and access solutions, including service-based computing. The objective of the course is to provide comprehensive and in-depth knowledge of Cloud Computing concepts, technologies, architecture and researching state-of-the-art in Cloud Computing fundamental issues, technologies, applications and implementations.

**Prerequisite**

- None

**Course Outcomes**

On the successful completion of the course, students will be able to

CO1: Describe the key technologies, architecture, strengths, limitations and applications of cloud computing	Understand
CO2: Explain the types and service models of cloud.	Understand
CO3: Describe the core issues such as security, privacy, and interoperability in cloud platform.	Understand
CO4: Apply suitable technologies, algorithms, and applications in the cloud computing driven systems	Apply
CO5: Provide appropriate cloud computing solutions for the given scenario	Analyze
CO6: Analyze how applications are deployed in cloud	Analyze

**Mapping with Programme Outcomes**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M											M
CO2	M				L							M
CO3	M											M
CO4	S	M	L		S		S	S	S	S		S
CO5	S	S	M	L	S	S	S	S	S	S	S	S
CO6	M	S			S		L					

S- Strong; M-Medium; L-Low

**Assessment Pattern**

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	20	20	20	20
Understand	50	40	40	40
Apply	30	40	40	40
Analyse	20	20	20	20
Evaluate	0	0	0	0
Create	0	0	0	0

### Course Level Assessment Questions

#### Describe the key technologies, architecture, strengths, limitations and applications of cloud computing (CO1):

1. Define Cloud computing, Enlist and explain essential characteristics of cloud computing.
2. What is self service provisioning?
3. Explain in brief, how cloud helps reducing capital expenditure?
4. What is the difference between process virtual machines, host VMMs and native VMMs?
5. What is the fundamental differences between the virtual machine as perceived by a traditional operating system processes and a system VM?

#### Explain the types and service models of cloud (CO2):

1. Explain the services provided by the Amazon infrastructure cloud from a user perspective.
2. What is cloud computing? Enlist and explain three service models, and four deployment models of cloud computing.
3. Explain a user view of Google App Engine with suitable block schematic.

#### Describe the core issues such as security, privacy, and interoperability in cloud platform (CO3)

1. Examine some of the common pitfalls that come with virtualization.
2. Show the reasons of Cloud Computing brings new threats
3. Classify secure execution environment and communication in cloud?
4. Show risks from multi-tenancy, with respect to various cloud environments.
5. Illustrate trusted cloud computing?

#### Apply suitable technologies, algorithms, and applications in the cloud computing driven systems (CO4)

1. Construct the SOAP and REST paradigms in the context of programmatic communication between applications deployed on different cloud providers, or between cloud applications and those deployed in -house.
2. Show conceptual representation of the Eucalyptus Cloud. Explain in brief the components within the Eucalyptus system.
3. Illustrate Nimbus? What is the main way to deploy Nimbus Infrastructure? What is the difference between cloudinit.d and the Context Broker?
4. Show Open Nebula Cloud? Explain main components of Open Nebula.
5. Show Xen Cloud Platform (XCP) with suitable block diagram.

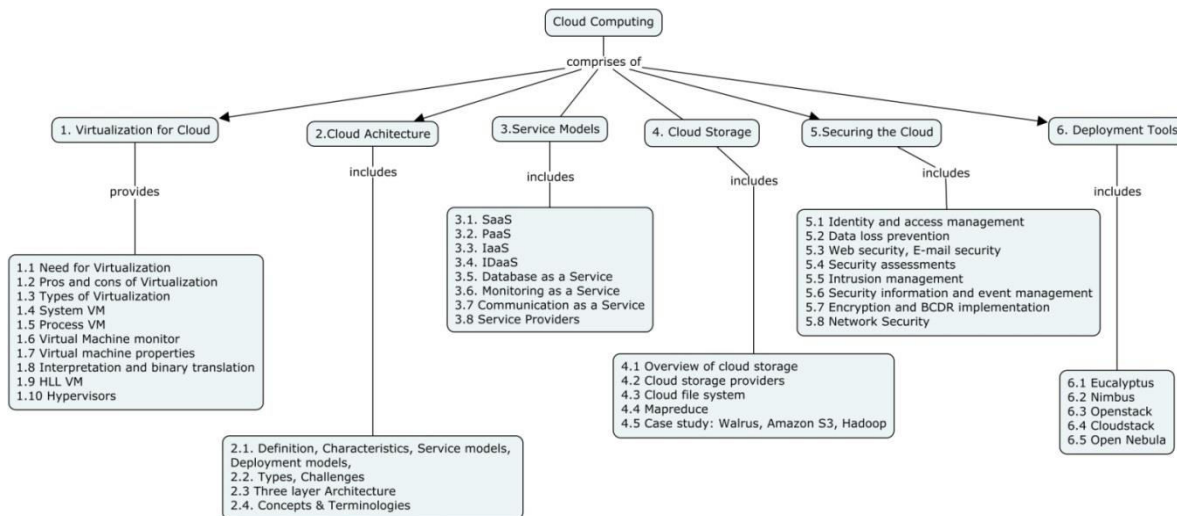
#### Provide appropriate cloud computing solutions for the given scenario (CO5)

1. Construct the architecture of cloud file systems (GFS, HDFS).
2. Solve with suitable example, how a relational join could be executed in parallel using MapReduce.
3. Show how Big tables are stored on a distributed file system such as GFS or HDFS.
4. Construct MapReduce model with suitable example.

#### Analyze how applications are deployed in cloud (CO6):

1. Explaining how applications are deployed on IaaS, PaaS, private, public and hybrid clouds.
2. Classify the Deployment tools for Cloud Application.
3. Identify different models for deployment in cloud computing.

## Concept Map



## Syllabus

**Virtualization for Cloud:** Need for Virtualization, Pros and cons of Virtualization, Types of Virtualization, System VM, Process VM, Virtual Machine monitor, Virtual machine properties, Interpretation and binary translation, HLL VM, Hypervisors : Xen, KVM , VMWare, Virtual Box, Hyper-V.

**Cloud Architecture:** Definition, Characteristics, Service models, Deployment models, Types, Challenges, Three-layer architecture, Concepts & Terminologies - Virtualization, Load balancing, Scalability and elasticity, Deployment, Replication, Monitoring, Software defined networking, Network function virtualization, Service level agreement, Billing.

**Service Models:** SaaS – Multitenant, OpenSaaS, SOA. PaaS – IT Evolution, Benefits, Disadvantages. IaaS – Improving performance, System and storage redundancy, Cloud based NAS devices, Advantages, Server types. IDaaS – Single Sign-on, OpenID. Database as a Service, Monitoring as a Service, Communication as services. Service providers - Google, Amazon, Microsoft Azure, IBM, Sales force.

**Cloud Storage:** Overview of cloud storage, cloud storage providers, Cloud file system, Mapreduce. Case study: Walrus, Amazon S3, Hadoop.

**Securing the Cloud:** Identity and access management, Data loss prevention, Web security, E-mail security, Security assessments, Intrusion management, Security information and event management, Encryption and BCDR implementation, Network Security.

**Deployment Tools:** Eucalyptus, Nimbus, Openstack, Cloudstack, Open Nebula.

## Reference Books

1. James E Smith, Ravi Nair, "Virtual Machines", Morgan Kaufmann Publishers, 2006.
2. John Rittinghouse & James Ransome, "Cloud Computing, Implementation, Management and Strategy", CRC Press, 2010.
3. T. Velte, A. Velte, R. Elsenpeter, "Cloud Computing, A Practical Approach", McGraw-Hill, 2009.
4. Cloud Security Alliance, "Providing greater clarity in Security as a Service", 2013.
5. RajkumarBuyya, Christian Vecchiola, S.ThamaraiSelvi, "Mastering cloud

- computing”,  
Morgan Kaufman, 2013.
6. Dr. Kris Jamsa, “Cloud Computing: SaaS, PaaS, IaaS, Virtualization, Business Models, Mobile, Security and More”, Jones and Bartlett learning, First edition, 2013.
  7. Arshdeep Bahga, Vijay Madisetti, “Cloud Computing: A Hands-On Approach”, CreateSpace Independent Publishing Platform, 1st edition, 2013.
  8. Massimo Cafaro (Editor), Giovanni Aloisio (Editor), “Grids, Clouds and Virtualization” Springer; edition, 2011.
  9. Gautam Shroff, “Enterprise Cloud Computing Technology Architecture Applications”, Cambridge University Press; 1 edition, 2010.

### Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures
1	<b>Virtualization for Cloud</b>	
1.1	Need for Virtualization	1
1.2	Pros and cons of Virtualization	
1.3	Types of Virtualization	1
1.4	System VM	
1.5	Process VM	2
1.6	Virtual Machine monitor	
1.7	Virtual machine properties,	2
1.8	Interpretation and binary translation	
1.9	HLL VM	2
1.10	Hypervisors :Xen, KVM , VMWare, Virtual Box, Hyper-V.	1
2	<b>Cloud Architecture</b>	
2.1	Definition, Characteristics, Service models, Deployment models	1
2.2	Types, Challenges	1
2.3	Three-layer architecture	
2.4	Concepts & Terminologies - Virtualization, Load balancing, Scalability and elasticity, Deployment, Replication, Monitoring, Software defined networking, Network function virtualization, Service level agreement, Billing.	2
3	<b>Service Models</b>	
3.1	SaaS – Multitenant, OpenSaaS, SOA.	1
3.2	PaaS – IT Evolution, Benefits, Disadvantages.	1
3.3	IaaS – Improving performance, System and storage redundancy, Cloud based NAS devices, Advantages, Server types.	2
3.4	IDaaS – Single Sign-on, OpenID.	1
3.5	Database as a Service	1
3.6	Monitoring as a Service	1
3.7	Communication as services	1
3.8	Service providers - Google, Amazon, Microsoft Azure, IBM, Sales force.	1
4	<b>Cloud Storage</b>	
4.1	Overview of cloud storage	1
4.2	Cloud storage providers	1
4.3	Cloud file system	1
4.4	Mapreduce.	1

4.5	Case study: Walrus, Amazon S3, Hadoop.	2
5	<b>Securing the Cloud</b>	
5.1	Identity and access management	1
5.2	Data loss prevention	
5.3	Web security, E-mail security	1
5.4	Security assessments	
5.5	Intrusion management	1
5.6	Security information and event management	
5.7	Encryption and BCDR implementation	2
5.8	Network Security.	
6	<b>Deployment Tools</b>	
6.1	Eucalyptus	1
6.2	Nimbus	
6.3	Openstack	1
6.4	Cloudstack	
6.5	Open Nebula	1
<b>Total Lectures</b>		<b>36</b>

**Course Designer:**

Dr. T. Chandrakumar

tckcse@tce.edu



<b>20CAPH0</b>	<b>INTERNET OF THINGS</b>	Category	L	T	P	Credit
		PE	3	0	0	3

### Preamble

This course aims at providing a basic understanding of Internet of Things. It aims at providing hands on training for building simple applications using appropriate sensors, microcontroller board and other components.

### Prerequisite

20CA120 – Programming in C

### Course Outcomes

Upon successful completion of this course students will be able to:

CO1: Describe Internet of Things and the protocols of IoT	Remember
CO2: Identify the various IoT components such as sensors, shields (Arduino, Raspberry Pi, Bluetooth, WiFi)	Understand
CO3: Program using Arduino studio and Rasperry Pi toolkits	Apply
CO4: Experiment various problems related with IoT using controllers and processors.	Apply
CO5: Explain the role of cloud and security in IoT	Understand
CO6: Demonstrate various applications of IoT integrating with Cloud and ensuring security	Apply

### Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M											
CO2	S	M										M
CO3	S	M										M
CO4	S	M										M
CO5	M	L										
CO6	S	M	L		S	M	S	L	S	S	M	M

S- Strong; M-Medium; L-Low

### Assessment Pattern

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	30	20	20	20
Understand	40	40	40	40
Apply	30	40	40	40
Analyse	0	0	0	0
Evaluate	0	0	0	0
Create	0	0	0	0

### CO6 is assessed through Mini Project. Mini Project Details

- Team formation (Team size: 5)
- Problem identification on various IT, societal, business and environmental needs (Eg. IoT based Smart Bin Application, Home



- Automation etc.)
- Identify and assemble the appropriate components needed to build the microcontroller board.
- Test the board with sample input.

### Course Level Assessment Questions

#### Describe Internet of Things and the protocols of IoT (CO1):

1. Discuss the Evolution of IoT.
2. List the some of the protocols involved in IoT.
3. Distinguish among Raspberry Pi, Arduino and Zigbee.
4. Describe the architecture of Cloud of Things.
5. Discuss the two pillars of IoT

#### Identify the various IoT components such as sensors, shields (Arduino, Raspberry Pi, Bluetooth, WiFi) (CO2):

1. Illustrate the Arduino board with a diagram and name its components.
2. Illustrate the Rasperry Pi with a diagram and name its components.

#### Program using Arduino studio and Rasperry Pi toolkits (CO3):

1. Write a program to read the input from a switch port and copy it to a LED.
2. Write a program to glow an LED using Arduino

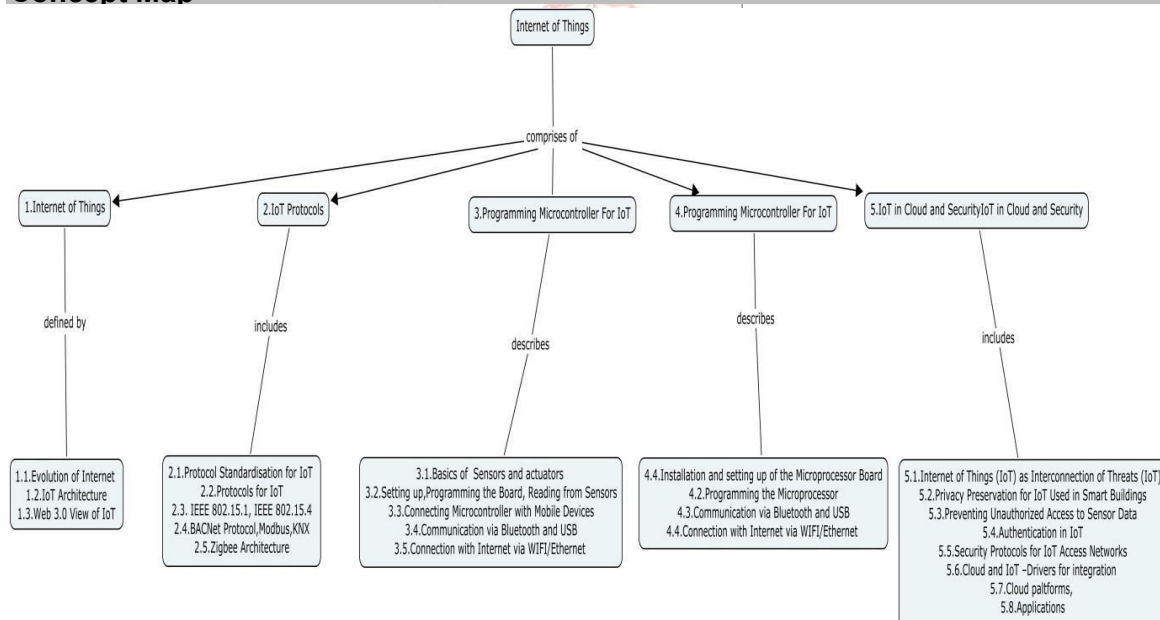
#### Experiment various problems related with IoT using controllers and processors (CO4):

1. Illustrate how a microcontroller can be connected to mobile device.
2. Show the connectivity of microcontroller with Bluetooth and USB.
3. Demonstrate the connectivity issues involved in IoT

#### Explain the role of cloud and security in IoT (CO5):

1. Discuss the integration of IoT with Cloud
2. Describe the right components to build cloud based applications in IoT
3. List the security issues related to IoT

### Concept Map



## Syllabus

**Internet of Things:** Evolution of Internet, IoT Architecture, Web 3.0 View of IoT

**IoT Protocols:** Protocol Standardisation for IoT ,Protocols for IoT, IEEE 802.15.1, IEEE 802.15.4,BACNet Protocol, Modbus, KNX, Zigbee Architecture

**Programming Microcontroller For IoT:** Basics of microcontroller, Setting up, Programming the Board, Reading from Sensors, Connecting Microcontroller with Mobile Devices, Communication via Bluetooth and USB, Connection with Internet via WIFI/Ethernet

**Programming Microprocessor For IoT:** Installation and Setting up of the Microprocessor board, Programming the Microprocessor, Communication via Bluetooth and USB, Connection with Internet via WIFI/Ethernet

**IoT in Cloud and Security:** Internet of Things (IoT) as Interconnection of Threats (IoT), Privacy Preservation for IoT Used in Smart Buildings, Preventing Unauthorized Access to Sensor Data, Authentication in IoT, Security Protocols for IoT Access Networks, Cloud and IoT –Drivers for integration ,Cloud platforms, Applications

## Reference Books & Web resources

1. Charalampos Doukas -Building Internet of Things with the Arduino, Create space, April 2012.
2. Fei Hu- Security and Privacy in Internet of Things (IoTs): Models, Algorithms and Implementation, CRC press ,2016
3. Dieter Uckelmann, Mark Harrison, Florian Michahelles -Architecting the Internet of Things, Springer,2011
4. Donald Norris -The Internet of Things: Do-It-Yourself at Home Projects for Arduino, Raspberry Pi and BeagleBone Black, Mc.Graw Hill,2015.
5. Cuno Pfister -Getting Started with the Internet of Things, O'Reilly Media, Inc.,2011
6. Honbo Zhou -The Internet of Things in the Cloud: A Middleware Perspective, CRC Press,2012
7. Olivier Hersent, David Boswarthick, Omar Elloumi, -The Internet of Things, Key applications and Protocols, Wiley, 2012
8. <https://www.raspberrypi.org/>
9. <https://developer.ibm.com/iot/>
10. <http://www.microsoft.com/en-in/server-cloud/internet-of-things.aspx>

## Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures
<b>1</b>	<b>Internet of Things</b>	
1.1	Evolution of Internet	1
1.2	IoT Architecture	
1.3	Web 3.0 View of IoT	1
<b>2</b>	<b>IoT Protocols</b>	
2.1	Protocol Standardisation for IoT	1
2.2	Protocols for IoT	2
2.3	IEEE 802.15.1, IEEE 802.15.4	1
2.4	BACNet Protocol,Modbus,KNX	1
2.5	Zigbee Architecture	1

<b>3</b>	<b>Programming Microcontroller For IoT</b>	
3.1	Basics of sensors and actuators	2
3.2	Setting up, Programming the Board, reading from Sensors	2
3.3	Connecting Microcontroller with Mobile Devices	2
3.4	Communication via Bluetooth and USB	2
3.5	Connection with Internet via WIFI/Ethernet	2
<b>4</b>	<b>Programming Microprocessor For IoT</b>	
4.1	Installation and Setting up of the Microprocessor board	2
4.2	Programming the Microprocessor	2
4.3	Communication via Bluetooth and USB	2
4.4	Connection with Internet via WIFI/Ethernet	2
<b>5</b>	<b>IoT in Cloud and Security</b>	
5.1	Internet of Things (IoT) as Interconnection of Threats (IoT)	1
5.2	Privacy Preservation for IoT Used in Smart Buildings	1
5.3	Preventing Unauthorized Access to Sensor Data	1
5.4	Authentication in IoT	1
5.5	Security Protocols for IoT Access Networks	1
5.6	Cloud and IoT –Drivers for integration	1
5.7	Cloud paltforms	1
5.8	Applications	1
	<b>Total Lectures</b>	<b>36</b>

**Course Designers:**

Dr. D. Anitha

anithad@tce.edu

<b>20CAPJ0</b>	<b>CYBER SECURITY AND LAW</b>	Category	L	T	P	Credit
		PE	3	0	0	3

**Preamble**

The course aims to provide an understanding of enterprise structure, policies, accountability mechanisms and monitoring practices in place to achieve cyber security. Also, knowledge on security mechanisms and cyber law to ensure confidentiality, integrity and availability of information assets is provided.

**Prerequisite**

- None

**Course Outcomes**

On successful completion of the course, the students will be able to

CO1:	Recognize the cyber security policies, standards and procedures for completeness and alignment with generally accepted practices	Understand
CO2:	Explain the functionalities of cyber security	Understand
CO3:	Describe E governance and IT Act	Understand
CO4:	Analyze the root causes of cyber crime	Apply
CO5:	Implement suitable security techniques for a given problem	Apply
CO6:	Suggest appropriate security countermeasures for the given scenario	Analyze

**Mapping with Programme Outcomes**

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	L	M	L	M							
CO2	M	L	M	L	M							
CO3	L	L	L	L	L							
CO4	S	S	S	M	S							
CO5	S	S	S	M	S							
CO6	S	S	S	M	S							L

S- Strong; M-Medium; L-Low

**Assessment Pattern**

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	30	20	20	20
Understand	40	40	40	40
Apply	30	30	30	30
Analyze	0	10	10	10
Evaluate	0	0	0	0
Create	0	0	0	0

### Course Level Assessment Questions

**Recognize the cyber security policies, standards and procedures for completeness and alignment with generally accepted practices (CO1):**

1. Identify the cyber security policies, standards and procedures for completeness
2. Why cyber security is needed
3. What are the practices followed?

**Explain the functionalities of cyber security (CO2):**

1. What are the functionalities of cyber security
2. Explain the Cyber security techniques
3. List down the objectives of cyber security

**Describe E governance and IT Act (CO3):**

1. List down the legal recognition of electronic records
2. List down the legal recognition of digital signatures

**Analyze the root causes of cyber crime (CO4):**

1. How to tamper the computer source documents
2. Write down offences related with digital signatures

**Implement suitable security techniques for a given problem (CO5):**

1. Convert the message “**Crypto**” by applying

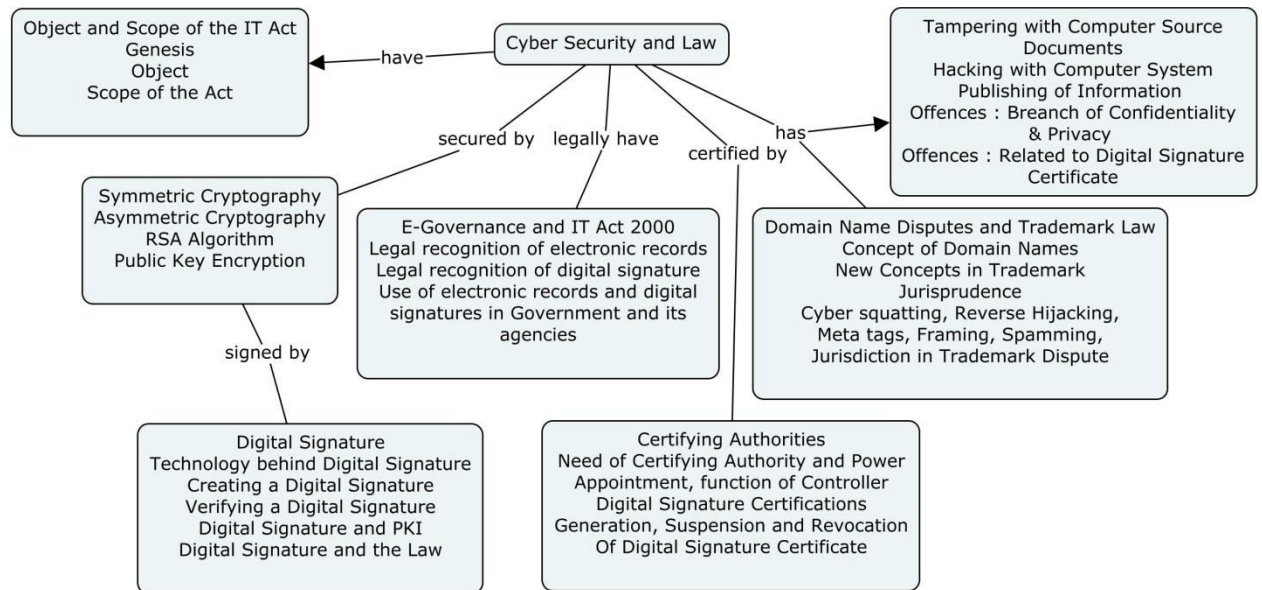
Hill cipher with  $K = \begin{pmatrix} 17 & 17 & 5 \\ 15 & 17 & 6 \end{pmatrix}$  &  $K^{-1} = \begin{pmatrix} 4 & 9 & 15 \\ 24 & 0 & 17 \end{pmatrix}$

2. Do One round **DES** encryption for the given Plaintext **012...F** and Key **ABCDEF0123456789**
3. Analyze the Cipher text for the given Plain text “**THIAGARAJAR COLLEGE OF ENGINEERING**” with the Key **6 3 2 5 4 7 1** by performing 2 stages of **Transposition** techniques.

**Suggest appropriate security counter measures for the given scenario (CO6):**

1. Discuss the Disaster Recovery Plan Tests and Drill.
2. Describe how to Provide assurance that the processes for information systems operations, maintenance and support meet the organization’s strategies and objectives
3. Predict the finest way of conducting periodic reviews of information systems.

## Concept Map



## Syllabus

**Object and Scope of the IT Act**-Genesis, Object, Scope of the Act, **Encryption**-Symmetric Cryptography, Asymmetric Cryptography, RSA Algorithm, Public Key Encryption, **Digital Signature**-Technology behind Digital Signature, Creating a Digital Signature, Verifying a Digital Signature, Digital Signature and PKI, Digital Signature and the Law, **E-Governance and IT Act 2000**-Legal recognition of electronic records- Legal recognition of digital signature-Use of electronic records and digital signatures in Government and its agencies, **Certifying Authorities**-Need of Certifying Authority and Power-Appointment, function of Controller-Digital Signature Certifications-Generation, Suspension and Revocation Of Digital Signature Certificate, **Domain Name Disputes and Trademark Law**-Concept of Domain Names-New Concepts in Trademark Jurisprudence-Cyber squatting, Reverse Hijacking, Meta tags, Framing, Spamming-Jurisdiction in Trademark Dispute, **The Cyber Crimes** - Tampering with Computer Source Documents-Hacking with Computer System-Publishing of Information- Offences : Breach of Confidentiality & Privacy-Related to Digital Signature Certificate.

## Reference books & web resources

1. Vakul Sharma, "Information Technology Law and Practice", 3<sup>rd</sup> edition, Universal Law Publishing Co. Pvt. Ltd., 2012.
2. [Universal's Concise Commentary](#), "The Information Technology Act, 2000" Universal Law Publishing Co. Pvt. Ltd., 2016.
3. William Stallings, Cryptography and Network Security: Principles and Practices, Seventh Edition, Pearson Education Asia, 2016.
4. [https://onlinecourses.nptel.ac.in/noc15\\_cs03](https://onlinecourses.nptel.ac.in/noc15_cs03)
5. [nptel.ac.in/courses/106105031/](https://nptel.ac.in/courses/106105031/)
6. [https://onlinecourses.nptel.ac.in/noc17\\_cs08](https://onlinecourses.nptel.ac.in/noc17_cs08)
7. [https://onlinecourses.nptel.ac.in/noc17\\_cs0](https://onlinecourses.nptel.ac.in/noc17_cs0)
8. [www.nptelvideos.in/2012/11/cryptography-and-network-security.html](http://www.nptelvideos.in/2012/11/cryptography-and-network-security.html)

**Course Contents and Lecture Schedule**

Module No.	Topic	No. of Lectures
<b>1</b>	<b>Object and Scope of the IT Act</b>	
1.1	Genesis	1
1.2	Object	1
1.3	Scope of the Act	1
<b>2</b>	<b>Encryption</b>	
2.1	Symmetric Cryptography	2
2.2	Asymmetric Cryptography	2
2.3	RSA Algorithm	1
2.4	Public Key Encryption	1
<b>3</b>	<b>Digital Signature</b>	
3.1	Technology behind Digital Signature	1
3.2	Creating a Digital Signature	1
3.3	Verifying a Digital Signature	1
3.4	Digital Signature and PKI	1
3.5	Digital Signature and the Law	2
<b>4</b>	<b>E-Governance and IT Act 2000</b>	
4.1	Legal recognition of electronic records	1
4.2	Legal recognition of digital signature	1
4.3	Use of electronic records and digital signatures in Government and its agencies	1
<b>5</b>	<b>Certifying Authorities</b>	
5.1	Need of Certifying Authority and Power	1
5.2	Appointment, function of Controller	2
5.3	Digital Signature Certifications	2
5.4	Generation, Suspension and Revocation	1
<b>6</b>	<b>Domain Name Disputes and Trademark Law</b>	
6.1	Concept of Domain Names	1
6.2	New Concepts in Trademark Jurisprudence	1
6.3	Cyber squatting, Reverse Hijacking, Meta tags, Framing, Spamming	2
6.4	Jurisdiction in Trademark Dispute	1
<b>7</b>	<b>The Cyber Crimes (S-65 to S-74)</b>	
7.1	Tampering with Computer Source Documents	1
7.2	Hacking with Computer System	1
7.3	Publishing of Information	1
7.5	Offences : Breach of Confidentiality & Privacy	1
7.6	Offences : Related to Digital Signature Certificate	1
	<b>Total Lectures</b>	<b>36</b>

**Course Designer:**

Dr. S. Vijayalakshmi

[svlcse@tce.edu](mailto:svlcse@tce.edu)

<b>20CAPK0</b>	<b>SOFTWARE QUALITY AND TESTING</b>	Category	L	T	P	Credit
		PE	3	0	0	3

### Preamble

This course aims at facilitating the student to learn the best practices followed in industries to do effective software testing and quality management activities.

### Prerequisite

- 20CA210 : Software Engineering

### Course Outcomes

On the successful completion of the course, students will be able to

CO1:	Explain the basic concepts and the processes that lead to software quality and testing	Understand
CO2:	Design test cases from the given requirements using Black box testing techniques	Apply
CO3:	Identify the test cases from Source code by means of white box testing techniques	Apply
CO4:	Know about user acceptance testing and generate test cases for it	Analyze
CO5 :	Examine the test adequacy criteria to complete the testing process	Analyze
CO6:	Develop test cases and test suite using automated testing tools	Apply

### Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	P12
CO1	S	S	S	M	M	-	S	-	-	-	-	-
CO2	S	S	S	S	L	-	M	-	-	-	-	-
CO3	S	S	S	S	L	-	M	-	-	-	-	-
CO4	S	S	S	M	L	-	M	S	-	-	-	-
CO5	S	S	S	S	S	-	M	S	-	-	-	-
CO6	S	S	S	S	S	-	M	S				

S- Strong; M-Medium; L-Low



**Assessment Pattern**

Bloom's Category	Continuous Assessment Tests			Terminal Examination
	1	2	3	
Remember	20	20	20	20
Understand	20	20	20	30
Apply	40	40	20	30
Analyse	20	20	40	20
Evaluate	0	0	0	0
Create	0	0	0	0

**Course Level Assessment Questions****Understand the basic concepts and the processes that lead to software quality and testing (CO1):**

27. Define: Quality and Software testing.
28. What are the phases involved in PDCA life cycle in continuous quality improvement? Explain.
29. List the various types of products based on their criticality.
30. What are software quality audits?
31. Define: Error, Defect and Failure.
32. What are the challenges involved in developers becoming testers?
33. Discuss the factors affecting the cost of testing.

**Design test cases by analyzing the given requirements using Black box testing techniques (CO2):**

1. An application takes two inputs  $x$  and  $y$  where  $x \leq y$  and  $-5 \leq y \leq 4$ . Apply Equivalence partitioning to (i) Partition the input domain using uni-dimensional and multidimensional partitioning. (ii) Derive test sets based on the partitions created in (i)
2. Consider an 'Online Product Purchase System'. Apply Category Partition method in it to derive test cases.
3. Illustrate cause-effect graphing by constructing a CE-Graph and a Decision table for the following requirement: (8)

Consider a Income Tax Calculation System. The employee's income tax is calculated based on two constraints: (i) Gender (ii) Gross Pay / Annum

There are two genders : Male, Female

If the Gender is Male and the Gross Pay/Annum  $< 1,50,000$  then No Tax

If the Gender is Male and the Gross Pay/Annum  $> 1,50,000$  and  $< 2,50,000$  then 12% Tax

If the Gender is Male and the Gross Pay/Annum  $> 2,50,000$  and  $< 3,50,000$  then 18% Tax

If the Gender is Male and the Gross Pay/Annum  $> 3,50,000$  then 20% Tax

If the Gender is Female and the Gross Pay/Annum  $< 1,90,000$  then No Tax

If the Gender is Female and the Gross Pay/Annum  $> 1,90,000$  and  $< 3,00,000$  then 10% Tax

If the Gender is Male and the Gross Pay/Annum  $> 3,00,000$  and  $< 4,50,000$  then 12% Tax

If the Gender is Male and the Gross Pay/Annum  $> 4,50,000$  then 18% Tax.

4. Consider the "Withdraw amount" module in an ATM application. Analyze this scenario and generate test cases by using BVA.

**Identify the test cases from Source code by means of whitebox testing techniques (CO3):**

1. What is white box testing? Consider the following program code. Construct a control flow chart and a control flow graph for it and list down all the independent paths in it. Also, identify infeasible paths in it.

```
begin
    int x, y,z;
    input(x,y);
    if(x>0)  && (y>0)
        z=pow(x,y)
    else    if (x<0) && (y>0)
        z=pow((-x),y);
    else    if(x>0) && (y<0)
        z=pow(x,(-y));
    else if(x<0) && (y<0)
        z=0;
    while(y>0){ z+=1;y--;}
    if(z>0) {z-=1;}
```

**Know about user acceptance testing and generate test cases for it (CO4):**

1. What is alpha testing?
2. Define: Beta Testing.
3. Generate test cases from the UI design.

**Examine the test adequacy criteria to complete the testing process (CO5):**

1. Consider the following code and which test adequacy criteria are satisfied by the given test set : Test set T={t1:<x= -3, y= -2>, <x=2, y= -4>}

```
begin
    int x,y;
    if(x<0) && (y<0)
        printf("Both x and y are 0");
    else
        printf("x or y is 0 or non zero");
    end;
```

2. Consider the following code. Given that, the test suite T contains {n=3 for all test cases t1:<a=2,b=3,c=2>, t2:<a=1,b=2,c=-1>, t3:<a=-1,b=2,c=-4>}. Find out the MC/DC percentage of T by considering feasible and infeasible conditions in the code:

```
void main()
{
    int a, b, c,d, n;
    scanf("%d",&n);
    while(n>0)
    { scanf("%d,%d,%d", &a,&b,&c);
      if (a>b) && (b>c) d=a+b;
      else if(a>c) && (c>b)
          { d=a+c;
            if (c>a && (b>c)) d+=b;
          }
      else d=0;
      n-=1;
    }
    printf("Calculated Value is %d",d);
}
```

3. Given that, the test suite T contains {n=3 for all test cases t1:<a=2,b=3,c=2>, t2:<a=1,b=2,c=-1>, t3:<a=-1,b=2,c=-4>} find the code coverage %.

```
void main()
{
  int a, b, c,d, n;
  scanf("%d",&n);
  while(n>0)
  {
    scanf("%d,%d,%d", &a,&b,&c);
    if (a>b) && (b>c) d=a+b;
    else if(a>c) && (c>b)
      { d=a+c;
        if (c>a && (b>c)) d+=b;
      }
    else d=0;
    n-=1;
  }
  printf("Calculated Value is %d",d); }
```

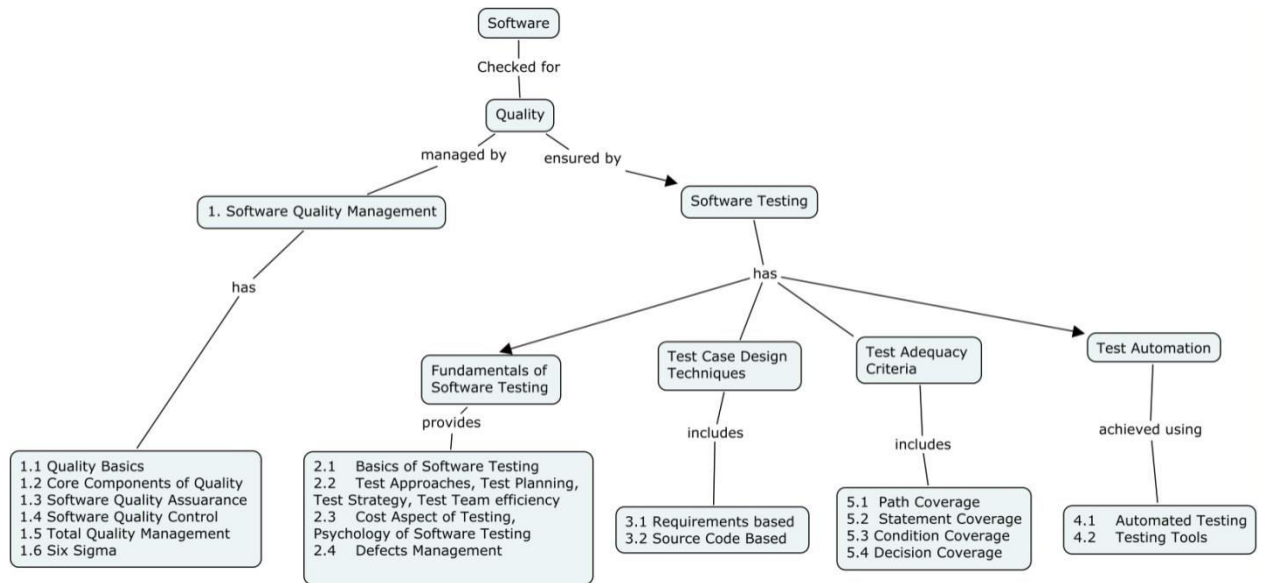
4. Calculate the block coverage and condition coverage for the Test set T={t1:<y= -3, n= -2>, <y=2, n= -4>, <y=2, n=2> }

```
main()
{
  int x,y,n,prod;
  read y,n;
  x=1; prod=y;
  while(n>0)
  {
    prod=x*y;
    display (x,y,prod);
    x+=1;
    n-=1;
  }
  if (prod<1) display("Error in input");
  else display("Final Value=",prod);
}
```

### Develop test cases and test suite using automated testing tools (CO6)

1. What is test automation?
2. What is a test script?
3. How unit test cases are generated using JUnit?
4. Generate test case using NUnit.
5. Generate at least two test cases one for positive and one for negative for "Login" screen that has userid and password as its components using JUnit.
6. How Selenium could be used to test a website?

## Concept Map



## Syllabus

**Software Quality Management** - Basics of Quality, Core Components of Quality, Software Quality Assurance, Software Quality Control, Total Quality Management, Six Sigma, **Fundamentals of Software Testing**- Basics of Software Testing, Test Approaches, Test Planning, Test Strategy, Defects Management, **Requirements based Test Case Design Techniques**- Equivalence Portioning, Boundary value analysis, Cause effect graphing, **Code Based Test Case Generation** - Cyclomatic Complexity - CFG Generation – Test Paths Generation - Test case generation from test paths **Test Adequacy Criteria** - Path Coverage, Statement Coverage, Condition Coverage, Decision Coverage, **Automated Software Testing** – Application of Automated Testing tools –JUnit/NUnit/PHPUnit/Selenium -

## Reference Books

1. Limaye M.G., "Software Testing Principles, Techniques and Tools", Second Reprint, TMH Publishers, 2010.
2. Aditya P.Mathur, "Foundations of Software Testing", 2<sup>nd</sup> Edition, Pearson Education, 2013.
3. Alan Gillies, "Software Quality Theory and Management", 2nd Edition, Cengage Learning Publishers, 2013 (Reprint)
4. Frank Appel, Testing with JUnit, 1st Edition, Packt Publishing Limited, 2015
5. Unmesh Gundecha, "Selenium Testing Tools Cookbook", 2nd Revised edition, Packt Publishing Limited, 2015
6. Kees Blokland, Jeroen Mengerink, Martin Pol, "Testing Cloud Services -How to Test SaaS, PaaS & IaaS", 1st Edition, Rocky Nook Publishers, O'Reilly Series, 2013
7. Srinivasan Desikan, Gopalswamy Ramesh, "Software Testing – Principles and Practices", 7<sup>th</sup> Reprint, Pearson Education, 2009.

**Course Contents and Lecture Schedule**

Module No.	Topic	No. of Lectures
<b>1</b>	<b>Software Quality Management</b>	
1.1	Basics of Quality	1
1.2	Core Components of Quality	1
1.3	Software Quality Assurance	1
1.5	Software Quality Control	1
1.6	Total Quality Management	1
1.7	Six Sigma	1
<b>2</b>	<b>Fundamentals of Software Testing</b>	
2.1	Basics of Software Testing	1
2.2	Test Approaches, Test Planning, Test Strategy,	2
2.4	Defects Management	1
<b>3</b>	<b>Requirements based Test Case Design Techniques</b>	
3.1	Requirements based test case generation introduction	2
3.2	Equivalence Class Portioning	2
3.3	Boundary value analysis	2
3.4	Cause effect graphing	2
3.5	Tutorial	
<b>4</b>	<b>Source Code Based Test Case Generation</b>	
4.1	CFG Creation	1
4.2	Cyclomatic Complexity	2
4.3	Test Paths Generation	1
4.4	Test Cases Generation	2
4.5	Tutorial	
<b>5</b>	<b>Test Adequacy Criteria</b>	
5.1	Path Coverage, Statement Coverage	2
5.2	Condition Coverage	2
5.3	Decision Coverage	2
	Tutorial	
<b>6.</b>	<b>Automated Software Testing</b>	
6.1	Automated Software Testing	2
6.2	JUnit /NUnit / PhUnit /Selenium	3
6.4	Tutorial	
	Total	36

**Course Designer:**

Dr. D. Anitha

[anithad@tce.edu](mailto:anithad@tce.edu)