B.E. COMPUTER SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENCE and MACHINE LEARNING) DEGREE PROGRAMME

CURRICULUM AND SYLLABI

FOR

FIRST SEMESTER to EIGTH SEMESTER

FOR THE STUDENTS ADMITTED IN THE

ACADEMIC YEAR 2024 - 25 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided Autonomous Institution affiliated to Anna University)
MADURAI – 625 015, TAMILNADU

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

Thiagarajar College of Engineering, Madurai - 625015
Credit Distribution for B.E.CSE (AIML) Programme – 2024 – 2028 Batch

S.No	Category of Courses	Credits	
		Regular Admission	**Lateral Entry Admission**
A	**Foundation Courses(FC)**	**54 - 66**	**22 - 36**
	Humanities and Social Sciences including Management Courses (HSMC)	09 - 12	06 – 09
	Basic Science Courses (BSC)	24 - 27	08 – 12
	Engineering Science Courses (ESC)	21 - 27	08 – 15
B	**Professional Core Courses (PCC)**	**55**	**44 - 49**
C	**Professional Elective Courses (PEC)**	**24 - 39**	**24 - 39**
	Programme Specific Electives (PSE)	15 – 24	15 – 24
	ProgrammeElectives for Expanded Scope (PEES)	9 - 15	9 - 15
D	**Open Elective Courses (OEC)**	**6 - 12**	**6 - 12**
	Interdisciplinary Elective (IE)	3 - 6	3 - 6
	Basic Science Elective (BSE)	3 - 6	3 - 6
E	**Project work**	**12**	**12**
F	Internship and Mandatory Audit as per the Regulatory Authorities	Non Credit (Not included for CGPA)	
	Minimum credits to be earned for the award of the Degree	**160 (from A to E) and the successful completion of Mandatory Courses**	**120 (from A to E) and the successful completion of Mandatory Courses**

Thiagarajar College of Engineering, Madurai-625015
Department of Computer science and Engineering
B.E. Computer Science and Engineering (Artificial Intelligence and Machine Learning) Programme
Scheduling of Courses – for those join in the year 2024 – 2025

Sem	Theory / Theory cum Practical / Practical										Credit
	1	2	3	4	5	6	7	8	9	Audit Courses (Mandatory Non- credit}	
I	22MA111 Calculus for Engineers (BS-4)	23PH120 Physics (BS-3)	24AM240 Introduction to Computer Systems (PC-3)	22EG140 Technical English (HSS-2)	24AM150 Engineering Exploration (ES-2)	24AM160 Problem Solving and Programming (PC-3) TCP	22EG170 English Laboratory (HSS-1)	22PH181 Physics Laboratory (BS-1)		22CHAA0 Environmental Science	19
II	24AM210 Linear Algebra (BS-4)	22CS320 Theory and Design of Programming Languages (ES-3)	24AM230 Cognitive Science (ES-3)	22CH130 Chemistry (BS-3)	24AM250 Object Oriented Programming (PC -3)	24CS260 Extended Reality: Principles and Practice (ES-3)(TCP)	24AM270 Object Oriented Programming Lab (PC-1)	22CH190 Chemistry Laboratory (BS-1)			21
III	24AM310 Probabilistic and Statistical Methods (BS-4)	24AM320 Mind and Brain Science (ES-3)	24AM330 Artificial Intelligence (PC-3)	22CS341 Data Structures and Algorithms (PC-3)		22CS361 Operating Systems (PC-4) TCP	22CS370 Data Structures Lab (PC-1)	24AM380 Artificial Intelligence Lab (PC-1)	22ES390 Design Thinking (ES-3)		22
IV	24AM410 Discrete Mathematics (BS-4)	22CS420 Design and Analysis of Algorithms (PC-3)	22CS430 Data Communication and Networks (PC-3)	24AM440 Machine Learning (PC-3)	24AM450 Database Systems (PC-3) TCP	22EG660 Professional Communication (HSS-2)	24AM470 Machine Learning Lab (PC-1)	22CS480 Algorithms Lab (PC-1)	22CS490 Project Management (HSS-3)	23CHAD0 Indian Constitution and Knowledge Systems	23
V	24AM510 Optimization Techniques (ES-3)	22CS520 Theory of Computation (ES-3)	24AM530 Deep Learning for NLP (PC-3)	24AMPX0 Programme Elective (PE - 3)	22XXGX0 Interdisciplinary Elective (OE-3)	24AM560 Reinforcement Learning (PC-3)	22CS570 Network Programming Lab (PC-1)	24AM580 Reinforcement Learning Lab (PC-1)	22CS590 Project – I (P-3)	23CHAE0 Universal Human Values and Ethics	23
VI	24AM610 Social and Ethical aspects of AI (HSS-3)	24AM620 Deep Learning for Computer Vision (PC-3)	24AM630 Human-AI Interaction (PC-3)	24AMPX0 Programme Elective (PE - 3)	22XXFX0 Basic Science Elective (OE-3)	24AMPX0 Programme Elective (PE - 3)	24AM670 Deep Learning Lab (PC-1)		22CS690 Project - II (P-3)		22
VII	24AM710 Emotional Artificial Intelligence (ES-2)	24AM720 Autonomous Agents (PC-3)	24AMPX0 Programme Elective (PE - 3)	24AMPX0 Programme Elective (PE - 3)	24AMPX0 Programme Elective (PE - 3)	24AMPX0 Programme Elective (PE - 3)	24AM770 Autonomous Agents Lab (PC-1)		22CS790 Project - III (P-3)		21
VIII	24AMPX0 Programme Elective (PE - 3)	24AMPX0 Programme Elective (PE - 3)							22CS890 Project - IV (P-3)		9

Passed in Board of Studies meeting on 27.04.2024

Approved in 67th Academic Council meeting on 25.05.2024

THIAGARAJAR COLLEGE OF ENGINEERING, MADURAI- 625 015
(A Govt. Aided, ISO 9001:2008 certified Autonomous Institution affiliated to Anna University)
Department of Computer Science and Engineering

B.E. Computer Science and Engineering (Artificial Intelligence and Machine Learning) Programme

Categorization of Courses

List of Humanities and Social Sciences including Management Courses (9-12)

- 22EG140 - Technical English (2)
- 22EG170 - English Laboratory (1)
- 22EG660 - Professional Communication (2)
- 22CS490 - Project Management (3)
- 24AM610 - Social and Ethical aspects of AI (3)

List of Basic Science Courses (24-27)

- 22MA111 - Calculus for Engineers (4)
- 24AM210 - Linear Algebra (4)
- 24AM310 - Probabilistic and Statistical Methods (4)
- 24AM410 - Discrete Mathematics (4)
- 23PH120 - Physics (3)
- 22CH130 - Chemistry (3)
- 22PH181 - Physics Laboratory (1)
- 22CH190 - Chemistry Laboratory (1)

List of Engineering Science Courses (21-27)

- 24AM150 - Engineering Exploration (2)
- 22CS320 – Theory and Design of Programming Languages (3)
- 24AM230 –Cognitive Science (3)
- 24CS260 - Extended Reality: Principles and Practice (3)
- 24AM320 –Mind and Brain Science (3)
- 22ES390 - Design Thinking (3)
- 24AM510 –Optimization Techniques (3)
- 22CS520 - Theory of Computation (3)
- 24AM710 - Emotional Artificial Intelligence (2)

List of Core Courses (55)

- 24AM160 - Problem solving and Programming (3) (TCP)
- 24AM240 - Introduction to Computer Systems (3)
- 24AM250 - Object Oriented Programming (3)
- 24AM270 - Object Oriented Programming Lab (1)
- 22CS341 - Data Structures and Algorithms (3)
- 24AM330 - Artificial Intelligence (3)
- 22CS361 - Operating Systems (4) (TCP)
- 22CS370 - Data Structures Lab (1)
- 24AM380 - Artificial Intelligence Lab (1)
- 22CS420 - Design and Analysis of Algorithms (3)

- 22CS430 – Data Communication and Networks (3)
- 24AM440 - Machine Learning (3)
- 24AM450 - Database Systems (3) (TCP)
- 24AM470 - Machine Learning Lab (1)
- 22CS480 - Algorithms Lab (1)
- 24AM530 - Deep Learning for NLP (3)
- 24AM560 - Reinforcement Learning (3)
- 22CS570 - Network Programming Lab (1)
- 24AM580 - Reinforcement Learning Lab (1)
- 24AM620 - Deep Learning for Computer Vision (3)
- 24AM630 - Human-AI Interaction (3)
- 24AM670 - Deep Learning Lab (1)
- 24AM720 - Autonomous Agents (3)
- 24AM770 - Autonomous Agents Lab (1)

Programme Elective Courses (24 – 39)

- Programme Electives (27)

Open Elective Courses (OEC): (6 – 12)

- Interdisciplinary Elective (3)
- Basic Science Elective (3)

Project (12)

- 22CS590 - Project – I (3)
- 22CS690 - Project - II (3)
- 22CS790 - Project – III (3)
- 22CS890 - Project – IV (3)

B.E. COMPUTER SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENCE and MACHINE LEARNING) DEGREE PROGRAMME

CURRICULUM AND SYLLABI

FOR

FIRST SEMESTER

FOR THE STUDENTS ADMITTED IN THE

ACADEMIC YEAR 2024 - 25 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided Autonomous Institution affiliated to Anna University)
MADURAI – 625 015, TAMILNADU

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

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
BECSE(AIML) Degree Programme

COURSES OF STUDY
(For the candidates admitted from 2024-25 onwards)

FIRST SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
22MA111	Calculus for Engineers	BSC	3	1	-	4
23PH120	Physics	BSC	3	-	-	3
22EG140	Technical English	HSMC	2	-	-	2
24AM150	Engineering Exploration	ESC	1	1	-	2
24AM240	Introduction to Computer Systems	PC	3	-	-	3
THEORY CUM PRACTICAL						
24AM160	Problem Solving and Programming	PC	2	-	2	3
PRACTICAL						
22EG170	English Laboratory	HSMC	-	-	2	1
22PH181	Physics Laboratory	BSC	-	-	2	1
AUDIT COURSE						
22CHAA0	Environmental Science	AC	-	-	-	-
22TAAA0	Heritage of Tamils		1	-	-	1
Total			**14**	**2**	**6**	**19**

BSC : Basic Science Courses
HSMC : Humanities and Social Sciences including Management Courses
ESC : Engineering Science Courses

L : Lecture
T : Tutorial
P : Practical

Note:
1 Hour Lecture is equivalent to 1 credit
1 Hour Tutorial is equivalent to 1 credit
2 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
BECSE(AIML) Degree Programme

SCHEME OF EXAMINATIONS
(For the candidates admitted from 2024-25 onwards)

FIRST SEMESTER

S.No.	Course 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
THEORY								
1	22MA111	Calculus for Engineers	3	40	60	100	27	50
2	23PH120	Physics	3	40	60	100	27	50
3	22EG140	Technical English	3	40	60	100	27	50
4	24AM150	Engineering Exploration	3	40	60	100	27	50
5	24AM240	Introduction to Computer Systems	3	40	60	100	27	50
THEORY CUM PRACTICAL								
6	24AM160	Problem Solving and Programming	3 (Terminal Exam Type : Theory)	50	50	100	22.5	50
PRACTICAL								
7	22EG170	English Laboratory	3	60	40	100	18	50
8	22PH181	Physics Laboratory	3	60	40	100	18	50

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

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

		Category	L	T	P	Credit
24AM150	**ENGINEERING EXPLORATION**	ES	1	1	-	2

Preamble

The course Engineering Exploration provides an introduction to the engineering field. It is designed to help the student to learn about engineering and how it affects our everyday lives. On the successful completion of the course, students will explain how engineering is different from science and technology and design basic combinational digital logic circuits.

Prerequisite

NIL

Course Outcomes

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

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency	Expected Attainment
CO1	Explain technological & engineering development, change and impacts of engineering	TPS2	70	70
CO2	Complete initial steps (Define a problem, list criteria and constraints, brainstorm potential solutions and document the ideas) in engineering design process	TPS3	70	70
CO3	Communicate possible solutions through drawings, modelling and Testing Final output	TPS3	70	70
CO4	Explain the working principle of semiconductor devices and its applications	TPS2	70	70
CO5	Perform simplification of boolean logic functions by applying the theorems and postulates of Boolean algebra and Karnaugh map and Perform code conversion and signed number arithmetic	TPS3	70	70
CO6	Design combinational logic circuits for given specifications and implement them using logic gates	TPS3	70	70

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1.	M	L	-	-	-	-	-	-	L	L	-	L	L	-	L
CO2.	S	M	L	-	-	L	L	L	L	L	-	L	M	L	L
CO3	S	M	L	-	-	L	L	L	L	L	-	L	M	L	L
CO4	M	L	-	-	-	-	-	-	L	L	-	L	L	-	L
CO5	S	M	L	-	-	-	-	L	M	M	-	L	M	-	L
CO6	S	M	L	-	-	-	-	L	M	M	-	L	M	-	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	Worksheet-1			Worksheet-2			CAT			Terminal (Theory)
TPS Scale	1	2	3	1	2	3	1	2	3	
CO1		10						5		10
CO2		5	10						10	20
CO3		5	20						10	20
CO4								5		10
CO5						25			10	20
CO6						25			10	20

Syllabus

What is Engineering: Engineering Requirement, Knowledge within Engineering disciplines, Engineering advancements

Engineering Design: Problem definition, idea generation through brainstorming and researching, solution creation through evaluating and communicating, text/analysis, final solution and design improvement.

Defining problems and Brainstorming: Researching design, sketching problem solving Communicating solution: Dimensioning orthographic drawing, perspective drawing

Modeling and Testing final output: Product evaluation, reverse engineering, final project report.

Electronics Engineering: History and overview- Diodes, Bipolar Junction Transistors - Integrated circuits -types-Applications -Storage cell using Transistors

Computer Engineering: Binary Number System :Binary Number system and its conversion to other number System-Complements, Signed Binary Numbers and Arithmetic- Boolean Algebra Theorems and Properties of Boolean Algebra-Digital Logic Gates and Logic Operations-Simplification of logic functions using Karnaugh Map Method.

Combinational Logic Circuits: Design and Implementation-Binary Adder, Subtractor and, decoder and encoder- Multiplexers and Demultiplexers

Text Books

1. Ryan A.Brown, Joshua W.Brown and Michael Berkihiser: “Engineering Fundamentals: Design, Principles, and Careers”, Goodheart-Willcox Publisher, Second Edition, 2014.
2. Malvino, A. P., & Bates, D. J. Electronic principles. McGraw-Hill/Higher Education,Ninth edition, 2020
3. M. Morris Mano, Micheal D. Ciletti, Digital Design: With an Introduction to the Verilog HDL, VHDL, and SystemVerilog, Pearson Education; Sixth edition,2018.

Course Contents and Lecture Schedule

Module	Topic	No. of Periods
1	**What is Engineering**	
1.1	Engineering Requirement, Knowledge within Engineering disciplines	1
1.2	Engineering advancements	1
2	**Engineering Design**	
2.1	Problem definition, idea generation through brainstorming and researching	1
2.2	solution creation through evaluating and communicating	1
2.3	text/analysis,	1
2.4	final solution and design improvement.	1
3	**Defining problems and Brainstorming**	
3.1	Researching design	1
3.2	sketching problem solving Communicating solution	1
3.3	Dimensioning orthographic drawing	1
3.4	perspective drawing	1
4	**Modeling and Testing final output**	
4.1	Product evaluation, reverse engineering	1
4.2	final project report.	1
5	**Electronics Engineering**	
5.1	History and overview	1
5.2	Diodes ,Bipolar Junction Transistors	**1**
5.3	Integrated circuits -types-Applications	1
5.4	Storage cell using Transistors	1
6	**Computer Engineering**:	
6.1	Binary Number System :Binary Number system and its conversion to other number System	1
6.2	number System-Complements	**1**
6.3	Signed Binary Numbers and Arithmetic	1
6.4	Boolean Algebra Theorems and Properties of Boolean Algebra	1
6.5	Digital Logic Gates and Logic Operations- Simplification of logic functions using Karnaugh Map	1
6.6	Combinational Logic Circuits: Design and Implementation-	1
6.7	Binary Adder, Subtractor decoder and encoder-	1
6.8	Multiplexers and Demultiplexers	1

Course Designer(s):

1. Dr.C.Senthil Kumar, Associate Professor/CSE cskcse@tce.edu

24AM240	INTRODUCTION TO COMPUTER SYSTEMS	Category	L	T	P	Credit
		PC	3	0	0	3

Preamble

This course provides an overview of computer systems, focusing on their architecture, components, operation, and interaction. Topics include hardware components, software systems, data representation, computer organization, Real-world applications and significance in modern computing

Prerequisite

-Nil-

Course Outcomes

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

CO	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Comprehend the basis operations that happen in a CPU	TPS 2	B	80
CO2	Sequence the data path and control path implementation	TPS 3	B	75
CO3	Customise code to support pipelining and improve performance	TPS 3	B	75
CO4	Apply knowledge of data representation and digital logic	.TPS 3	B	75
CO5	Categorize the memory hierarchy design and I/O design	TPS 3	B	75
CO6	Relate applications and algorithms for computing enhancements	TPS 3	B	75

Mapping with Programme Outcomes

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
C	M	L						L	L	L		L	L		L
CO2	S	M	L					L	L	L		L	M		L
CO3	S	M	L					L	L	L		L	M		L
CO4	S	M	L					L	L	L		L	M		L
CO5	S	M	L					L	L	L		L	M		L
CO.	S	M	L		L	L		L	L	L		L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
TPS Scale	**1**	**2**	**3**	**4**	**5**	**6**	**1**	**2**	**3**	**4**	**5**	**6**	**1**	**2**	**3**	**4**	**5**	**6**	**1**	**2**	**3**	**4**	**5**	**6**	**1**	**2**	**3**	**4**	**5**	**6**
CO1	10	10												20											5	5				
CO2		20	20												40											5	10			
CO3		20	20												40											5	10			
CO4							5	15	20												40					10	20			
CO5							10	10	20												40					5	10			
CO6							5	5	10												20					5	10			

Syllabus

Computer System Fundamentals

Motivation, IAS Computer Structure and Operation, Functional Units, Basic concepts

Computer Function and Interconnection

Top level view of components and functions, Instruction cycle and program execution, Interrupts and instruction cycles, Multiple interrupts, Interconnection structures, Bus interconnection, Multiple buses, Synchronous and asynchronous bus timings

Pipelining: Basic Concepts, Data Hazards, Instruction hazards, Influence on Instruction Sets, Data path and Control Considerations, Superscalar Operation, Performance Considerations.

Memory Design

Characteristics and hierarchy of memory, Cache memory principles and operation, Cache design and mapping functions, replacement algorithms, main memory, DRAM and SRAM, Types of ROMs, Module organization, Introduction to magnetic disks

I/O Systems

I/O transfer and disk performance, Interrupt driven and DMA transfers, Performance estimation and trade-offs in design.

Computer Arithmetic

Arithmetic and Logic Unit, Integer multiplication of unsigned and signed numbers, Booth's algorithm, Division of unsigned binary, Floating point arithmetic

GPU Design Principles

Need and opportunities, CPU, GPU, GPU Architecture – Overview of hardware and software technology, AI co processors, Spectral functions, Case studies on different architecture, evaluation with performance metrics, sustainability, energy efficiency

Text Book

1. William Stallings, Computer Organization and Architecture Designing for Performance, Eleventh edition, Prentice Hall, 2013.
2. Andrew S Tanenbaum and Todd Austin, Structured Computer Organization, Sixth edition, Pearson, 2013.
3. Carl Hamacher, Computer Organization and Embedded Systems, Sixth edition, McGrawHill, 2012.
4. Computer Organization and Design" by David A. Patterson and John L. Hennessy
5. Artificial Intelligence: A Modern Approach" by Stuart Russell and Peter Norvig.

Reference Books & web resources

1. DodiyaTripti, Computer Organisation and Advanced Microprocessors, First edition, Cengage Learning India,2012.

2. Barry B.Brey, The Intel Microprocessors Architecture Programming and Interfacing, Eighth edition, Pearson Prentice Hall, 2009.
3. N.Senthil Kumar, M.Saravanan and S. Jeevananthan, Microprocessors and Microcontrollers, First edition, Oxford University Press, 2010.
4. Charles H. Roth, Jr., “Fundamentals of Logic Design”, Jaico Publishing House, Mumbai,Fourth Edition 1992.
5. David A. Patterson and John L. Hennessy, “Computer Organization and Design: The
6. Hardware/Software Interface”, Second Edition.
7. https://www.spectroscopyonline.com/view/an-interview-with-ai-about-its-potential-role-in-vibrational-and-atomic-spectroscopy

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods	CO mapping
	Introduction to the course and course outcomes		
1	**Computer System Fundamentals (5)**		
1.1	Organisation and system architecture	1	CO1
1.2	Structures and functions	1	CO1
1.3	History and challenges evolution of computer and technology	1	CO1
1.4	IAS computer design and working	1	CO1
1.5	Measures of performance	1	CO1
2	**Computer Function and Interconnection (6)**		
2.1	Computer components and functions	1	CO2
2.2	Interconnection structures	1	CO2
2.3	Instruction cycle and program execution	1	CO2
2.4	Multiple buses	1	CO2
2.5	Synchronous and asynchronous bus timings	1	CO2
2.6	Interrupts and handling mechanism	1	CO2
3	**Pipelining (4)**		
3.1	Basic Concepts of pipelining	1	CO3
3.2	Hazards and its impact	1	CO3
3.3	Data path and Control Considerations	1	CO3
3.4	Superscalar Operations and performance Considerations	1	CO3
4	**Computer Arithmetic (5)**		
4.1	Arithmetic and Logic Unit	1	CO4
4.2	Integer multiplication of unsigned and signed numbers	1	CO4

Module No.	Topic	No. of Periods	CO mapping
4.3	Booth's algorithm	1	CO4
4.4	Division of unsigned binary	1	CO4
4.5	Floating point arithmetic	1	CO4
5	**MEMORY DESIGN (6)**		
5.1	Memory system overview and layout	1	CO5
5.2	Characteristics and hierarchy of memory	1	CO5
5.3	Cache memory principles and operation	1	CO5
5.4	Cache design its mapping functions and replacement algorithms	1	CO5
5.5	main memory, DRAM and SRAM	1	CO5
5.6	, Introduction to magnetic disks	1	CO5
6	**I/O Systems (5)**		
6.1	I/O transfer and disk performance	1	CO5
6.2	Programmed I/O	1	CO5
6.3	Interrupt driven I/O	1	CO5
6.4	DMA transfers	1	CO5
6.5	Performance estimation and trade-offs in design.	1	CO5
7	**GPU Design Principles (5)**		
7.1	Need and opportunities of CPU and GPU	1	CO6
7.2	GPU Architecture – Overview of hardware and software	1	CO6
7.3	Field programmable gates, AI co processors,	1	CO6
7.4	AI for Spectral functions, ML framework-Pytorch	1	
7.5	Case studies on different sustainable, energy efficiency system architecture and evaluation with performance metrics	1	CO6
	Total	36	

Course Designer(s):

1. K.NarasimhaMallikarjunan Associate Professor, CSE arjunkambaraj@tce.edu

24AM160	PROBLEM SOLVING AND PROGRAMMING	Category	L	T	P	Credit
		PC	2	0	2	3

Preamble

The objective of this course is to learn problem solving methodologies and aspects of C programming. Problem solving in programming helps the students to gain more knowledge over coding and programming. These problem solving skills also help them to develop more skills in a person and build a promising career.

Prerequisite

NIL

Course Outcomes

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

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Develop simple algorithms for arithmetic and logical problems.	TPS 2	B	85
CO2	Translate the algorithms to programs & execution	TPS 3	B	80
CO3	Implement conditional branching, iteration and recursion	TPS 3	B	80
CO4	Decompose a problem into functions and synthesize a complete program using divide and conquer approach	TPS 3	B	80
CO5	Implement arrays, pointers and structures to develop algorithms and programs	TPS 3	B	80
CO6	Design and Develop Solutions to problems using dynamic memory allocation and files	TPS 3	B	80

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
C(	M	L	-	-	L	-	-	L	L	L	L	L	L	-	L
CO2.	S	M	L	-	L	-	-	L	L	L	L	L	M	-	L
CO3.	S	M	L	-	L	-	-	L	L	L	L	L	M	-	L
CO4.	S	M	L	-	L	-	-	L	L	L	L	L	M	-	L
CO5.	S	M	L	-	L	-	-	L	L	L	L	L	M	-	L
CO6.	S	M	L	-	L	L	L	L	L	M	L	L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal(Theory)					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	10												20											5	5	-			
CO2		10	30												40											5	10			

CO3		10	30											40										5	15			
CO4							10	10	20											30				5	5	10			
CO5								10	20											30					5	15			
CO6								10	20											40					5	10			

***** Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to Programming: Introduction to components of a computer system- Idea of Algorithm - Representation of Algorithm: Flowchart - From algorithms to programs; source code, variables and memory locations- Syntax and Logical Errors in compilation-debugging and testing-object and executable code.

Control Structures: Arithmetic expressions and precedence- Conditional Branching and Loops- Arrays: Arrays (1-D, 2-D), Character arrays and Strings

Basic Algorithms: Searching- Basic Sorting Algorithms (Bubble and Selection)- Functions using built in libraries- Parameter passing in functions- call by value- Passing arrays to functions: idea of call by reference

Recursion: Recursion-a different way of solving problems -Example programs: Factorial, Fibonacci series, Towers of Hanoi. Structure: Structures, defining structures and Array of Structures. **Pointers**: Idea of pointers- Defining pointers, Use of Pointers in self-referential structures, notion of linked list (no implementation).

Dynamic Memory Allocation: Allocation of a Block of memory: malloc, allocating multiple blocks of memory: calloc, memory leak, releasing the used space: free, Altering the size of a block: realloc. Concept of Link list.

File Processing: Defining and Opening a file, closing a file, input/output operations on files, error handling during I/O operations, Command Line Arguments.

Text Book

1. Jeri R. Hanly, and Elliot B. Koffman, Problem Solving and Program Design in C, Pearson Education India, 2015.
2. Byron Gottfried, Programming with C, McGraw Hill International Edition, Fourth Edition, 2018.

Reference Books & web resources

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language 2e, Pearson Education India, 2015.
2. Behrouz A. Forouzan and Richard F. Gilberg, Computer Science: A Structured Programming Approach Using C, Course Technology Inc, Third Edition , 2006.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	**Introduction to Programming**	
1.1	Introduction to components of a computer system- Idea of Algorithm	1

Module No.	Topic	No. of Periods
1.2	Representation of Algorithm: Flowchart - From algorithms to programs	2
1.3	Source code, variables and memory locations- Syntax and Logical Errors in compilation- debugging and testing- object and executable code	1
2	**Control Structures**	
2.1	Arithmetic expressions and precedence	1
2.2	Conditional branching and loops	2
2.3	Arrays: Arrays (1-D, 2-D), Character arrays and Strings	2
3	**Basic Algorithms**	
3.1	Searching- Basic Sorting Algorithms (Bubble and Selection)	2
3.2	Functions: Functions using built in libraries- Parameter passing in functions- call by value- Passing arrays to functions: idea of call by reference	2
4	**Recursion**	
4.1	Recursion-a different way of solving problems - Example programs: Factorial, Fibonacci series, Towers of Hanoi	2
4.2	Structure: Structures, Defining structures and Array of Structures.	2
4.3	**Pointers**: Idea of pointers- Defining pointers, Use of Pointers in self-referential structures, notion of linked list (no implementation)	2
5	**Dynamic Memory Allocation**	
5.1	Allocation of a Block of memory: malloc, allocating multiple blocks of memory: calloc, memory leak - releasing the used space: free, Altering the size of a block: realloc. Concept of Link list.	2
6	**File Processing**	
6.1	Defining and Opening a file, closing a file, input/output operations on files,	1
6.2	Error handling during I/O operations, Command Line Arguments.	2
	Total	24

List of Experiments for Laboratory

1. Write a C program to generate the first n terms of the Fibonacci sequence.
2. Write a C program to generate prime numbers from 1 to n.
3. Write a C program to check whether given number is Armstrong Number or not.
4. Write a C program to perform arithmetic operations using switch statement.

5. Write a C program to find factorial of a given integer using recursive and non-recursive function.
6. Write C program to find GCD of two integers by using recursive function.
7. Write a C program to Sort the Array in an Ascending Order
8. Write a C program to find whether given matrix is symmetric or not.
9. Write a C program to perform matrix manipulation.
10. Develop a C Program for user defined Data Type: Structure
11. Illustrate C pre-processor directives through programming.
12. Simple Application Development

Course Designer(s):

1. Dr.M.P.Ramkumar,
   Associate Professor, CSE Department,
   ramkumar@tce.edu

B.E. COMPUTER SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENCE and MACHINE LEARNING) DEGREE PROGRAMME

CURRICULUM AND DETAILED SYLLABI

FOR

SECOND SEMESTER

FOR THE STUDENTS ADMITTED IN THE

ACADEMIC YEAR 2024 - 25 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided Autonomous Institution affiliated to Anna University)
MADURAI – 625 015, TAMILNADU

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

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
BECSE(AIML) Degree Programme

COURSES OF STUDY
(For the candidates admitted from 2024-25 onwards)

SECOND SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
24AM210	Linear Algebra	BSC	3	1	-	4
22CS320	Theory and Design of Programming Languages	ESC	3	-	-	3
24AM230	Cognitive Science	ESC	3	-	-	3
22CH130	Chemistry	BSC	3	-	-	3
24AM250	Object Oriented Programming	PC	3	-	-	3
THEORY CUM PRACTICAL						
24CS260	Extended Reality: Principles and Practice	ESC	2	-	2	3
PRACTICAL						
24AM270	Object Oriented Programming Lab	PC	-	-	2	1
22CH190	Chemistry Laboratory	BSC	-	-	2	1
			-			
Total			**17**	**1**	**6**	**21**

BSC : Basic Science Courses
ESC : Engineering Science Courses
PC : Professional Core Courses

L : Lecture
T : Tutorial
P : Practical

Note:
1 Hour Lecture is equivalent to 1 credit
1 Hour Tutorial is equivalent to 1 credit
2 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
BECSE(AIML) Degree Programme

SCHEME OF EXAMINATIONS
(For the candidates admitted from 2024-25 onwards)

SECOND SEMESTER

S.No.	Course 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
THEORY								
1	24AM210	Linear Algebra	3	40	60	100	27	50
2	22CS320	Theory and Design of Programming Languages	3	40	60	100	27	50
3	24AM230	Cognitive Science	3	40	60	100	27	50
4	22CH130	Chemistry	3	40	60	100	27	50
5	24AM250	Object Oriented Programming	3	40	60	100	27	50
THEORY CUM PRACTICAL								
6	24CS260	Extended Reality : Principles and Practice (Terminal Exam: Theory)	3	50	50	100	22.5	50
PRACTICAL								
7	24AM270	Object Oriented Programming Lab	3	60	40	100	18	50
8	22CH190	Chemistry Laboratory	3	60	40	100	18	50

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

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

		Category	L	T	P	Credit
24AM210	**LINEAR ALGEBRA**	BS	3	1	0	4

Preamble

Linear algebra and Matrices are essential tools for most algorithms in artificial intelligence and machine learning. This course introduces the idea of solving systems of linear and non-linear equations. Students learn to use the fundamental notion of vectors, vector space, linear independent, spanning, and basis. Also, it provides about linear transformations, the matrix of linear transformation, Eigen values, and Eigen vectors. They can perform approximations and orthogonal projections and construct an orthonormal basis for vector spaces. Moreover, this course demonstrates the various techniques for decomposition of matrices and principal component analysis.

Prerequisite

Nil

Course Outcomes

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

		TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Solve linear and non-linear systems of equations using direct and iterative methods.	TPS3	B	60
CO2	Demonstrate vector space and subspace, Identify the basis, row space, column space, null space, and dimension of a vector space.	TPS3	B	60
CO3	Compute an orthonormal basis of an inner product space for a given basis	TPS3	B	60
CO4	Apply matrix algebra techniques to transform a matrix into a diagonalizable matrix	TPS3	B	60
CO5	Decompose a matrix using LU, Singular Value Decomposition, and QR factorization method	TPS3	B	60
CO6	Perform dimensionality reduction on the given data using Principal Component Analysis (PCA)	TPS3	B	60

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO	S	M	L		M			L	M	L		M	M	L	L
CO2.	S	M	L		M			L	M	L		M	M	L	L
CO3.	S	M	L		M			L	M	L		M	M	L	L
CO4	S	M	L		M			L	M	L		M	M	L	L
CO5	S	M	L		M			L	M	L		M	M	L	L
CO6	S	M	L		M			L	M	L		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	Assessment 1		Assessment 2		Terminal (%)
	CAT 1 (%)	Assignment 1 (%)	CAT 2 (%)	Assignment 2 (%)	

TPS	1 10	2 30	3 60	1 -	2 -	3 100	1 10	2 30	3 60	1 -	2 -	3 100	1 -	2 30	3 70
CO1	-	10	17	-	-	25	-	-	-	-	-	-		3	8
CO2	7	10	20	-	-	38	-	-	-	-	-	-		6	12
CO3	3	10	23	-	-	37	-	-	-	-	-	-		6	12
CO4	-	-	-	-	-	-	10	20	20	-	-	50		12	14
CO5	-	-	-	-	-	-	-	5	20	-	-	25		-	12
CO6	-	-	-	-	-	-	-	5	20	-	-	25		3	12
TOTAL	10	30	60	-	-	100	10	30	60	-	-	100	-	30	70

Syllabus

System of linear Equations: System of linear equation - Row Echelon forms - Gauss Elimination. Nonlinear systems: System of Non-Linear Equations- Fixed point iteration- Newton Raphson Method-Gauss Jordan Method.

Vector Spaces: Definition and Examples- Subspaces, Linear Independence, Basis and Dimension, Row space and Column spaces.

Orthogonality: The scalar Product in R^n - Orthogonal Subspaces- Inner Product spaces, Orthonormal sets - Gram-Schmidt process.

Linear Transformations: Linear Transformations-Definitions and Examples - Matrix Representations of Linear Transformations -Similarity. Eigen systems: Eigen values of Matrices and Eigen vectors of Matrices–Diagonalization of Matrices – Quadratic Forms.

Decomposition and PCA: LU Decomposition, QR-Factorization- Singular Value Decomposition – Generalized Inverse (Pseudo Inverse)- Principal Component Analysis.

Text Books

1. Steven J. Leon., “Linear Algebra with Application” Ninth Edition, Pearson, 2015.
2. Steven C. Chapra & Raymond P. Canale., Numerical Methods for Engineers. 7th Edition, McGraw Hill Publications. 2015.
3. Gilbert Strang., “Introduction to Linear Algebra” 5th Edition, 2016

Reference Books & web resources

1. David C. Lay., “Linear Algebra and Its Applications” 5th Edition, 2015.
2. Richard Bronson., “Schaum’s Outline of Matrix Operations” Second Edition, McGraw Hill Publications. 2015.

Course Contents and Lecture Schedule

Module No.	Topic	No. of. Lectures
1	**LINEAR AND NONLINEAR SYSTEMS**	
1.1	System of linear equations, Row Echelon form	1
1.2	Gauss Elimination	1
1.3	System of Non-Linear Equations, Fixed point iteration	1
	Tutorial	1
1.4	Newton Raphson Method, Gauss Jordan Method	1
	Tutorial	1
	Program Using Python	
		(6)
2	**VECTOR SPACES**	
2.1	Vector space-Definition and Examples	2
2.2	Subspaces	1
	Tutorial	1

2.3	Linearly Independence	1
2.4	Basis and Dimension	1
	Tutorial	1
2.5	Row space and Column space	1
	Tutorial	1
	Program Using Python	
		(9)
3	**ORTHOGONALITY**	
3.1	The scalar Product in $\mathbf{R^n}$	1
3.2	Orthogonal Subspaces	1
	Tutorial	1
3.3	Inner Product spaces	1
3.4	Orthonormal sets	1
	Tutorial	1
3.5	Gram-Schmidt process	2
	Tutorial	1
	Program Using Python	
		(9)
4	**LINEAR TRANSFORMATION & EIGEN SYSTEMS**	
4.1	Linear Transformation- Definition and Examples	1
4.2	Matrix Representations of Linear Transformations	2
	Tutorial	1
4.3	Similarity Transformation	1
4.4	Eigenvalues of Matrices and Eigen vectors of Matrices	2
	Tutorial	1
4.5	Diagonalization of Matrices	2
4.6	Quadratic Forms	1
	Tutorial	1
	Program Using Python	
		(12)
5	**MATRIX DECOMPOSITION AND PCA**	
5.1	LU Decomposition	2
5.2	QR factorisation	1
	Tutorial	1
5.3	The Single Value Decomposition (SVD) of Matrices	2
	Tutorial	1
5.6	Pseudo Inverse	2
5.7	Principal Component Analysis	2
	Tutorial	1
	Program Using Python	
		(12)
	Total	**48**

Course Designer(s):

1. Dr. P. Victor pvmat@tce.edu
2. Dr. P. Krishnapriya pkamat@tce.edu
3. Dr. M. Sundar msrmat@tce.edu

24AM230 COGNITIVE SCIENCE

Category	L	T	P	Credit
ESC	3	0	0	3

Preamble

To introduce students with the fundamentals of cognitive science, i.e. the study of the mind! The course begins with the origins of the field and goes on to familiarize students with its interdisciplinary perspectives on how the mind is organized and processes information.

Prerequisite

Nil

Course Outcomes

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

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Cognize the basic concepts of Cognitive Science, Psychology, Nervous system and brain	TPS2	B	85
CO2	Explain the Brain and sensory motor information and the representation of sensory information	TPS2	B	85
CO3	Implement affordances in biological and artificial systems and also Cognitive Development.	TPS3	B	80
CO4	Develop a comprehensive understanding of how the human mind functions in both individual and social contexts	TPS3	B	80
CO5	Identify the role of perception and action in cognition, understanding how sensory input is processed by the brain and how this leads to behavioural responses.	TPS3	B	80
CO6	Determine the cognitive processes involved in learning and memory and their neural substrates.	TPS3	B	80

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L						L	L	L	L		M	L	L
CO2	M	L			M		M	L	L	L	L		M	L	L
CO3	S	M	L		M	M	M	L	L	L	L		M	L	L
CO4	S	M	L			M		L	L	L	L		M	L	L
CO5	S	M	L		M		M	L	L	L	L		M	L	L
CO6	S	M	L	L		M		L	L	L	L		M	L	L

S- Strong- M-Medium- L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	20												30											4	6				
CO2	10	20												30											4	6				
CO3	10	10	20												40										2	6	12			
CO4							5	10	15												30				2	6	12			
CO5							5	15	15												30				2	6	12			
CO6							5	15	15												40				2	6	12			

Syllabus

Introduction: Introduction to the study of cognitive sciences. A brief history of cognitive science. Methodological concerns in philosophy, artificial intelligence and psychology. Structure and constituents of the brain- Brief history of neuroscience- Mathematical models- Looking at brain signals- Processing of sensory information in the brain.

Neural Network Models: Neural Network Models- Processing of sensory information in the brain- motor and sensory areas- Brain Imaging, fMRI, MEG, PET, EEG- Multisensory integration in cortex- information fusion- from sensation to cognition, cybernetics- From physics to meaning- Analog vs. Digital: Code duality.

Linguistic Knowledge: What is language? Linguistic knowledge: Syntax, semantics, (and pragmatics)- Generative linguistics- Brain and language- Language disorders- Lateralization- The great past tense debate- Cognitivist and emergent standpoints- A robotic perspective.

Robotics: Affordances, direct perception, Ecological Psychology, affordance learning in robotics- Development, child and robotic development- Attention and related concepts- Human visual attention- Computational models of attention- Applications of computational models of attentional.

Machine Learning: Categories and Concepts - Concept learning- Logic- Machine learning- Constructing memories- Explicit vs. implicit memory- Information processing (three-boxes) model of memory- Sensory memory- Short term memory- Long term memory- Rationality- Bounded rationality- Prospect theory- Heuristics and biases- Reasoning in computers- Key points in social cognition- Context and social judgment- Schemas- Social signals.

Case studies:

1. Cognitive Design in the Development of a Mobile Banking App to improve user experience (UX)
2. Improving User Interface (UI) Design Using Cognitive Science Principles for an E-Commerce Website
3. DeepMind's AlphaGo: A case study on how AlphaGo, an AI program by DeepMind, uses cognitive modeling (such as reinforcement learning) to outperform human players in the game of Go.
4. Vision, Language, and Cognitive Control in Action Planning: Investigating how individuals use visual cues to plan and execute motor actions, particularly when language is involved in directing action.
5. The Apple iPhone: A case study on how Apple used cognitive principles (such as Fitts' Law and Gestalt principles) to design the iPhone’s touchscreen interface, focusing on ease of use and intuitive navigation.

Text Books

1. Don Norman, “The Design of Everyday Things”,Basic Books Publications,2014
2. Nir Eyal, “Hooked: How to Build Habit-Forming Products”, Penguin Books Limited,2014.

3. Jay Freidenberg and Gordon Silverman, Cognitive Science: An Introduction to the Study of Mind,SAGE, 2016.
4. Judith Hurwitz, Marcia Kaufman, Adrian Bowles, Cognitive Computing and Big Data Analytics, Wiley Publications, 2015.
5. Robert A. Wilson, Frank C. Keil, "The MIT Encyclopedia of the Cognitive Sciences",The MIT Press, 1999.
6.Jose Luis Bermúdez, Cognitive Science -An Introduction to the Science of the Mind, Cambridge University Press 2020.

References Books & web resources

1. Noah D. Goodman, Andreas Stuhlmuller, "The Design and Implementation of Probabilistic Programming Languages", Electronic version of book, https://dippl.org/.
2. Noah D. Goodman, Joshua B. Tenenbaum, The ProbMods Contributors, "Probabilistic Models of Cognition", Second Edition, 2016.
3. Vijay V Raghavan,Venkat N.Gudivada, VenuGovindaraju, C.R. Rao, Cognitive Computing: Theory and Applications: (Handbook of Statistics 35), Elsevier publications, 2016.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	**Introduction**	
1.1	Introduction to the study of cognitive sciences- A brief history of cognitive science	1
1.2	Methodological concerns in philosophy, artificial intelligence and psychology.	2
1.3	Structure and constituents of the brain- Brief history of neuroscience	1
1.4	Mathematical models- Looking at brain signals	1
1.5	Processing of sensory information in the brain	1
2	**Neural Network Models**	
2.1	Neural Network Models- Processing of sensory information in the brain	1
2.2	Motor and Sensory areas- Brain Imaging, fMRI, MEG, PET, EEG	2
2.3	Multisensory integration in cortex	1
2.4	Information fusion from sensation to cognition	1
2.5	Cybernetics, from physics to meaning.	1
2.6	Analog vs. Digital: Code duality.	1
3	**Linguistic Knowledge**	
3.1	What is language? Linguistic knowledge	1
3.2	Syntax, semantics and pragmatics	1
3.3	Generative linguistics- Brain and language	1
3.4	Language disorders- Lateralization	1
3.5	The great past tense debate	1
3.6	Cognitivist and emergent standpoints- A robotic perspective	1
4	**Robotics**	
4.1	Affordances, direct perception, Ecological Psychology	1

Module No.	Topic	No. of Periods
4.2	Affordance learning in robotics- Development, child and robotic development-	1
4.3	Attention and related concepts- Human visual attention	1
4.4	Computational models of attention	1
4.5	Applications of computational models of attentional.	1
5	**Machine Learning Categories and concepts**	
5.1	Concept learning- Logic - Machine learning	1
5.2	Constructing memories- Explicit vs. implicit memory	1
5.3	Information processing (three-boxes) model of memory	1
5.4	Sensory memory- Short term memory- Long term memory	1
5.5	Rationality- Bounded rationality- Prospect theory	1
5.6	Heuristics and biases- Reasoning in computers- Key points in social cognition	1
5.7	Context and social judgment- Schemas- Social signals.	1
6	**Case study**	
6.1	Cognitive Design in the Development of a Mobile Banking App to improve user experience (UX)	1
6.2	Improving User Interface (UI) Design Using Cognitive Science Principles for an E-Commerce Website	1
6.3	DeepMind's AlphaGo: A case study on how AlphaGo, an AI program by DeepMind, uses cognitive modeling (such as reinforcement learning) to outperform human players in the game of Go.	1
6.4	Vision, Language, and Cognitive Control in Action Planning: Investigating how individuals use visual cues to plan and execute motor actions, particularly when language is involved in directing action.	1
6.5	The Apple iPhone: A case study on how Apple used cognitive principles (such as Fitts' Law and Gestalt principles) to design the iPhone's touchscreen interface, focusing on ease of use and intuitive navigation.	1
	TOTAL	**36**

Course Designer(s):

1. Mrs.S.Jeniba,Assistant Professor- sjacse@tce.edu
2. Mrs.S.Saradha,Assistant Professor-ssacse@tce.edu

24AM250	OBJECT ORIENTED PROGRAMMING	Category	L	T	P	Credit
		PC	3	0	0	3

Preamble

This syllabus is intended for BECSE(AIML) students and enables them to learn Object Oriented Programming and the design of computer solutions in a precise manner. The syllabus emphasizes OOP concepts, Functions, Polymorphism, Inheritance, I/O and Patterns. The intention is to provide sufficient depth in these topics to enable students to apply Object Oriented approach to programming. The modules in the syllabus reflect solving general problems via programming solutions. Thus, modules collectively focus on programming concepts, strategies and techniques; and the application of these toward the development of programming solutions.

Prerequisite

Programming fundamentals

Course Outcomes

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

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Construct Object oriented programs using methods and passing arrays, objects, and array of objects to them.	TPS3	B	80
CO2	Demonstrate Compile-time and Run-time polymorphism using object oriented programs.	TPS3	B	80
CO3	Illustrate the relationships between objects using inheritance hierarchies and aggregation	TPS3	B	80
CO4	Develop Object Oriented programs to handle exceptions	TPS3	B	80
CO5	Develop Object Oriented programs to handle data using Java collections, Files and Object Serialization	TPS3	B	80
CO6	Develop Object Oriented programs to demonstrate event driven programming, concurrent programming.	TPS3	B	80

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L	L	L	-	-	L	L	L	L	L	M	L	L
CO2	S	M	L	L	L	-	-	L	L	L	L	L	M	L	L
CO3	S	M	L	L	L	-	-	L	L	L	L	L	M	L	L
CO4	S	M	L	L	L	-	-	L	L	L	L	L	M	L	L
CO5	S	M	L	L	L	-	-	L	L	L	L	L	M	L	L
CO6	S	M	L	L	L	-	-	L	L	L	L	L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment		Terminal					
													1	2						
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	3	3	1	2	3	4	5	6
CO1	10		10										20		2	4	10			
CO2	10	10	10										40		2	4	10			
CO3	10	20	20										40		4	4	10			
CO4							10		10					30	4	4	10			
CO5							10	10	10					30	4	4	10			
CO6							10	20	20					40	4		10			

***** Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Basics of Object Oriented Programming Object oriented programming and its benefits - Object oriented programming concepts: Encapsulation, Information hiding and Abstraction – Generalization/Specialization and Polymorphism - Object oriented design: finding the Classes and their Responsibilities – Object oriented programming language: Java and Python

Classes and Objects Instance fields and Methods-Constructors–Passing Arguments to a Method – Returning Value from a Method – Method overloading –Constructor overloading-Passing Arrays as Arguments to Methods – Passing Objects to Methods- Returning Objects from Methods

Class collaborations and Polymorphism Object Oriented Design: Class Collaborations – Aggregation –Composition –Chains of Inheritance – Overriding Super class methods – Abstract Classes and Abstract Methods – Interfaces

I/O Handling and Exception Handling – Binary files – Random-Access files- Object serialization – Exception handling

Collection Framework – Introduction to collections, Array List, Vector, Hash table, Stack, List, Tuple, Set, and Dictionary,

Event-Driven Programming Concurrent Programming Network programming – Text-related GUI components – other GUI components – Handling mouse events and button events – Thread life cycle and methods – Runnable interface – Thread Synchronization

Text Book

1. Herbert Schildt: “Java: The Complete Reference”, Twelfth Edition, McGraw-Hill, 2021.
2. Tony Gaddis, “Starting Out with Java: From Control Structures through Objects”, Sixth edition, Pearson Education Limited, 2016.
3. Bart Baesens, Aimee Backiel, SeppevandenBroucke, “Beginning Java Programming:The Object-Oriented Approach”, John Wiley & Sons, 2015.
4. Allen B. Downey, “Think Python: How to Think like a Computer Scientist”, 2nd Edition, O’Reilly Publishers, 2016. Grady Booch, Robert Maksimchuk, Michael Engel, Bobbi Young, Jim Conallen, Kelli Houston“ Object Oriented Analysis and Design with Applications”, Third Edition, 2012.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Hours
1	**Basics of Object oriented Programming (6)**	
1.1	Object oriented programming and its benefits	1
1.2	Object oriented programming concepts: Encapsulation, Information hiding and Abstraction, Generalization/Specialization and Polymorphism	1
1.3	Object oriented design: finding the Classes and their Responsibilities	2
1.4	Object oriented programming language: Java	1
1.5	Object oriented programming language: Python	1
2	**Classes and Objects (5)**	
2.1	Instance fields and Methods-Constructors	1
2.2	Passing Arguments to a Method – Returning Value from a Method - Method overloading	1
2.3	Constructor overloading	1
2.4	Passing Arrays As Arguments to Methods	1
2.5	Passing Objects to Methods, Returning Objects from Methods	1
3	**Class collaborations and Polymorphism (6)**	
3.1	Object oriented Design: Class Collaborations	1
3.2	Aggregation –Composition	1
3.3	Chains of Inheritance	1
3.4	Overriding Super class methods	1
3.5	Abstract Classes and Abstract Methods	1
3.6	Interfaces	1
4	**I/O Handling and Exception Handling (5)**	
4.1	Binary files	1
4.2	Random-Access files	1
4.3	Object serialization	1
4.4	Exception handling	2
5	**Collection Framework (7)**	
5.1	Introduction to collections	1
5.2	Array List, Vector	2
5.3	Hash table, Stack	2

5.4	List, Tuple, Set, and Dictionary	2
6	**Event-Driven Programming Concurrent Programming Network programming (7)**	
6.1	Frameworks	1
6.2	Text-related GUI components, other GUI components	1
6.3	Handling mouse events and button events	1
6.4	Thread life cycle and methods	1
6.5	Runnable interface	1
6.6	Thread Synchronization	1

Course Designer(s):

1. Dr M.Vijayalakshmi, Professor, CSE – mviji@tce.edu

24AM270	OBJECT ORIENTED PROGRAMMING LAB	Category	L	T	P	Credit
		PC	0	0	2	1

Preamble

This syllabus is intended for the BECSE(AIML) students and enables them to learn Object Oriented Programming and the design of computer solutions in a precise manner. The experiments emphasize on OOP concepts, Functions, Polymorphism, Inheritance, I/O, event- driven, concurrent and network programming. The intention is to provide sufficient depth in these topics to enable candidates to apply Object Oriented Programming approach to programming.

Prerequisite

Programming Fundamentals

Course Outcomes

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

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Construct Object oriented programs using methods and passing arrays, objects, and array of objects to them.	TPS3	B	80
CO2	Demonstrate Compile-time and Run-time polymorphism using object oriented programs.	TPS3	B	80
CO3	Illustrate the relationships between objects using inheritance hierarchies and aggregation	TPS3	B	80
CO4	Develop Object Oriented programs to handle exceptions	TPS3	B	80
CO5	Develop Object Oriented programs to handle data using Java collections, Files and Object Serialization	TPS3	B	80
CO6	Develop Object Oriented programs to demonstrate event driven programming, concurrent programming.	TPS3	B	80

Mapping with Programme Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L	M	M	M		M	M	M	M	M	M	M	M
CO2	S	M	L	M	M	M		M	M	M	M	M	M	M	M
CO3	S	M	L	M	M	M		M	M	M	M	M	M	M	M
CO4	S	M	L	M	M	M		M	M	M	M	M	M	M	M
CO5	S	M	L	M	M	M		M	M	M	M	M	M	M	M
CO6	S	M	L	M	M	M		M	M	M	M	M	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Model Examination	Terminal Examination
Remember		
Understand		
Apply	100	100
Analyse		
Evaluate		
Create		

Assessment Pattern: Psychomotor

	Experiment	CO
1.	Develop Object Oriented Program for passing arguments to a method and returning value from a method using Java and Python	CO1
2.	Develop Object Oriented Program for passing arrays and objects as arguments to method and returning objects from methods using Java and Python	CO1
3.	Construct Object Oriented Program for method overloading and constructor overloading using Java and Python	CO2
4.	Demonstrate aggregation and composition using object-oriented program using Java and Python	CO2
5.	Develop Object Oriented Program to demonstrate inheritance and overriding super class methods in Java and Python	CO3

6.	Develop Object Oriented Program to demonstrate abstract base classes abstract methods in Java and Python	CO3
7.	Construct Object Oriented Program to demonstrate exception handling in Java and Python	CO4
8.	Construct Object Oriented Program in Java and Python to demonstrate File handling and Object Serialization	CO5
9.	Develop Object Oriented Program for manipulation of data using Collections in Java and Python	CO5
10.	Develop event-driven programs using Java and Python	CO6
11.	Develop concurrent programs using Java and Python	CO6

Learning Resources

1. Herbert Schildt: “Java: The Complete Reference”, Twelfth Edition, McGraw-Hill, 2021.
2. Tony Gaddis, “Starting Out with Java: From Control Structures through Objects”, Sixth edition, Pearson Education Limited, 2016.
3. Bart Baesens, Aimee Backiel, SeppevandenBroucke, “Beginning Java Programming:The Object-Oriented Approach”, John Wiley & Sons, 2015.
4. Allen B. Downey, “Think Python: How to Think like a Computer Scientist”, 2nd Edition, O’Reilly Publishers, 2016. Grady Booch, Robert Maksimchuk, Michael Engel, Bobbi Young, Jim Conallen, Kelli Houston“ Object Oriented Analysis and Design with Applications”, Third Edition, 2012.

Course Designers:

1. Dr M.Vijayalakshmi, Professor, CSE – mviji@tce.edu

B.E. COMPUTER SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENCE and MACHINE LEARNING) DEGREE PROGRAMME

CURRICULUM AND DETAILED SYLLABI

FOR

THIRD SEMESTER

FOR THE STUDENTS ADMITTED IN THE

ACADEMIC YEAR 2024 - 25 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided Autonomous Institution affiliated to Anna University)
MADURAI – 625 015, TAMILNADU

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

Passed in Board of Studies meeting on 09.11.2024 Approved in 68th Academic Council meeting on 21.12.2024

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
BECSE(AIML) Degree Programme

COURSES OF STUDY
(For the candidates admitted from 2024-25 onwards)

THIRD SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
24AM310	Probabilistic and Statistical Methods	BSC	3	1	-	4
24AM320	Mind and Brain Science	ESC	3	-	-	3
24AM330	Artificial Intelligence	PC	3	-	-	3
22CS340	Data Structures and Algorithms	PC	3	-	-	3
THEORY CUM PRACTICAL						
22CS361	Operating Systems	PC	3	-	2	4
PRACTICAL						
22CS370	Data Structures Lab	PC	-	-	2	1
24AM380	Artificial Intelligence Lab	PC	-	-	2	1
22ES390	Design Thinking	ESC				3
Total			**15**	**1**	**6**	**22**

BSC : Basic Science Courses
ESC : Engineering Science Courses
PC : Professional Core Courses

L : Lecture
T : Tutorial
P : Practical

Note:
1 Hour Lecture is equivalent to 1 credit
1 Hour Tutorial is equivalent to 1 credit
2 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
BECSE(AIML) Degree Programme

SCHEME OF EXAMINATIONS
(For the candidates admitted from 2024-25 onwards)

THIRD SEMESTER

S.No.	Course 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
THEORY								
1	24AM310	Probabilistic and Statistical Methods	3	40	60	100	27	50
2	24AM320	Mind and Brain Science	3	40	60	100	27	50
3	24AM330	Artificial Intelligence	3	40	60	100	27	50
4	22CS340	Data Structures and Algorithms	3	40	60	100	27	50
5			3	40	60	100	27	50
THEORY CUM PRACTICAL								
6	22CS361	Operating Systems	3 (Terminal Exam Type: Theory)	50	50	100	22.5	50
PRACTICAL								
7	22CS370	Data Structures Lab	3	60	40	100	18	50
8	24AM380	Artificial Intelligence Lab	3	60	40	100	18	50
9.	22ES390	Design Thinking						

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

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

		Category	L	T	P	Credit
24AM310	**PROBABILISTIC AND STATISTICAL METHEODS**	BSC	3	1	0	4

Preamble

This course provides AI and ML students with foundational skills in probability and statistics, essential for modelling and interpreting data. Key topics include probability concepts, random variables and distributions (Binomial, Poisson, Gaussian), and estimation methods (Central Limit Theorem, point estimation). Hypothesis testing and ANOVA techniques, correlation, regression, time series analysis and stochastic processes are also covered, enabling students to assess model performance and optimize experimental designs. This statistical toolkit supports effective, data-driven decision-making in AI and ML applications.

Prerequisite

Nil

Course Outcomes

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

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Apply the concept of probability, conditional probability to solve problems in data analysis	TPS3	B	85
CO2	Solve the technical problems using appropriate probability distributions and estimate the unknown parameters	TPS3	B	85
CO3	Apply the test of hypothesis for decision making based on observed and experimental data	TPS3	B	85
CO4	Design and conduct of experiments involving a single factor, two factors and three factors	TPS3	B	85
CO5	Apply the concepts of correlation, regression, and time series to feature selection, predictive analysis and data handling in machine learning	TPS3	B	85
CO6	Apply the concept of stochastic processes to solve problems in AI applications.	TPS3	B	85

Mapping with Programme Outcomes

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO1**	**PSO2**	**PSO3**
CO3.	S	M	L		L			L	L	L	M	L	M	L	L
CO4.	S	M	L		L			L	L	L	M	L	M	L	L
CO5.	S	M	L		L			L	L	L	M	L	M	L	L
CO4	S	M	L		L			L	L	L	M	L	M	L	L

CO5	S	M	L		L			L	L	L	M	L	M	L	L
CO6	S	M	L		L			L	L	L	M	L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	Assessment 1						Assessment 2						Terminal (%)		
	CAT 1			Assignment 1			CAT 2			Assignment 2					
TPS	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
CO1	3	10	17			20							-	6	8
CO2	3	10	20			40							-	6	10
CO3	4	10	23			40							-	6	12
CO4							3	5	13			20	-	-	12
CO5							3	15	23			40	-	6	14
CO6							4	10	24			40	-	6	14
TOTAL	10	30	60			100	10	30	60			100	-	30	70

*****Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Random Variables and Their Distributions: Introduction to Probability, random variables (discrete and continuous), probability density and mass functions, distribution functions, mean and variance, Moment generating and Characteristic functions, Joint probability distribution and conditional distribution, special distributions (Binomial, Poisson, Exponential, Normal), Central limit theorem, Introduction to parameter estimation and characteristics of estimators, Maximum likelihood estimation method, Method of moments.

Testing of Hypothesis: Testing of Hypothesis, Null and alternative hypothesis, level of significance, one-tailed and two-tailed tests, tests for large samples (tests for single mean, difference of means), tests for small samples (t-test for single mean and difference of means, F-test for comparison of variances), Chi-square test for goodness of fit, analysis of variance: Single factor ANOVA, two factor ANOVA and Latin square design.

Correlation and Basics of Time Series: Correlation and regression - Multiple and partial correlation - rank correlation and Karl Pearson coefficient of correlation - lines of regression. Components of a time series, Moving averages, exponential smoothing

Stochastic Process: Definitions and classifications - Markov processes and Markov chains: Markov Chains - Matrix approach - Steady-state distribution. Counting processes: Binomial process - Poisson process - Simulation of stochastic processes

Text Books

1. Jay L. Devore, “Probability and Statistics for Engineering and the Sciences” (English) 9th Edition, Cengage Learning India Pvt Ltd, New Delhi, 2012.

2. Douglas C. Montgomery, George C. Runger, “Applied Statistics and Probability for Engineers”, 7th Edition, Wiley, 2018.
3. Michael Baron, “Probability and Statistics for Computer Scientists”, 3rd Edition, Chapman & Hall, 2019.

4. Hogg, McKean, Craig, “Introduction to Mathematical Statistics”, 8th Edition, Pearson, 2019.
5. Paul Newbold, William L. Carlson, Betty M. Thorne “Statistics for Business and Economics”, 9th Edition, Pearson Education, USA, 2020.

Reference Books& web resources

1. Ronald E. Walpole, Sharon L. Myers, Keying Ye, “Probability & Statistics for Engineers and Scientists”, 9th Edition, Pearson Education, New Delhi, 2012.

2. Mendenhall, William, Robert J. Beaver, Barbara M. Beaver., “Introduction to probability and Statistics” Fifteenth Edition, 2019.
3. Sheldon M. Ross, “Introduction to Probability and Statistics for Engineers and Scientists Fifthedition,2014

Course Contents and Lecture Schedule

Module No.	Topic	No. of. Lectures	
1	**Introduction to Probability and Random Variables**		
1.1	Introduction to Probability, random variables (discrete and continuous)	1	CO1
1.2	Probability density and mass functions, distribution functions	1	CO1
	Tutorial	1	CO1
1.3	Mean and variance	1	CO1
1.4	Moment generating and Characteristic functions	1	CO1
1.5	Joint probability distribution and conditional distribution	1	CO1
	Tutorial	1	CO1
2	**Distributions and Method of Estimations**		
2.1	Special distributions (Binomial, Poisson, Exponential, Normal)	2	CO2
2.2	Central limit theorem	1	CO2
	Tutorial	1	CO2
2.3	Introduction to parameter estimation and characteristics of estimators	1	CO2
2.4	Maximum likelihood estimation method	1	CO2
2.5	Method of moments	1	CO2
	Tutorial	1	CO2
3.	**Testing of Hypothesis**		

3.1	Testing of Hypothesis, Null and alternative hypothesis, level of significance, one-tailed and two-tailed tests	1	CO3
3.2	Tests for large samples (tests for single mean, difference of means, single proportion, difference of proportions)	2	CO3
	Tutorial	1	CO3
3.3	Tests for small samples (t-test for single mean and difference of means, paired t-test)	2	CO3
3.4	F-test for comparison of variances	1	CO3
3.5	Chi-square test for goodness of fit	1	CO3
	Tutorial	1	CO3
4.	**Analysis of Variance**		
4.1	Single Factor ANOVA	2	CO4
4.2	Two Factor ANOVA	1	CO4
4.3	Latin Square Design	1	CO4
	Tutorial	1	CO4
5.	**Correlation and Basics of Time Series**		
5.1	Correlation and regression	1	CO5
5.2	Multiple and partial correlation	2	CO5
	Tutorial	1	CO5
5.3	Rank correlation and Karl Pearson coefficient of correlation	1	CO5
5.4	Lines of regression	1	CO5
	Tutorial	1	CO5
5.5	Components of a time series	1	CO5
5.6	Moving averages and Exponential smoothing	1	CO5
	Tutorial	1	CO5
6.	**Stochastic Process**		
6.1	Definitions and classifications	1	CO6
6.2	Markov processes and Markov chains	1	CO6
6.3	Markov Chains - Matrix approach	1	CO6
6.4	Steady-state distribution	1	CO6

	Tutorial	1	CO6
6.5	Counting processes: Binomial process and Poisson process	2	CO6
6.6	Simulation of stochastic processes	1	CO6
	Tutorial	1	CO6
	Total	48	

Course Designer(s):

1. Dr. C. S. Senthilkumar kumarstays@tce.edu
2. Dr. P. Victor pvmat@tce.edu
3. Dr. P. Jeyadurga pjdmat@tce.edu

		Category	L	T	P	Credit
24AM320	**MIND AND BRAIN SCIENCE**	ES	3	0	0	3

Preamble

This course will introduce the core perceptual and cognitive abilities of the human mind and how they are implemented in the brain. The learner's will explore distinct domains of cognition like face recognition, navigation, number, language, music, and social cognition, and the cortical regions and networks that implement these functions. The course also emphasizes the inferences that can (and cannot) be drawn from each of the main methods in human cognitive neuroscience.

Prerequisite

NIL

Course Outcomes

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

CO	Course Outcome 1 (CO1)	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Infer Neuroanatomy of the brain and its cortical area for visual motion information	TPS 2	B	80
CO2	Articulate computation theory behind Color Vision and Face Recognition	TPS 3	B	75
CO3	Determine memory and audition in Number, Hearing, Speech and Music Processing	TPS 3	B	75
CO4	Examine fundamental problems of navigation and Reorienting process	TPS 3	B	75
CO5	Execute Languages in computers and the human mind with moral reasoning	TPS 3	B	75
CO6	Experiment the causes of Brain network in attention and awareness mechanism	TPS 3	B	75

Mapping with Programme Outcomes

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
CO	M	L				L		M	M	M		M	L	L	M
CO2.	S	M	L			L		M	M	M		M	M	L	M
CO3.	S	M	L			L		M	M	M		M	M	L	M
CO4.	S	M	L			L		M	M	M		M	M	L	M
CO5	S	M	L			L		M	M	M		M	M	L	M
CO6	S	M	L			L		M	M	M		M	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	**CAT1**						**CAT2**						**Assignment 1**						**Assignment 2**						**Terminal**					
TPS Scale	**1**	**2**	**3**	**4**	**5**	**6**	**1**	**2**	**3**	**4**	**5**	**6**	**1**	**2**	**3**	**4**	**5**	**6**	**1**	**2**	**3**	**4**	**5**	**6**	**1**	**2**	**3**	**4**	**5**	**6**

CO1	10	10												20									6	6				
CO2	10	10	20											40									2	5	10			
CO3	10	10	20											40									2	5	10			
CO4							5	5	20											30			3	5	10			
CO5							5	5	25											35			3	5	10			
CO6							5	5	25											35			3	5	10			

***** Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Neuroanatomy: Study of Human Brain-Major component of Human Brain-Receptive Field-Visual Motion information-Criteria for cortical area-case study: visual motion area MT (V5)

Neuroscience Reasoning Methods: What is Mind? -Computational Theory: Color, Face Recognition -Strength and Weakness of Behavioral Methods - Case Study: Color Vision-fMRI, EEG, MEG

Memory and Audition : Basics of Brain Development : Grey and White Matters -Origin of Knowledge-Concept of Number and Quantity - Number Sense - Hearing, Reverberation - Speech Perception - Auditory Processing Pathway - Music Perception - Selective Components

Navigation: Feats of Animal Navigation - Fundamental Problems of Navigation - Cognitive Map - Place, Direction, Grid Cells – Reorienting

Language and Theory of Mind: Language and Cognition - Semantics vs., syntax - Percept, Belief, Desire - Thinking about Thoughts - Moral Reasoning

Brain Networks: Diffusion imaging and Tractography - Resting Functional Correlations- Crux of Attention - Brain Mechanism of Attention - Neural correlates of awareness

Text Books

1. E. Bruce Goldstein, “The Mind: Consciousness, Prediction, and the Brain “. The MIT Press, 2020.
2. Paul Thagard, “Brain-Mind: From Neurons to Consciousness and Creativity”, Oxford University Press, 2019.
3. Smith, C. U. M.;Whitaker, Harry, “Brain, Mind and Consciousness in the History of Neuroscience “, Springer 2014.
4. Robert L. Solso, “Mind and Brain Sciences in the 21st Century”, The MIT Press., 1999

Reference Books & web resources

1. Leonard Talmy, “The Targeting System of Language”, The MIT Press, 2018.
2. Fischer, J., J.G. Mikhael, et al. “Functional Neuroanatomy of Intuitive Physical Inference.” *PNAS* 113, no. 34 (2016): E5072–E5081. DOI: /10.1073/pnas.161034411
3. Vilayanur Ramachandran, The Emerging Mind, Profile Books; Main edition, 2003.
4. Downing, P.E., Y. Jiang, et al. “A Cortical Area Selective for Visual Processing of the Human Body.” *Science* 293 (2001): 2470–73. DOI: 0.1126/science.1063414

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	**Neuroanatomy (5)**	
1.1	Study of Human Brain- Major component of Human Brain	1
1.2	Receptive Field	1
1.3	Visual Motion information	1
1.4	Criteria for cortical area	1
1.5	case study: visual motion area MT(V5)	1
2	**Neuroscience Reasoning Methods (5)**	
2.1	What is Mind?	1
2.2	Computational Theory: Color, Face Recognition	2
2.3	Strength and Weakness of Behavioural Methods	1
2.4	Case Study: Color Vision-fMRI, EEG, MEG	1
3	**Memory and Audition (10)**	
3.1	Basics of Brain Development : Grey and White Matters	1
3.2	Origin of Knowledge	1
3.3	Concept of Number and Quantity - Number Sense	2
3.4	Hearing, Reverberation	2
3.5	Speech Perception -Auditory Processing Pathway	2
3.6	Music Perception - Selective Components	2
4	**Navigation (5)**	
4.1	Feats of Animal Navigation	1
4.2	Fundamental Problems of Navigation	1
4.3	Cognitive Map	1
4.4	Place, Direction, Grid Cells	1
4.5	Reorienting	1
5	**Language and Theory of Mind (5)**	
5.1	Language and Cognition	1

Module No.	Topic	No. of Periods
5.2	Semantics vs, syntax	1
5.3	Percept, Belief, Desire	1
5.4	Thinking about Thoughts	1
5.5	Moral Reasoning	1
6	**Brain Networks (5)**	
6.1	Diffusion imaging and Tractography	1
6.2	Resting Functional Correlations	1
6.3	Crux of Attention	1
6.4	Brain Mechanism of Attention	1
6.5	Neural correlates of awareness	1
	Total	36

Course Designer(s):

1. Dr.R.Leena Sri, Associate Professor, Dept. of CSE, rlsit@tce.edu

		Category	L	T	P	Credit
24AM330	**ARTIFICIAL INTELLIGENCE**	PC	3	0	0	3

Preamble

This course introduces the fundamentals of problem solving using Artificial Intelligence techniques. Artificial intelligence is the sub-area of computer science that deals with the algorithms and reasoning to make computers think like human beings. Students will be able to learn the techniques used in knowledge representation, search strategies and inferencing. This course will help the students to gain generic problem-solving skills that have applicability to a wide range of real-world problems.

Prerequisite

Students are expected to have

- Basic Programming knowledge in Python
- Knowledge of search algorithms
- Graph data structures

Course Outcomes

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

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Formulate the suitable characteristics of Intelligent agents for problem solving	TPS3	B	75
CO2	Develop AI problems as a state space representation and solve them by applying the appropriate search techniques	TPS3	B	75
CO3	Construct logical agents to do inference using first-order logic	TPS3	B	75
CO4	Construct an agent that can use the various ways of planning and acting in the real world	TPS3	B	75
CO5	Build solutions that can reason under uncertainty and performs inference in a Bayesian Network	TPS3	B	75
CO6	Analyze the various search techniques behind the single agent and multi agent problems.	TPS4	B	75

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
C	S	M	L		L			M	L	L		M	M	M	L
CO2	S	M	L		L			M	L	L		M	M	M	L
CO 3	S	M	L		L			M	L	L		M	M	M	L
CO 4	S	M	L		L			M	L	L		M	M	M	L
CO 5.	S	M	L		L			M	L	L		M	M	M	L
CO 6	S	S	M	L	M	L		M	L	L		M	S	M	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT 1						Assignment 1			CAT 2						Assignment 2				Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	1	2	3	4	5	6	1	2	3	4	1	2	3	4	5	6
CO1	10	10	15	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-		2	5		-	-	-
CO2	10	10	15	-	-	-	-	-	50	-	-	-	-	-	-	-	-	-		2	5	10	-	-	-
CO3	-	10	20	-	-	-	-	-	50	-	-	-	-	-	-	-	-	-		2	5	10	-	-	-
CO4	-	-	-	-	-	-	-	-	-	10	10	15	-	-	-	-	-	30		2	5	10	-	-	-
CO5	-	-	-	-	-	-	-	-	-	10	10	15	-	-	-	-	-	30		2	5	10	-	-	-
CO6	-	-	-	-	-	-	-	-	-	10	-	-	20	-	-	-	-	-	40		5		20		

*****Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction

The Turing Test, Rationality, Foundation of Artificial Intelligence, Need, Applications, State of the art, Future of Artificial Intelligence (AI) - Characteristics of Intelligent Agents –Environments, Rationality - The Nature of Environments – Properties - Types of Agents - Structure of Agents – How the components of agent programs work

Problem Solving Methods

Formulating Problems, Problem Solving Approach to Typical AI problems- Search Strategies – Uninformed Search– Breadth–first search, Depth-first search - Searching with Partial Observations- Informed Search - Greedy Best First search - A* Search, Local Search Algorithms and Optimization Problems– Measure of Performance and analysis of search algorithms.

Knowledge Representation

Game Playing- The minimax algorithm, Perfect Decisions- Imperfect Decisions- State space tree- Alpha-Beta pruning, Constraint Satisfaction Problem – Definition, Constraint Propagation, Search for CSPs , Inference In First Order Logic: Propositional Logic vs. First-Order Inference - Unification and First-Order Inference- Forward Chaining - Backward Chaining - Resolution.

Planning

Planning with state-space search –Classical Planning- Algorithms for Classical Planning—Heuristics for Planning- Partial-order Planning –Hierarchical Planning

Uncertain Knowledge and Reasoning

Uncertainty - Inference Using Full Joint Distributions – Independence – Bayes Rule, and its Use - Probabilistic Reasoning –Representing Knowledge in an Uncertain Domain, The Semantics of Bayesian Networks - ExactInferences in Bayesian networks- Inference in Temporal models – Most likely Sequence– Viterbi Algorithm—Application of Viterbi Algorithm in Natural Language Processing

Text Book

1. Stuart Russell and Peter Norvig, “Artificial Intelligence – A Modern Approach”, Fourth Edition, Pearson Education, 2021.

Reference Books& web resources

1. Dan W. Patterson, “Introduction to AI and ES”, Pearson Education,2007
2. Kevin Night, Elaine Rich, and Nair B., “Artificial Intelligence”, McGraw Hill, 2008
3. Patrick H. Winston, "Artificial Intelligence", Third Edition, Pearson Education, 2006
4. Deepak Khemani, “Artificial Intelligence”, Tata McGraw Hill Education, 2013.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	**Introduction**	
1.1	Introduction to AI - Definition, Need, Applications, State of Art	1
1.2	Future of Artificial Intelligence	1
1.3	Characteristics of Intelligent Agents Environments, Rationality - The Nature of Environments – Properties	1
1.4	Types of Agents- Structure of Agents	1
1.5	How the components of agent programs work	1
2	**Problem Solving Methods**	
2.1	Formulating Problems - Problem Solving Approach to Typical AI problems	1
2.2	Search Strategies - Uninformed Search– Breadth–first search, Depth-first search	2

Module No.	Topic	No. of Periods
2.3	Searching with Partial Observations	2
2.4	Informed Search - Greedy Best First Search	2
2.5	A* Search	1
2.6	Local Search Algorithms and Optimization Problems	1
2.7	Measure of Performance and analysis of search algorithms	1
3	**Knowledge Representation**	
3.1	Game Playing- The minimax algorithm, Perfect Decisions- Imperfect Decisions- State space tree- Alpha-Beta pruning	2
3.2	Constraint Satisfaction Problem – Definition, Constraint Propagation, Search for CSPs	1
3.3	Inference In First-Order Logic: Propositional Logic vs. First-Order Inference -	1
3.4	Unification and First-Order Inference	1
3.5	Forward Chaining - Backward Chaining	2
3.6	Resolution	1
4	**Planning**	
4.1	Planning with state-space search	1
4.2	Classical Planning- Algorithms for Classical Planning	1
4.3	Heuristics for Planning	1
4.4	Partial-order Planning	2
4.5	Hierarchical Planning	1
5	**Uncertain Knowledge and Reasoning**	
5.1	Uncertainty - Inference Using Full Joint Distributions	1
5.2	Independence – Bayes Rule, and its Use - Probabilistic Reasoning	1
5.3	Representing Knowledge in an Uncertain Domain, The Semantics of Bayesian Networks	2
5.4	ExactInferences in Bayesian networks	1
5.5	Inference in Temporal models – Most likely Sequence	1
5.6	Viterbi Algorithm—Application of Viterbi Algorithm in Natural Language Processing	1

Module No.	Topic	No. of Periods
	Total	**36**

Course Designer(s):

1. Mrs.S.Saradha, APCSE ssacse@tce.edu
2. Dr.K.Sundarakantham,,PCSE kskcse@tce.edu
3. Dr.S.MercyShalinie, PCSE shalinie@tce.edu

22AM380	ARTIFICIAL INTELLIGENCE LAB	Category	L	T	P	C
		PC	0	0	2	1

Preamble

The laboratory course will facilitate the students to use appropriate search algorithms like Uniformed and Heuristic search techniques for problem-solving. Students can learn to apply the concept of artificial intelligence for different problems like N-Queens, Travelling Salesman Problem using machine learning libraries, and Python. These experiments are aimed at imparting practical exposure to the students to gain generic problem-solving skills that have applicability to a wide range of real-world problems.

Prerequisite

Fundamentals of Python programming

Course Outcomes

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

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency (%)	Expected Attainment Level (%)
CO1	Implement Uninformed search algorithms to solve given state space representable AI Problems	TPS3	A	75
CO2	Implement an efficient Informed search technique for the given problem	TPS3	A	75
CO3	Solve given 2-ply game using Min-max algorithm and optimize using alpha-beta pruning	TPS3	A	75
CO4	Represent the given Constraint Satisfaction Problem as Constraint graph and solve using Backtracking, forward checking	TPS3	A	75
CO5	Construct rule based systems for any application using logic programming language	TPS3	A	75
CO6	Examine the approximate inferencing using Bayesian belief Networks	TPS4	A	75

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L		M	L		S	S	S		M	M	L	M
CO2	S	M	L		M	L		S	S	S		M	M	L	M
CO3	S	M	L		M	L		S	S	S		M	M	L	M

CO4	S	M	L		M	L		S	S	S		M	M	L	M
CO5	S	M	L		M	L		S	S	S		M	M	L	M
CO6	S	S	M	L	M	L		S	S	S	L	M	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Model Examination	Terminal Examination
Remember		
Understand		
Apply	80	80
Analyze	20	20
Evaluate		
Create		

List of Experiments/Activities with CO Mapping

Sl.No	Experiments	CO
1.	Implement Uninformed search strategies (Breadth-first, Uniform cost, Depth-first, Depth limited)	CO1
2.	Implement Informed Search strategies (Best-first, A* algorithms)	CO2
3.	Implement 8-Puzzle, N - Queens problem, Cryptarithmetic, Travelling Salesman Problem	CO2
4.	Implement multiplayer games using Minimax, Alpha-Beta pruning	CO3
5.	Solve constraint satisfaction problems(E.g: Map coloring)	CO4
6.	Implement Nim Game You are playing the following Nim Game with your friend: • Initially, there is a heap of stones on the table. • You and your friend will alternate taking turns, and you go first. • On each turn, the person whose turn it is will remove 1 to 3 stones from the heap. • The one who removes the last stone is the winner. Given n, the number of stones in the heap, return true if you can win the game assuming both you and your friend play optimally, otherwise return false.	CO5
7.	Implement logic-based inference.	CO5
8.	Implement Sorting algorithms using forward chaining, backward chaining, and resolution strategies	CO5
9.	Build Naïve Bayes models	CO6
10.	Implement inference in Bayesian Networks	CO6
11.	Mini-Project	CO6

Learning Resources

1. Stuart Russell and Peter Norvig. Artificial Intelligence: A Modern Approach, 4th Edition, Prentice Hall, Feb 2020

2. Deepak Khemani. A First Course in Artificial Intelligence, McGraw Hill Education (India), 2013.
3. Elain Rich and Kevin Knight, “Artificial Intelligence ", Tata McGraw Hill, Third Edition, 2008

Course Designers

1. Mrs.S.Saradha, APCSE ssacse@tce.edu
2. Dr.K.Sundarakantham, PCSE kskcse@tce.edu
3. Dr.S.MercyShalinie, PCSE shalinie@tce.edu

B.E. COMPUTER SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENCE and MACHINE LEARNING) DEGREE PROGRAMME

CURRICULUM AND DETAILED SYLLABI

FOR

FOURTH SEMESTER

FOR THE STUDENTS ADMITTED IN THE

ACADEMIC YEAR 2024 - 25 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided Autonomous Institution affiliated to Anna University)
MADURAI – 625 015, TAMILNADU

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

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
BECSE(AIML) Degree Programme

COURSES OF STUDY
(For the candidates admitted from 2024-25 onwards)

FOURTH SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
24AM410	Discrete Mathematics	BSC	3	1	-	4
22CS420	Design and Analysis of Algorithms	PC	3	-	-	3
22CS430	Data Communication and Networks	PC	3	-	-	3
24AM440	Machine Learning	PC	3	-	-	3
22CS490	Project Management	HSS	3	-	-	3
THEORY CUM PRACTICAL						
24AM450	Database Systems	PC	2	-	2	3
22EG660	Professional Communication	HSS	-	1	2	2
PRACTICAL						
24AM470	Machine Learning Lab	PC	-	-	2	1
22CS480	Algorithms Lab	PC	-	-	2	1
AUDIT COURSE						
23CHAD0	Indian Constitution and Knowledge Systems	AC	-	-	-	-
Total			**17**	**2**	**8**	**23**

BSC : Basic Science Courses
ESC : Engineering Science Courses
PC : Professional Core Courses

L : Lecture
T : Tutorial
P : Practical

Note:
1 Hour Lecture is equivalent to 1 credit
1 Hour Tutorial is equivalent to 1 credit
2 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
BECSE(AIML) Degree Programme

SCHEME OF EXAMINATIONS
(For the candidates admitted from 2024-25 onwards)

FOURTH SEMESTER

S.No.	Course 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
THEORY								
1	24AM410	Discrete Mathematics	3	40	60	100	27	50
2	22CS420	Design and Analysis of Algorithms	3	40	60	100	27	50
3	22CS430	Data Communication and Networks	3	40	60	100	27	50
4	24AM440	Machine Learning	3	40	60	100	27	50
5	22CS490	Project Management	3	40	60	100	27	50
THEORY CUM PRACTICAL								
6	24AM450	Database Systems	3 (Terminal Exam Type: Theory)	50	50	100	22.5	50
7	22EG660	Professional Communication	3	50	50	100	22.5	50
PRACTICAL								
8	24AM470	Machine Learning Lab	3	60	40	100	18	50
9	22CS480	Algorithms Lab	3	60	40	100	18	50

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

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

24AM410 DISCRETE MATHEMATICS

Category	L	T	P	Credit
BSC	3	1	0	4

Preamble

Students should be able to understand Discrete Mathematical Structures (DMS) for the development of theoretical computer science, problem solving in programming language using Discrete Structure and importance of discrete structures towards simulation of a problem in computer science and engineering.

Prerequisite

Nil

Course Outcomes

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

CO	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Transform statements articulated in English into logical expressions and ascertain their equivalences	TPS3	B	80
CO2	Employ rules of logical inference to assess the validity of arguments	TPS3	B	80
CO3	Apply the concept of relations and recursion, Lattices to solve technical problems	TPS3	B	80
CO4	Apply counting techniques to solve combinatorial problems	TPS3	B	80
CO5	Apply the concept of recurrence relation to solve problems	TPS3	B	80
CO6	Apply the concept of graph theory and algebraic structures to address and solve data problems	TPS3	B	80

Mapping with Programme Outcomes

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
CO	S	M	L		L			L	L	L		L	M	L	L
CO2.	S	M	L		L			L	L	L		L	M	L	L
CO3.	S	M	L		L			L	L	L		L	M	L	L
CO4	S	M	L		L			L	L	L		L	M	L	L
CO5	S	M	L		L			L	L	L		L	M	L	L
CO6	S	M	L		L			L	L	L		L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	ASSESSMENT 1						ASSESSMENT 2						TERMINAL		
	CAT 1 (%)			ASSIGNMENT 1			CAT 2 (%)			ASSIGNMENT 2					
TPS	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
CO1	3	10	17	-	-	30	-	-	-	-	-	-	-	6	9
CO2	3	10	17	-	-	30	-	-	-	-	-	-	-	3	12
CO3	4	10	26	-	-	40	-	-	-	-	-	-	-	6	12
CO4		-		-	-		3	10	13	-	-	30	-	6	10
CO5		-	-	-	-	-		5	20	-	-	20	-	-	12
CO6		-	-	-	-	-	7	15	27	-	-	50	-	9	15
TOTAL	10	30	60	-	-	100	10	30	60	-	-	100	-	30	70

*****Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

The Foundations: Logics & Proofs: Propositional logic – Applications of propositional logic – propositional equivalences – predicates and quantifiers – Nested quantifier – rules of inferences -Introduction to proofs.

Relations: Equivalence relations - Recurrence relations, Recursion: Introduction - Recursively defined functions - Recursively defined sets and structures- Structural and general induction

Lattices: Posets -Lattices, Definitions & Examples-Properties of Lattices, Sub Lattices-Direct product and Homomorphism- Some Special Lattices

Counting: The basics of counting – The Pigeonhole principle, the generalized Pigeonhole principle – Permutations and Combinations- Binomial Coefficients and Identities- Generating functions -Introduction, solve recurrence relations using generating functions

Graphs: Graphs and Graph Models-Graph Terminology and Special Types of Graphs-Connectivity- Euler and Hamilton Paths –Planar- Graph Coloring

Algebraic Structures: The structure of Algebras- Semigroups - Monoids and Groups

Text Book

1. Kenneth H. Rosen., “Discrete Mathematics and Its Applications” 8th Edition, McGraw hill publications 2019.

Reference Books& web resources

1. T.P. Tremblay and R. Manohar, "Discrete Mathematical Structures with application to Computer Science”, Tata McGraw Hill, 2002.
2. S.B. Singh, Discrete Structures/ 3rd Edition, Khanna Book Publishing, 2019.
3. S.B. Singh, Combinatorics and Graph Theory/ 3rd Edition, Khanna Book Publishing, 2018.
4. C. L. Liu, Elements of Discrete Mathematics, 2nd Edn., Tata McGraw-Hill 2000.
5. J .L. Mott, A. Kandel, T.P .Baker, Discrete Mathematics for Computer Scientists and Mathematicians, Second edition, Prentice Hall of India 1986.

6. W. K. Grassmann and J. P. Trembnlay, Logic and Discrete Mathematics, A Computer Science Perspective, Prentice Hall Inc 1996

Course Contents and Lecture Schedule

Module No.	Topic	No. of. Lectures	COs
1	**The Foundations: Logics & Proofs**		CO1
1.1	Propositional logic	1	
1.2	Applications of propositional logic	2	
	Tutorial	1	
1.3	Propositional equivalences	2	
1.4	Predicates and quantifiers	1	
1.5	Nested quantifiers	1	CO2
	Tutorial	1	
1.6	Rules of inferences	2	
1.7	Introduction to proofs	2	
	Tutorial	1	
2	**Relations**		CO3
2.1	Equivalence relations	1	
2.2	Recurrence relations, Recursion: Introduction - Recursively defined functions	1	
2.3	Recursively defined sets and structures- Structural and general induction.	1	
	Tutorial	1	
	Lattices		
2.4	Posets	1	
2.5	Lattices, Definitions & Examples	1	
2.6	Properties of Lattices, Sub Lattices	1	
	Tutorial	1	
2.7	Direct product & Homomorphism, Some special Lattices	1	
3	**Counting**		CO4
3.1	The basics of counting	1	
3.2	The pigeonhole principle, generalized pigeonhole principle, applications of pigeonhole principle	1	
	Tutorial	1	
3.3	Permutations and combinations	2	
3.4	Binomial Coefficients and Identities	1	
	Tutorial	1	
3.5	Applications of recurrence relations	1	CO5
3.6	Solving linear recurrence relations	2	
	Tutorial	1	
3.7	Generating functions – Introduction, solve recurrence relations using generating functions	2	

4	**Graphs**		CO6
4.1	Graphs and Graph Models	1	
	Tutorial	1	
4.2	Graph Terminology and Special Types of Graphs	1	
4.5	Connectivity	2	
	Tutorial	1	
4.6	Euler and Hamilton Paths	1	
4.7	Planar	1	
4.8	Graph Coloring	1	
	Tutorial	1	
	Algebraic Structures		
4.9	The structure of algebras, Semigroups, Monoids and Groups	1	
	Tutorial	1	
	Total	**48**	

Course Designer(s):

1. Dr.C.S.Senthilkumar kumarstays@tce.edu
2. Dr. P. Krishnapriya pkamat@tce.edu
3. Dr. P. Victor pvmat@tce.edu

		Category	L	T	P	Credit
24AM440	**MACHINE LEARNING**	PC	3	0	0	3

Preamble

Machine Learning (ML) is a transformative field that enables computers to learn from and make decisions based on data. This course aims to equip students with foundational knowledge and practical skills in ML, focusing on algorithms, models, and real-world applications. It provides a structured approach to supervised, unsupervised, and semi-supervised learning, along with essential concepts in model evaluation, dimensionality reduction, and optimization. Students will develop a fundamental understanding of machine learning techniques and gain experience through case studies and hands-on implementation.

Prerequisite

- Knowledge of Linear Algebra
- Basics of Probability and Statistics

Course Outcomes

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

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Demonstrate data pre-processing techniques such as handling missing values, feature engineering, and normalization to prepare datasets for machine learning tasks.	TPS3	B	80
CO2	Analyze supervised learning methods like linear regression, logistic regression, k-NN, and SVM to solve regression and classification problems and evaluate using appropriate metrics.	TPS4	B	80
CO3	Classify ensemble methods such as Bagging, Boosting, and Random Forests based on their suitability, and break down how cross-validation affects model performance.	TPS4	B	80
CO4	Compare k-Means and Gaussian Mixture Models, and evaluate their performance using metrics like silhouette score and BIC	TPS4	B	80
CO5	Categorize dimensionality reduction techniques like PCA and Kernel PCA, and illustrate their role in simplifying datasets while preserving key patterns.	TPS3	B	80
CO6	Solve multi-class classification problems using multi-class optimization techniques and evaluate the results with accuracy and F1-score metrics.	TPS3	B	80
CO7	Construct neural networks using multi-layer perceptron (MLP) and evaluate performance metrics like cross-entropy loss to assess model effectiveness.	TPS3	B	80
CO8	Apply semi-supervised learning models by comparing their performance with fully supervised models and illustrate their practical applications.	TPS3	B	80

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L	L	L	L		L	L	L	M	M	M	L	L
CO2	S	M	M	L	L	L		L	L	L	M	M	M	L	L
CO3	S	M	M	L	L	L		L	L	L	M	M	M	L	L
CO4	S	M	M	L	L	L		L	L	L	M	M	M	L	L
CO5	S	M	L	L	L	L		L	L	L	M	M	M	L	L
CO6	S	M	L	L	L	L		L	L	L	M	M	M	L	L
CO7	S	M	L	L	L	L		L	L	L	M	M	M	L	L
CO8	S	M	L	L	L	L		L	L	L	M	M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	5	5	15												25											2	10			
CO2	5	5		15												25										2		10		
CO3	5	5		15												25										2		10		
CO4	5	5		15												25										2		10		
CO5							5	5	15												25					2	10			
CO6							5	5	15												25					4	10			
CO7							5	5	15												25					4	10			
CO8							5	5	15												25					2	10			

***** Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

INTRODUCTION TO MACHINE LEARNING AND DATA FOUNDATIONS- Introduction to Machine Learning: Motivation, applications, types (supervised, unsupervised, semi-supervised). Data and Data Curation Techniques: Data pre-processing, handling missing values, feature engineering, normalization.

SUPERVISED LEARNING - REGRESSION AND CLASSIFICATION- Regression: Linear Regression: Simple and multiple linear regression, performance evaluation with MSE (Mean Squared Error). Regularization: Ridge and Lasso Regression to prevent overfitting, evaluation using R^2 and adjusted R^2 metrics. Classification: Non-Parametric Density Estimation. Linear Classification Models: Logistic regression, linear discriminant analysis, with evaluation metrics like accuracy, precision, recall, and F1-score. Support Vector Machines (SVMs): Introduction to linear and non-linear kernel-based SVMs, performance analysis using metrics such as ROC-AUC for binary and multi-class cases. k-Nearest Neighbor (k-NN): Simple classifier with evaluation based on confusion matrix and accuracy.

ENSEMBLE LEARNING - Ensemble Techniques: Bagging, Boosting and Adaboost with cross-validation techniques for model assessment. Random Forests: Using out-of-bag (OOB) error as an internal validation measure. Model Combination Schemes: voting and stacking.

UNSUPERVISED LEARNING – Clustering: k-Means and Kernel k-Means, with evaluation metrics such as silhouette score and elbow method for optimal clusters. Gaussian Mixture Models (GMM) with Expectation-Maximization (EM) Algorithm, evaluating model fit with metrics like log-likelihood and Bayesian Information Criterion (BIC). Dimensionality Reduction: PCA, Kernel PCA evaluated by reconstruction error.

ADVANCED TOPICS - Advanced Models: Multi-Class Optimization: Techniques for extending binary classifiers to multi-class, with model assessment based on overall accuracy and F1-score. Neural Networks: Introduction to perceptron and multi-layer perceptron, with evaluation using metrics like cross-entropy loss. SVMs: Emphasis on non-linear kernel applications and dual optimization, evaluated using accuracy and F1 for non-linear separable classes. Semi-Supervised Learning: Overview of semi-supervised techniques and their real-world applications, assessing model quality by comparison with fully supervised models. Reinforcement Learning: Basic principles and performance assessment based on reward metrics.

Text Books

1. Tom M. Mitchell, Machine Learning, McGraw Hill Education, 1997.
2. Christopher Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

3. Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements of Statistical Learning, 2nd Edition, Springer, 2009.
4. Kevin Murphy, Machine Learning: A Probabilistic Perspective, The MIT Press, 2012.
5. Duda, Richard, Peter Hart, and David Stork. Pattern Classification. 2nd ed. New York, NY: Wiley-Interscience, 2000. ISBN: 9780471056690.

Reference Books & web resources

1. Peter Flach, Machine Learning: The Art and Science of Algorithms that Make Sense of Data, Cambridge University Press, 2012.
2. Shai Shalev-Shwartz and Shai Ben-David. Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press, 2014.
3. Ethem Alpaydin. Introduction to Machine Learning, PHI Learning Pvt. Ltd, 2015.
4. Richard S. Sutton and Andrew G. Barto. Reinforcement Learning – An Introduction (Adaptive Computation and Machine Learning series). MIT Press, second edition, 2018.
5. Hal Daume III, A Course in Machine Learning, 2015 (freely available online).
6. John Hopcroft, Ravindran Kannan, Foundations of Data Science, 2014 (freely available online).
7. Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. Foundations of Machine Learning, The MIT Press, 2012.
8. MacKay, David. Information Theory, Inference, and Learning Algorithms. Cambridge, UK: Cambridge University Press, 2003.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	**INTRODUCTION TO MACHINE LEARNING AND DATA FOUNDATIONS**	
1.1	Review of Linear Algebra for machine learning	1
1.2	Introduction to Machine Learning: Motivation, applications	1
1.3	Types (supervised, unsupervised, semi-supervised)	1
1.4	Data and Data Curation Techniques: Data pre-processing.	1
1.5	Handling missing values, feature engineering, normalization	1
2	**SUPERVISED LEARNING - REGRESSION AND CLASSIFICATION**	
2.1	Regression: Linear Regression: Simple and multiple linear regression, performance evaluation with MSE (Mean Squared Error)	1
2.2	Regularization: Ridge and Lasso Regression to prevent overfitting- Evaluation using R^2 and adjusted R^2 metrics	1
2.3	Classification: Non-Parametric Density Estimation.	1
2.4	Linear Classification Models: Logistic regression, linear discriminant analysis, with evaluation metrics like accuracy, precision, recall, and F1-score	2

Module No.	Topic	No. of Periods
2.5	Support Vector Machines (SVMs): Introduction to linear and non-linear kernel-based SVMs, performance analysis using metrics such as ROC-AUC for binary and multi-class cases.	2
2.6	k-Nearest Neighbor (k-NN): Simple classifier with evaluation based on confusion matrix and accuracy.	1
3	**ENSEMBLE LEARNING**	
3.1	Ensemble Techniques: Bagging, Boosting	2
3.2	Adaboost with cross-validation techniques for model assessment.	1
3.3	Random Forests: Using out-of-bag (OOB) error as an internal validation measure.	2
3.4	Model Combination Schemes: voting and stacking.	2
4	**UNSUPERVISED LEARNING**	
4.1	Clustering: k-Means and Kernel k-Means, with evaluation metrics such as silhouette score and elbow method for optimal clusters.	2
4.2	Gaussian Mixture Models (GMM) with Expectation	1
4.3	Maximization (EM) Algorithm	1
4.4	Evaluating model fit with metrics like log-likelihood and Bayesian Information Criterion (BIC).	2
4.5	Dimensionality Reduction: PCA, Kernel PCA evaluated by reconstruction error.	2
5	**ADVANCED TOPICS**	
5.1	Advanced Models: Multi-Class Optimization: Techniques for extending binary classifiers to multi-class	1
5.2	Model assessment based on overall accuracy and F1-score	1
5.3	Neural Networks: Introduction to perceptron and multi-layer perceptron, with evaluation using metrics like cross-entropy loss	1
5.4	SVMs: Emphasis on non-linear kernel applications and dual optimization,	1
5.5	Evaluated using accuracy and F1 for non-linear separable classes	1
5.6	Semi-Supervised Learning: Overview of semi-supervised techniques and their real-world applications	1

Module No.	Topic	No. of Periods
5.7	Assessing model quality by comparison with fully supervised models.	1
5.8	Reinforcement Learning: Basic principles and performance assessment based on reward metrics.	1
	Total	36

Course Designer(s):

1. Dr.M.K.Kavitha Devi,Professor,CSE- mkkdit@tce.edu.
2. Mrs.S.Jeniba,Assistant Professor- sjacse@tce.edu

24AM450	DATABASE SYSTEMS	Category	L	T	P	Credit
		PC	2	0	2	3

Preamble

The course aims at making the students to learn the concepts and functionalities of Database Systems. The course introduces different data models to store & retrieve data and make the students to execute queries using relational query languages. Further, it presents the concepts of normalization to effectively design databases. It also makes the students to know about the concepts of transaction management, concurrency control schemes and recovery mechanisms. This course also provides the concepts of big data and different forms NoSQL models to handle large data.

Prerequisite

Data Structures and Algorithms

Course Outcomes

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

	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Develop appropriate data models for the given requirements in designing the database applications	TPS3	B	80
CO2	Construct SQL queries on the database systems to implement data storage, retrieval and manipulation.	TPS3	B	80
CO3	Design database schemas using normalization and functional dependency to store data without redundancy and inconsistency	TPS3	B	80
CO4	Practice the data structures like index, hash tables to support faster retrieval of data and explain the concepts of query processing and query optimization methods	TPS3	B	80
CO5	Use the concepts of transactions, concurrency and recovery mechanisms to manage the database systems in a multi-user environment.	TPS3	B	80
CO6	Demonstrate the use of NoSQL databases for accessing and managing the data.	TPS3	B	80

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1.	S	M	L		M	L		M	M	M		M	M	L	L

CO2.	S	M	L		M	L		M	M	M		M	M	L	L
CO3.	S	M	L		L	L		M	M	M		M	M	L	L
CO4.	S	M	L						M	M		M	M		L
CO5.	S	M	L		L				M	M		M	M		L
CO6.	S	M	L		L	L			M	M		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT 1(Theory Component)						CAT 2(Theory Component)						Model (Lab Component)				OCR				Terminal (Theory Component)					
	100						100						100				100				100					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	1	2	3	4	1	2	3	4	5	6
CO1	5	10	15												20				20		2	5	10			
CO2	5	10	25												40				40		-	5	15			
CO3	5	10	15												10				10		2	5	10			
CO4							5	10	25						10				10		2	5	10			
CO5							5	10	25						10				10		2	5	10			
CO6							5	5	10						10				10		2	5	5			

*****Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to database system and Data Modelling: purpose, applications, architecture – users and administrators. Data Modelling: ER Model - Features of ER model, conceptual design using ER model, Relational Model – Structure, Reduction to relational schema

Query Languages: Relational Algebra operations, SQL: Basic operations, Integrity Constraints, set operations, Aggregate Functions, Joins, Nested and correlated queries, Views, Indexes, PL/ SQL – Procedures, Functions and Packages, Triggers, Database application development

Database Design – Normalization, Database Anomalies, Functional Dependencies, Soundness and completeness of Armstrong's axioms, Candidate and Super Key, Loss- less Decomposition, Dependency Preservation, First, Second, Third and BCNF Normal Forms.

Data Storage and Indexes - File organizations, primary, secondary index structures, hash-based indices – static and dynamic hashing techniques, B+ tree – Organization, insert and delete operations in B+ tree. **Overview of Query Processing and query optimization** – query plans and operators, cost based query optimization

Transaction Processing and Concurrency Control– Transaction concepts, states, ACID Properties, Serializability – conflict and view serializability, Concurrency control – lock based protocols, Deadlock handling. **Recovery Mechanisms:** Recovery and atomicity, Log-based recovery.

Overview of big data systems – Big data – Characteristics – Evolution – Definition - Challenges with Big Data, NoSQL, Comparison of SQL and NoSQL, NoSQL Data model – Graph databases, MongoDB Query Language, Features of Object Oriented databases, Hierarchical and Network databases

Course Project: Database application development

Text Books

1. Silberschatz, A, Henry F. Korth, and S. Sudharshan, "Database System Concepts", 7th Edition, Tata McGraw Hill, 2019.
2. R Elmasri, S Navathe, Fundamentals of Database Systems, 7th edition, Pearson, 2016

Reference Books

1. R Ramakrishnan, J Gehrke, Database Management Systems, 3rd edition, McGrawHill, 2003..
2. Andreas Meier, Michael Kaufmann, "SQL& NoSQL Databases - Models, Languages, Consistency Options and Architectures for Big Data Management" , Springer Fachmedien Wiesbaden, 2019.
3. P. Sadalage and M. Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Addison Wesley, 2012.
4. H. Garcia-Molina, J. Ullman, J. Widom, Database System: The Complete Book, 2nd Edition, Pearson, 2008.
5. https://nptel.ac.in/courses/106105175 (Database Management System, IIT Kharagpur)
6. https://nptel.ac.in/courses/106/106/106106220

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	**Introduction to Database and Data Models**	**(4)**
1.1	Purpose, applications, architecture, users and administrators	1
1.2	Data Modelling: ER Model - Features of ER model	1
1.3	Conceptual design using ER model	1
1.4	Relational Model – Structure, Reduction to relational schema	1
2	**Query Languages**	**(5)**
2.1	Relational Algebra operations	1
2.2	SQL : Basic operations, Integrity Constraints, Set operations, Aggregate Functions, Joins	1
2.3	Nested and Correlated queries, Views, Indexes	1

Module No.	Topic	No. of Periods
2.4	PL/ SQL – Procedures, Functions and Packages, Triggers	1
2.5	Database application development	1
3	**Database Design**	**(4)**
3.1	Normalization, Database Anomalies, Functional Dependencies, Soundness and completeness of Armstrong's axioms	1
3.2	Candidate and Super Key, Loss-less Decomposition, Dependency Preservation	1
3.3	First, Second Normal form, Third and BCNF Normal Forms	2
4	**Data Storage and Indexes**	**(4)**
4.1	File organizations, primary, secondary index structures	1
4.2	Hash-based indices - static and dynamic hashing techniques	1
4.3	B+ tree – Organization, insert and delete operations in B+ tree.	1
4.4	Query plans and operators, cost based query optimization	1
5	**Transaction Processing and Concurrency Control**	**(4)**
5.1	Transaction concepts, states ACID Properties	1
5.2	Serializability – conflict and view serializability	1
5.3	Concurrency control – lock based protocols, Deadlock handling	1
5.4	Recovery and atomicity, Log-based recovery	1
6	**Overview of big data systems**	**(3)**
6.1	Big data – Characteristics – Evolution – Definition, Challenges with Big Data	1
6.2	NoSQL, Comparison of SQL and NoSQL, NoSQL Data model – Graph databases, MongoDB Query Language	1
6.3	Features of Object Oriented databases, Hierarchical and Network databases	1
	Total	**24**

List of Experiments

Module No.	Topic	No. of Lectures	Course Outcome
1.	Design of database application using ER and Relational Model	2	CO1
2.	Create databases using DDL commands and integrity constraints	2	CO2
3.	Modify the relational databases using normalization principles.	2	CO3
4.	Implement SQL queries using set operations, joins and aggregate functions	2	CO2
5.	Implement SQL queries using correlated sub queries and nested subqueries	2	CO2
6.	Create different database objects such as views, indexes, sequences.	2	CO2, CO4
7.	Practice DCL and TCL commands such as grant, revoke, commit, roll back	2	CO5
8.	Develop PL/SQL Blocks, Procedures and Functions with cursors.	2	CO2
9.	Develop PL/SQL triggers, packages and exception handlers.	2	CO2
10.	Create database using MongoDB and execution of queries.	2	CO6
11.	Develop front-end application with database connectivity for storage and retrieval.	4	CO1..CO6
	Total Hours	**24**	

Course Designer(s):

1. Dr.B.Subbulakshmi bscse@tce.edu

2. Dr. N.Anita naacse@tce.edu

		Category	L	T	P	Credit
24AM470	**MACHINE LEARNING LAB**	PC	0	0	2	1

Preamble

The students are able to design and implement various machine learning algorithms in a range of real-world applications. They can interpret and analyze results with reasoning using different ML techniques. They are able to build supervised learning models and can explore the regression models. They learn to compare and evaluate the performance of different models

Prerequisite

Problem Solving and Programming

Course Outcomes

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

COs	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Apply data preprocessing steps such as handling missing data, feature scaling, encoding categorical variables, and feature selection for preparing datasets.	TPS3	A	75
CO2	Evaluate the performance of regression models such as linear regression using metrics like RMSE, MAE, and R^2, and assess their strengths and weaknesses for different datasets.	TPS3	A	75
CO3	Implement and evaluate classification algorithms, including k-NN, Decision Trees, and Naïve Bayes, and assess their effectiveness using classification metrics such as accuracy, precision, recall, and F1-score.	TPS3	A	75
CO4	Analyze the performance of Support Vector Machines (SVMs) with different kernel functions (linear, polynomial, RBF) for classification tasks.	TPS4	A	75
CO5	Apply clustering techniques such as k-Means and Gaussian Mixture Models to group data points and identify underlying patterns in unlabeled datasets.	TPS3	A	75
CO6	Implement dimensionality reduction techniques like PCA to simplify datasets while preserving critical patterns and reducing computational complexity.	TPS3	A	75

CO7	Evaluate the impact of ensemble methods such as bagging and boosting (e.g., AdaBoost) on model performance and stability, and select the best technique based on evaluation metrics.	TPS3	A	75
CO8	Apply regularization techniques to evaluate their impact on model generalization and optimization techniques to train a neural network.	TPS3	A	75

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L	L	L	L		M	M	M	L	M	M	L	L
CO2	S	M	L	M	M	L		M	M	M	L	M	M	L	L
CO3	S	M	L	M	L	L	M	M	M	M	L	M	M	L	L
CO4	S	S	M	L	L	M		M	M	M	L	M	M	L	L
CO5	S	M	L	M	L	L		M	M	M	L	M	M	L	L
CO6	S	M	L	M	M	M	M	M	M	M	L	M	M	L	L
CO7	S	M	L	M	M	L		M	M	M	L	M	M	L	L
CO8	S	M	L	M	M	L		M	M	M	L	M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Model Examination	Terminal Examination
Remember		
Understand		
Apply	80	80
Analyse	20	20
Evaluate		
Create		

List of Experiments/Activities with CO mapping

Module No	Topic	CO
1	Prepare the dataset by cleaning, transforming, and selecting relevant features to improve the model's performance.	CO1
2	Implement linear regression and assess the model's performance using evaluation metrics.	CO2
3	Implement k-Nearest Neighbors (k-NN) and Decision Tree classifiers, and evaluate their performance.	CO3
4	Implement the Naïve Bayes algorithm for classifying text data (e.g., spam detection).	CO3

5	Implement a Support Vector Machine (SVM) classifier using different kernel functions.	CO4
6	Implement unsupervised clustering algorithms to group data points and identify hidden patterns.	CO5
7	Implement Principal Component Analysis (PCA) for dimensionality reduction.	CO6
8	Implement ensemble methods like Bagging and Boosting to enhance model accuracy.	CO7
9	Build a simple Neural Network from scratch.	CO8
10	Implement cross-validation for model selection.	CO8
11	Implement regularization techniques to prevent overfitting in regression and neural networks.	CO8
12	Implement Semi-supervised learning and anomaly detection	CO8

Learning Resources

1. Andreas Muller, “Introduction to Machine Learning with Python A Guide For Data Scientists”, O'Reilly, 2016.
2. Christopher Bishop, “Pattern Recognition and Machine Learning”, Springer, 2006.
3. Ethem Alpaydin, “Introduction to Machine Learning”, MIT Press, Fourth Edition, 2020.
4. Vijayvargia, Abhishek, Machine Learning with Python: An Approach to Applied Machine Learning, BPB Publications, 1st edition, 2018.

Course Designer(s):

1. Dr.M.K.Kavitha Devi,Professor,CSE- mkkdit@tce.edu.
2. Mrs.S.Jeniba,Assistant Professor,CSE- sjacse@tce.edu

B.E. COMPUTER SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENCE and MACHINE LEARNING) DEGREE PROGRAMME

CURRICULUM AND DETAILED SYLLABI

FOR

FIFTH SEMESTER

FOR THE STUDENTS ADMITTED IN THE

ACADEMIC YEAR 2024 - 25 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided Autonomous Institution affiliated to Anna University)
MADURAI – 625 015, TAMILNADU

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

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
BECSE(AIML) Degree Programme

COURSES OF STUDY
(For the candidates admitted from 2024-25 onwards)

FIFTH SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
24AM510	Optimization Techniques	ESC	3	-	-	3
22CS520	Theory of Computation	ESC	3	-	-	3
24AM530	Deep Learning for NLP	PC	3	-	-	3
24AMPX0	Programme Elective	PE	3	-	-	3
22XXGX0	Interdisciplinary Elective	OE	3	-	-	3
24AM560	Reinforcement Learning	PC	3	-	-	3
PRACTICAL						
22CS570	Network Programming Lab	PC	-	-	2	1
22AM580	Reinforcement Learning Lab	PC	-	-	2	1
22CS590	Project – I	PW	-	-	6	3
AUDIT COURSE						
23CHAE0	Universal Human Values and Ethics	AC	-	-	-	-
Total			**18**	**-**	**10**	**23**

BSC : Basic Science Courses
ESC : Engineering Science Courses
PC : Professional Core Courses

L : Lecture
T : Tutorial
P : Practical

Note:
1 Hour Lecture is equivalent to 1 credit
1 Hour Tutorial is equivalent to 1 credit
2 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
BECSE(AIML) Degree Programme

SCHEME OF EXAMINATIONS
(For the candidates admitted from 2024-25 onwards)

FIFTH SEMESTER

S.No.	Course 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
THEORY								
1	24AM510	Optimization Techniques	3	40	60	100	27	50
2	22CS520	Theory of Computation	3	40	60	100	27	50
3	24AM530	Deep Learning for NLP	3	40	60	100	27	50
4	24AMPX0	Programme Elective	3	40	60	100	27	50
5	22XXGX0	Interdisciplinary Elective	3	40	60	100	27	50
6	24AM560	Reinforcement Learning	3	40	60	100	27	50
PRACTICAL								
7	22CS570	Network Programming Lab	3	60	40	100	18	50
8	22AM580	Reinforcement Learning Lab	3	60	40	100	18	50
9	22CS590	Project - I	3	40	60	100	27	50

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

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

		Category	L	T	P	Credit
24AM510	**OPTIMIZATION TECHNIQUES**	ESC	3	0	0	3

Preamble

An engineering UG student needs to have some basic mathematical tools and techniques to apply in diverse applications in Engineering. Optimization is a scientific approach to decision making that seeks to best design and operate a system, usually under conditions requiring the allocation of scarce resources. Various techniques of optimization have been dealt on the title "Optimization". Because of the complexity of most real-world optimization problems, it has been necessary to reduce the complexity of the problem by either simplifying the problem or constraining it by making reasonable assumptions. The course is designed to impart the knowledge and understanding the concepts on optimization techniques.

Prerequisite

- Nil

Course Outcomes

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

		TCE Proficiency Scale	Expected Proficiency	Expected Attainment Level %
CO1	Formulate mathematical models of Linear programming problems and solve it using graphical methods	TPS3	B	65
CO2	Solve linear programming problems by Simlex techniques	TPS3	B	65
CO3	Solve Integer programming problems by using cutting plane, branch and bound method.	TPS3	B	65
CO4	Apply the concepts of convex optimization to solve technical problems.	TPS3	B	65
CO5	Determine the optimum values of unconstrained non-linear programming problems using appropriate methods.	TPS3	B	65
CO6	Determine the optimum values of constrained non-linear programming problems using appropriate methods.	TPS3	B	65

Mapping with Programme Outcomes

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
CO	S	M	L		L			L	L	L	M	L	M	L	L
CO2.	S	M	L		L			L	L	L	M	L	M	L	L
CO3.	S	M	L		L			L	L	L	M	L	M	L	L
CO4	S	M	L		L			L	L	L	M	L	M	L	L
CO5	S	M	L		L			L	L	L	M	L	M	L	L
CO6	S	M	L		L			L	L	L	M	L	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	Assessment 1						Assessment 2						Terminal (%)		
	CAT 1			Assignment 1			CAT 2			Assignment 2					
TPS	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
CO1	3	10	15	-	-	30	-	-	-	-	-	-	-	6	8
CO2	4	10	30	-	-	40	-	-	-	-	-	-	-	6	15
CO3	3	10	15	-	-	30	-	-	-	-	-	-	-	3	12
CO4	-	-	-	-	-	-	4	10	14	-	-	25	-	6	8
CO5	-	-	-	-	-	-	3	10	23	-	-	40	-	6	12
CO6	-	-	-	-	-	-	3	10	23	-	-	35	-	3	15
TOTAL	10	30	60	-	-	100	10	30	60	-	-	100	-	30	70

Syllabus

Linear programming: Introduction to LPP – Formation of LPP -Geometry of Polyhedra: Feasible region of LPs and polyhedra**. -** Graphical LP solution (Extreme point solution method), The Simplex algorithm (maximization case only) – Big M- Method - Unrestricted variables, Degeneracy- Alternative Optimal Solutions, Unbounded solutions, Infeasible solution- - Primal Dual Concept- Dual Simplex Method- Introduction to Ellipsoid method.

Integer programming problem Types of integer programming problem- Gomory's cutting plane method for all and mixed integer programming problem- Branch and Bound method.

Convex Optimization: convex and Affine sets, examples and properties, convex functions, strict and strong convexity, Testing of Convexity using Hessian matrix, Examples- Introduction to Convex optimization and non-linear programming- Introduction to Quadratic Programming problems.

Unconstrained Optimization Methods: Gradient descent -Conjugate Gradient method - Newton method- Quasi newton method.

Constrained Optimization Methods: Multivariable Optimization with Equality constraints: Lagrange Multipliers Method - Multivariable optimization with inequality constraints: Kuhn-Tucker conditions – Graphical solution method.

Text Books

1. J.K.Sharma., "Operations Research Theory and Applications", 6th Edition, Trinity Press, India, 2017.
2. S.S.Rao., "Engineering Optimization Theory & Practice", Fourth edition, John Wiley & sons publications, 2009.
3. Boyd, Stephen, and Lieven Vandenberghe. "Convex optimization". Cambridge university press, 2004.
4. P.S.Gupta and D.S.Hira., "Operations Research", Seventh Edition, S.Chand & Company PVT.Ltd. 2014.

Reference Books& web resources

1. Luenberger, David G., and Yinyu Ye. Linear and nonlinear programming. 4th edition. Springer, 2015.
2. Frederick S.Hillier and Gerald J. Lieberman., " Introduction to Operations research" Tenth edition, Mc GrawHill Education, 2015.
3. Hamdy A. Taha, "Operations Research - An Introduction", Tenth Edition, Pearson, 2017.
4. J. Nocedal and S. Wright, "Numerical optimization", Second Edition, Springer, 2006.

Contents and Lecture Schedule

1	**Linear programming**		CO
1.1	Introduction to LPP – Formation of LPP	2	CO1
1.2	Geometry of Polyhedra: Feasible region of LPs and	1	CO1
	Tutorial	1	CO1
1.4	Graphical LP solution (Extreme point solution method)	1	CO1

1.5	The Simplex algorithm (maximization case only)	2	CO2
	Tutorial	1	CO2
1.6	Big M- Method	1	CO2
1.7	Unrestricted variables, Degeneracy- Alternative Optimal	1	CO2
	Tutorial	1	CO2
1.8	Primal Dual Concept- Dual Simplex Method-	1	CO2
1.9	Introduction to Ellipsoid method	1	CO2
	Program Using Python / Matlab		
2	**Integer programming problem**		
2.1	Introduction and Types of Integer programming	1	CO3
2.2	Gomory's cutting plane method all Integer Programming	1	CO3
2.3	Gomory's cutting plane method mixed Integer	1	CO3
	Tutorial	1	CO3
2.3	Branch and Bound method.	1	CO3
	Program Using Python / Matlab		
3.	**Convex Optimization**		
3.1	Convex and Affine sets, examples and properties, convex	2	CO4
3.2	Introduction to Convex optimization and non-linear	1	CO4
	Tutorial	1	CO4
3.3	Introduction to Quadratic Programming problems	1	CO4
	Program Using Python / Matlab		
4	**Unconstrained Optimization Methods**		
4.1	Gradient descent Method	2	CO5
4.2	Conjugate Gradient method	2	CO5
	Tutorial	1	CO5
4.3	Newton method	1	CO5
4.4	Quasi newton method.	1	CO5
	Program Using Python / Matlab		
5	**Constrained Optimization Methods**:		
5.1	Multivariable Optimization with Equality constrains:	2	CO6
5.2	Multivariable optimization with inequality constrains:	2	CO6
	Tutorial	1	CO6
5.3	Graphical solution method for NLPP	1	CO6
	Program Using Python / Matlab		
	Total	**36**	

Course Designer(s):

1. Dr.C.S.Senthilkumar kumarstays@tce.edu
2. Dr. P. Victor pvmat@tce.edu
3. Dr. P. Krishnapriya pkamat@tce.edu

24AM530 DEEP LEARNING FOR NLP

Category	L	T	P	Credit
PC	3	0	0	3

Preamble

To introduce students to deep learning techniques in Natural Language Processing (NLP), covering methods from word embeddings to advanced neural architectures. The course emphasizes practical applications using models like RNNs, LSTMs, and Transformers. Students will apply these models to real-world tasks such as translation, QA, and chatbots.

Prerequisite

- Foundations of Machine Learning
- Basics of Natural Language Processing

Course Outcomes

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

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Articulate pattern matching with regular expressions, text normalization, and edit distance for pre-processing and cleaning textual data in NLP.	TPS 3	70	80
CO2	Explore language models, N-gram approaches and machine learning techniques .	TPS 3	70	80
CO3	Use vector representations and neural networks to capture semantics and context in textual data	TPS 3	70	80
CO4	Apply sequential models like RNNs, LSTMs, Transformers, and pre-trained models for tasks such as sequence labelling	TPS 3	70	80
CO5	Simulate language models, use masked language models, and apply prompting techniques for specific NLP tasks	TPS 3	70	80
CO6	Correlate NLP algorithms in real-world applications like translation, question answering.	TPS 4	70	80

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
C	S	M	L			L	L	M	M	L		M	M	L	M
CO2.	S	M	L			L	L	M	M	L		M	M	L	M
CO3.	S	M	L			L	L	M	M	L		M	M	L	M
CO4.	S	M	L			L	L	M	M	L		M	M	L	M
CO5.	S	M	L			L	L	M	M	L		M	M	L	M
CO6.	S	S	M	L	L	L	L	M	M	L	L	M	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	10	10											30											2	4	5			
CO2	10	10	10											30											2	4	5			
CO3	10	10	20												40										2	4	10			
CO4							5	10	15												30				2	4	10			
CO5							5	15	15												30				2	4	10			
CO6							5	10	10	10											40				2	4	10	14		

*****Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Working with Text data –

Modern LLM Task- Regular Expressions – Words – Corpora – Text Normalization – Minimum Edit Distance- Computational Semantics and Semantic Parsing - Relation and Event Extraction - Time and Temporal Reasoning.

N-gram Language Models - N-Grams - Evaluating Language Models - Sampling sentences from a language model - Generalization and Zeros – Smoothing - Huge Language Models and Stupid Backoff -Lexical Semantics.

Vector Semantics and Embeddings - Vector Semantics - Words and Vectors - Cosine for measuring Similarity-Weighing terms in the vector - Pointwise Mutual Information (PMI) - Word2vec - Visualizing Embeddings - Semantic properties of embeddings - Bias and Embeddings.

Sequence Labelling for Parts of Speech and Named Entities - Part-of-Speech Tagging - Named Entities and Named Entity Tagging - Recurrent Neural Networks - RNNs as Language Models - The LSTM - The Encoder-Decoder Model with RNNs: Transformers - Transformers as Language Models – Sampling - Beam Search - Pretraining Large Language Models**.**

Fine-Tuning and Masked Language Models - Bidirectional Transformer Encoders - Training Bidirectional Encoders - Transfer Learning through Fine-Tuning - Training Corpora -Prompting.

Applications – Question Answering–Knowledge graph- Graph models, Graph Network, Graph analytics and Indic models- Machine Translation - Information Retrieval - IR-based Factoid Question Answering - Knowledge-based Question Answering.

Text Book

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, “Deep Learning”, MIT Press, 2016
2. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition Third Edition draft Daniel Jurafsky Stanford University James H. Martin University of Colorado at Boulder, 2023.
3. James Allen “Natural Language Understanding”, Pearson Education, 2003
4. Christopher D.Manning and Hinrich Schutze, “ Foundations of Statistical Natural Language Processing “, MIT Press, 1999.
5. Daniel Jurafsky and James H. Martin, “ Speech and Language Processing” , Pearson, 2008.

Reference Books& web resources

1. Ron Cole, J.Mariani, et.al “Survey of the State of the Art in Human Language Technology”, Cambridge University press, 1997.
2. Michael W. Berry, “ Survey of Text Mining: Clustering, Classification and Retrieval”, Springer Verlag, 2003.
3. Steven Bird, Ewan Klein, Edward Loper, “Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit”. First Edition, O’REILLY, 2009.
4. Christopher D. Manning. 2015. Computational Linguistics and Deep Learning. Computational Linguistics 41(4): 701-707. http://nlp.stanford.edu/~manning/papers/Manning-Last-Words-COLI_a_00239.pdf

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	**Working with Text data**	
1.1	Modern LLM Task -Regular Expressions – Words-Corpora	1
1.2	Text Normalization- Minimum Edit Distance	1
1.3	Computational Semantics and Semantic Parsing	1
1.4	Relation and Event Extraction - Time and Temporal Reasoning	2
2	**N-gram Language Models**	
2.1	N-Grams - Evaluating Language Models	1
2.2	Sampling sentences from a language model-Generalization and Zeros	2
2.3	Smoothing- Huge Language Models and Stupid Backoff	2
2.4	Lexical Semantics	1
3	**Vector Semantics and Embeddings**	
3.1	Vector Semantics	1
3.2	Words and Vectors -Cosine for measuring similarity	2
3.3	Weighing terms in the vector - Pointwise Mutual Information (PMI)	2
3.4	Word2vec - Visualizing Embeddings	1

Module No.	Topic	No. of Periods
3.5	Semantic properties of embeddings - Bias and Embeddings.	2
4	**Sequence Labelling for Parts of Speech and Named Entities**	
4.1	Part-of-Speech Tagging - Named Entities and Named Entity Tagging	1
4.2	Recurrent Neural Networks	1
4.3	RNNs as Language Models	1
4.4	The LSTM - The Encoder-Decoder Model with RNNs	2
4.5	Transformers - Transformers as Language Models	1
4.6	Sampling - Beam Search	1
4.7	Pre-training Large Language Models**.**	1
5	**Fine-Tuning and Masked Language Models**	
5.1	Bidirectional Transformer Encoders - Training Bidirectional Encoders	1
5.2	Transfer Learning through Fine-Tuning	1
5.3	Training Corpora-Prompting	1
6	**Applications**.	
6.1	Question Answering- Knowledge graph-Graph models, Graph Network	2
6.2	Graph analytics and Indic models -Machine Translation	1
6.3	Information Retrieval - IR-based Factoid Question Answering	2
6.4	Knowledge-based Question Answering	1
	Total	**36**

Course Designer(s):

1. Mrs.S.Saradha,AssistantProfessor,ssacse@tce.edu

24AM560	REINFORCEMENT LEARNING	Category	L	T	P	Credit
		PC	3	0	0	3

Preamble

This course introduces the fundamental concepts of Reinforcement Learning, where agents learn to make decisions through trial and error in dynamic environments. Students will explore core topics such as Markov Decision Processes, Q-learning, and the exploration-exploitation trade-off. Advanced techniques including Deep Q-Networks, PPO, and policy optimization methods will be covered, with practical applications in robotics, gaming, finance, and healthcare.

Prerequisite

Knowledge of python programming, linear algebra, probability, statistics, and fundamental machine learning concepts.

Course Outcomes

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

	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Explain the foundational principles of Reinforcement Learning and formulate the Learning tasks	TPS3	70	80
CO2	Apply the concepts of Markov Decision Processes to model decision-making problems	TPS3	70	80
CO3	Implement the dynamic programming methods to solve MDP and find optimal policies.	TPS3	70	80
CO4	Apply Monte Carlo and Temporal-Difference methods to estimate value functions and improve policies.	TPS3	70	80
CO5	Demonstrate the function approximation methods to improve learning in complex environments.	TPS3	70	80
CO6	Apply RL in robots, games, healthcare, finance, impact on recommendations/plans/management.	TPS3	70	80

Mapping with Programme Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO7.	S	M	L		L			M	M	M		M	M	L	M
CO8.	S	M	L		L	L	L	M	M	M		M	M	L	M
CO9.	S	M	L		L	L	L	M	M	M		M	M	L	M
CO10	S	M	L		L	L	L	M	M	M		M	M	L	M

CO11	S	M	L		L	L	L	M	M	M		M	M	L	M
CO12	S	M	L	L	L	L	L	M	M	M	L	M	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass 1	CAT2						Ass2	Terminal					
TPS Scale	**1**	**2**	**3**	**4**	**5**	**6**		**1**	**2**	**3**	**4**	**5**	**6**		**1**	**2**	**3**	**4**	**5**	**6**
CO1	10	10	15	-	-	-	30	-	-	-	-	-	-	-	2	5	-	-	-	-
CO2	10	10	15	-	-	-	35	-	-	-	-	-	-	-	2	5	12	-	-	-
CO3	-	10	20	-	-	-	35	-	-	-	-	-	-	-	2	5	12	-	-	-
CO4	-	-	-	-	-	-	-		10	15	-	-	-	20	2	5	12	-	-	-
CO5	-	-	-	-	-	-	-	10	10	15	-	-	-	20	2	5	12	-	-	-
CO6	-	-	-	-	-	-	-	10	10	20	-	-	-	60		5	12	-	-	-

*****Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction: Introduction – Early History of Reinforcement Learning (RL), Elements of RL, scope and limitations, Bellman Equations, Policy and Value functions

Markov Decision Process: Introduction to Markov decision process (MDP), The Agent-Environment Interface, Goals and Rewards, Returns and Episodes-state and action value functions, Bellman expectation equations, optimality of value functions and policies, Bellman optimality equations, Partial Observable Markov Decision Process (POMDP)

Dynamic Programming: Overview of dynamic programming for MDP, definition and formulation of planning in MDPs, principle of optimality, iterative policy evaluation, policy iteration, value iteration.

Monte Carlo and Temporal - Difference: Prediction, Estimation of Action Values- Monte Carlo Control, Off-policy Prediction, Incremental Implementation, Temporal-Difference-TD(0), TD(1),SARSA, Q-Learning, Epsilon-greedy, Boltzmann, Upper Confidence Bound (UCB),Exploitation Strategies and Trade-offs.

Function approximation: Eligibility traces, Linear Squares TD (LSTD), Monte Carlo Policy gradient method-Semi-gradient TD(0) algorithms, Control with function approximation, Least squares, Incremental Methods and Batch Methods, Deep Q-Learning, Deep Q-Networks and Experience Replay Semi-gradient methods, SGD, Replay Buffer.

Applications and Case Studies: Deep RL, Hierarchical RL, Partially observable MDP, TD Gammon, Checker's player, Alpha Go, Robotics and Control Systems, Autonomous robotic manipulation, Path planning and adaptive control, Autonomous Vehicles-Traffic Control Systems-Ship Route Planning

Text Books

1. Richard S. Sutton and Andrew G. Barto, "Reinforcement learning: An introduction", Second Edition, MIT Press, 2019.

2. Russell, Stuart J., and Peter Norvig. "Artificial intelligence: a modern approach."Pearson Education Limited, 4th Edition, 2020.

Reference Books

1. Marco Wiering, Martijn van Otterlo(Ed), Reinforcement Learning, State-of-the-Art, Adaptation, Learning, and Optimization book series, ALO, volume 12, Springer, 2012.
2. Deep Learning, Ian Goodfellow, YoshuaBengio, and Aaron Courville, The MIT Press, 2016.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	**Introduction**	
1.1	Introduction – Early History of Reinforcement Learning (RL)	1
1.2	Elements of RL, scope and limitations	1
1.3	Bellman Equations	1
1.4	Policy and Value functions	1
2	**Markov Decision Process**	
2.1	Introduction to Markov decision process (MDP), The Agent-Environment Interface	2
2.2	Goals and Rewards, Returns and Episodes-state and action value functions	2
2.3	Bellman expectation equations, optimality of value functions and policies	2
2.4	Bellman optimality equations, Partial Observable Markov Decision Process (POMDP)	1
3	**Dynamic Programming**:	
3.1	Overview of dynamic programming for MDP	1
3.2	Definition and formulation of planning in MDPs	1
3.3	principle of optimality, iterative policy evaluation	2
3.4	Policy iteration, value iteration.	1
4	**Monte Carlo and Temporal – Difference**	

Module No.	Topic	No. of Periods
4.1	Prediction, Estimation of Action Values- Monte Carlo Control, Off-policy Prediction	2
4.2	Incremental Implementation	1
4.3	Temporal-Difference-TD(0)	1
4.4	TD(1),SARSA, Q-Learning,Epsilon-greedy, Boltzmann	2
4.5	Upper Confidence Bound (UCB),Exploitation Strategies and Trade-offs.	2
5	**Function approximation**	
5.1	Eligibility traces, Linear Squares TD (LSTD), Monte Carlo Policy gradient method	1
5.2	Semi-gradient TD(0) algorithms, Control with function approximation,	2
5.3	Least squares, Incremental Methods and Batch Methods	1
5.4	Deep Q-Learning, Deep Q-Networks and Experience Replay Semi-gradient methods,	2
5.5	SGD, Replay Buffer	1
6	**Applications and Case Studies**	
6.1	Deep RL, Hierarchical RL, Partially observable MDP	1
6.2	TD Gammon, Checker's player, Alpha Go-	1
6.3	Robotics and Control Systems- Autonomous robotic manipulation	1
6.4	Path planning and adaptive control, Autonomous Vehicles, Traffic Control Systems ,Ship Route Planning	2
	Total	**36**

Course Designer(s):

1. Mrs.S.Jeniba sjacse@tce.edu

24AM580	REINFORCEMENT LEARNING LAB	Category	L	T	P	C
		PC	0	0	2	1

Preamble

The Laboratory introduces students to the principles of learning through interaction with an environment using trial and error. Students will implement foundational RL algorithms such as Q-learning, Monte Carlo methods, and Deep Q-Networks using Python and machine learning libraries. Through hands-on experiments, they will train agents to solve problems like grid navigation, game playing, and decision-making in dynamic environments.

Prerequisite

Knowledge on Fundamentals of Python programming.

Course Outcomes

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

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency (%)	Expected Attainment Level (%)
CO1	Apply Markov Decision Processes (MDPs) and Bellman Equations to solve real-world decision-making problems.	TPS3	80	75
CO2	Implement Dynamic Programming techniques such as Value Iteration and Policy Iteration to compute and optimize policies for various MDPs.	TPS3	80	75
CO3	Apply Monte Carlo methods and Temporal Difference (TD) learning algorithms to estimate value functions.	TPS3	80	75
CO4	Implement Q-learning algorithms to enable agents to autonomously learn optimal policies through exploration-exploitation in complex environments.	TPS3	80	75
CO5	Use function approximation techniques like Linear Squares TD (LSTD) and Semi-Gradient TD(0) to scale reinforcement learning algorithms to large state and action spaces, improving performance in real-world applications.	TPS3	80	75
CO6	Design and apply Deep Q-Networks (DQN) using neural networks and Experience Replay for reinforcement learning in high-dimensional environments, and implement reinforcement learning for robotic path planning and adaptive control.	TPS3	80	75

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	S	M	L		M	M		M	M	M	M	M	M	M	M
CO2	S	M	L		M	M		M	M	M	M	M	M	M	M
CO3	S	M	L		M	M		M	M	M	M	M	M	M	M
CO4	S	M	L		M	M		M	M	M	M	M	M	M	M
CO5	S	M	L		M	M		M	M	M	M	M	M	M	M
CO6	S	M	L	M	M	M	M	M	M	M	M	M	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Model Examination	Terminal Examination
Remember		
Understand		
Apply	80	80
Analyze	20	20
Evaluate		
Create		

List of Experiments/Activities with CO Mapping

Sl.No	Experiments	CO
1.	Develop a simple grid-world environment where the agent learns to navigate using predefined states, actions, and rewards.	CO1
2.	Implement Bellman Expectation Equations and solve the value function for a given MDP using Policy Iteration.	CO1
3.	Implement Value Iteration algorithm to find the optimal policy for an MDP.	CO2
4.	Use Monte Carlo methods to estimate the value function and develop a policy for episodic tasks.	CO3
5.	Implement TD learning methods (TD (0) and TD (1)) to estimate value functions.	CO3
6.	Implement the Q-learning algorithm to optimize dynamic CPU allocation under varying loads.	CO4
7.	Implement Linear Squares TD (LSTD) for function approximation in reinforcement learning tasks.	CO5
8.	Implement a Deep Q-Network (DQN) using neural networks to approximate Q-values.	CO6
9.	Implement Experience Replay and Semi-gradient methods for improving Q-learning stability and convergence.	CO6
10.	Apply reinforcement learning to robotic path planning and adaptive control systems.	CO6

Learning Resources

1. Richard S. Sutton and Andrew G. Barto, "Reinforcement learning: An introduction", Second Edition, MIT Press, 2019.

2. Marco Wiering, Martijn van Otterlo(Ed), ”Reinforcement Learning, State-of-the-Art, Adaptation, Learning, and Optimization book series”, ALO, volume 12, Springer, 2012.

Course Designers

1. Mrs.S.Jeniba, APCSE sjacse@tce.edu

B.E. COMPUTER SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENCE and MACHINE LEARNING) DEGREE PROGRAMME

CURRICULUM AND DETAILED SYLLABI

FOR

SIXTH SEMESTER

FOR THE STUDENTS ADMITTED IN THE

ACADEMIC YEAR 2024 - 25 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided Autonomous Institution affiliated to Anna University)
MADURAI – 625 015, TAMILNADU

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

Passed in Board of Studies meeting on 03.05.2025 Approved in 69th Academic Council meeting on 28.06.2025

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
BECSE(AIML) Degree Programme

COURSES OF STUDY
(For the candidates admitted from 2024-25 onwards)

SIXTH SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
24AM610	Social and Ethical aspects of AI	HSS	3	-	-	3
24AM620	Deep Learning for Computer Vision	PC	3	-	-	3
24AM630	Human-AI Interaction	PC	3	-	-	3
24AMPX0	Programme Elective	PE	3	-	-	3
22XXFX0	Basic Science Elective	OE	3	-	-	3
24AMPX0	Programme Elective	PE	3	-	-	3
PRACTICAL						
24AM670	Deep Learning Lab	PC	-	-	2	1
22CS690	Project-II	PW	-	-	6	3
Total			**18**	**-**	**8**	**22**

BSC : Basic Science Courses
ESC : Engineering Science Courses
PC : Professional Core Courses

L : Lecture
T : Tutorial
P : Practical

Note:
1 Hour Lecture is equivalent to 1 credit
1 Hour Tutorial is equivalent to 1 credit
2 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
BECSE(AIML) Degree Programme

SCHEME OF EXAMINATIONS
(For the candidates admitted from 2024-25 onwards)

SIXTH SEMESTER

S.No.	Course 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
THEORY								
1	24AM610	Social and Ethical aspects of AI	3	40	60	100	27	50
2	24AM620	Deep Learning for Computer Vision	3	40	60	100	27	50
3	24AM630	Human-AI Interaction	3	40	60	100	27	50
4	24AMPX0	Program Elective	3	40	60	100	27	50
5	22XXFX0	Basic Science Elective	3	40	60	100	27	50
6	24AMPX0	Program Elective	3	40	60	100	27	50
PRACTICAL								
7	24AM670	Deep Learning Lab	3	60	40	100	18	50
8	22CS690	Project-II	3	40	60	100	27	50

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

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

24AM610	SOCIAL AND ETHICAL ASPECTS OF AI	Category	L	T	P	Credit
		HSS	3	0	0	3

Preamble

The aim of this course is to make learners understand the importance of ethics in AI development, how data collection and training can introduce bias, fairness frameworks and techniques to mitigate bias, to assess fairness in AI systems. The learners may also know the importance of accountability in AI, how transparency helps build trust in AI systems, data protection regulations and their implications for AI. The course may enforce the ethical issues related to autonomous AI systems, the potential benefits and harms of AI in medical decision-making, the ethical concerns around job displacement and automation, the role of governments and regulatory bodies in ensuring ethical AI practices. At the end of this course learners will be able to apply ethical principles and Tools learned throughout the course.

Prerequisite

- Basic Understanding of AI Concepts
- Basic Understanding of Data Science concepts
- Basic Understanding of Ethics
- Critical Thinking and Analytical Skills.

Course Outcomes

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

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Describe about historical context and evolution of AI with the Ethical Concerns.	TPS2	70	80
CO2	Develop strategies to reduce bias and understand the types & impact of AI Bias on Society.	TPS3	70	75
CO3	Use the concepts of fairness, accountability and transparency with machine learning to promote AI development.	TPS3	70	75
CO4	Protect user data by adapting global privacy laws and regulation with respect to General Data Protection Regulation (GDPR).	TPS3	70	75
CO5	Explore methods, frameworks and tools for auditing and monitoring ethical AI Systems.	TPS3	70	75
CO6	Analyse the Real-world AI System for Ethical Issues and build ethical AI System.	TPS4	70	75

Mapping with Programme Outcomes

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1	M	L			L	L	L	L	L	L		L	L	L	L
CO2	S	M	L		M	M	L	M	M	M		M	M	L	M
CO3	S	M	L		M	M	L	M	M	M		M	M	L	M
CO4	S	M	L		M	M	L	M	M	M		M	M	L	M
CO5	S	M	L		M	M	L	M	M	M		M	M	L	M
CO6	S	M	L	L	M	M	L	M	M	M	L	S	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass1	CAT2						Ass2	Terminal					
TPS Scale	1	2	3	4	5	6		1	2	3	4	5	6		1	2	3	4	5	6
CO1	10	20	-	-	-	-	30	-	-	-	-	-	-	-	2	5	-	-	-	-
CO2	5	10	20	-	-	-	35	-	-	-	-	-	-	-	2	5	12	-	-	-
CO3	5	10	20	-	-	-	35	-	-	-	-	-	-	-	2	5	12	-	-	-
CO4	-	-	-	-	-	-	-	5	10	20	-	-	-	25	2	5	12	-	-	-
CO5	-	-	-	-	-	-	-	5	10	20	-	-	-	25	2	5	12	-	-	-
CO6	-	-	-	-	-	-	-	5	10	-	15	-	-	50	-	5	-	12	-	-

*****Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

CO6 can be evaluated through Mini project

Syllabus

Introduction to AI Ethics: AI Ethics -The Need for Ethical Considerations in AI Development-Definition of morality and ethics-Moral Theories: Utilitarianism, Deontology, and Virtue Ethics- AI-Impact on Society-Impact on human Psychology-Impact on the legal System-Impact on the environment and the Planet-Impact on trust.

Bias in AI Systems: Types of Bias: Algorithmic, Data, Model, and User Bias-Sources of Bias in Data -Sampling Bias, Measurement Bias-Bias qualifying metrics - Equal Opportunity Difference, and Statistical Parity measure unfair treatment or outcomes across different demographic groups- **The Impact of AI Bias on Society: Case Study** :**Employment:** Ethical Challenges in Mass Job Displacement-Economic Inequality and AI-driven Unemployment-Policies for a Fair Transition to an AI-enabled Workforce, **Healthcare**: AI-based Diagnostics and Ethical Challenges in Accuracy-**Criminal Justice** and AI Bias-Addressing discrimination in AI systems-Discrimination in AI hiring Algorithms.

Fairness in AI: Frameworks for Fairness: Equalized Odds, Demographic Parity, and Calibration-Techniques for Ensuring Fairness in AI Models-The Trade-offs between Fairness, Accuracy, and Utility- Fairness Key metrics -Disparate Impact, Statistical Parity, Equal Opportunity, Equalized Odds, and Calibration Error.

Accountability and Transparency in AI: Defining the Role of Stakeholders: Developers, Companies, Governments-Algorithmic Transparency and Explainability-Black-box AI Models vs. Explainable AI (XAI)-XAI Techniques: SHAP and LIME -Saliency maps- Rule-based systems and counterfactual explanations- Regulatory Approaches: The European Union's AI Act and General Data Protection Regulation (GDPR).

Privacy and Data Protection in AI: Privacy Implications of AI: Surveillance, Profiling, and Data Collection-Privacy-Preserving Techniques in AI: Differential Privacy, Federated Learning-Data Anonymization and Encryption-Data Protection Regulations: Indian Government Regulations and their Impact on AI-**Case Study:** Data Breaches and AI's Role in Privacy Violations-National Strategy for Artificial Intelligence (NSAI) -Responsible AI Guidelines (2021) -IndiaAI Mission (2023) -Personal Data Protection Act (2023) - India's NITI Aayog-EU's AI Act-US and UK -AI Safety Institutes

Building Ethical AI Systems: Best Practices and Tools-Best Practices for Ethical AI Development-Ethics in AI Research and Model Deployment-AI Ethics Frameworks: AI4People, Google's AI Principles-Tools for Auditing and Monitoring AI Systems for Bias and Fairness-Future Trends in AI Ethics

Text Books:

1. Paula Boddington, "AI Ethics: Artificial Intelligence: Foundations, Theory, and Algorithms", Springer Singapore ,1st Edition,2023.
2. Mishra, S., & Agarwal, S. "Responsible AI: Implementing ethical and unbiased algorithms, Springer Nature Switzerland AG,1st Edition,2021.

Reference Books:

1. Christoph Bartneck, Christoph Lutge, Alan Wagner, Sean Welsh, "An Introduction to Ethics in Robotics and AI", Springer Nature Switzerland,1st Edition, 2021.
2. Bernd Carsten Stahl, Doris Schroeder, Rowena Rodrigues, "Ethics of Artificial Intelligence", Springer Cham, 1st Edition, 2022.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	**Introduction to AI and Ethics**	
1.1	**Introduction to AI Ethics:** AI Ethics -The Need for Ethical Considerations in AI Development	2
1.2	Definition of morality and ethics-Moral Theories: Utilitarianism, Deontology, and Virtue Ethics	2
1.3	AI-Impact on Society-Impact on human Psychology-Impact on the legal System-Impact on the environment and the Planet-Impact on trust	2
2	**Bias in AI Systems**	
2.1	Types of Bias: Algorithmic, Data, Model, and User Bias-Sources of Bias in Data -Sampling Bias, Measurement Bias	2
2.2	**The Impact of AI Bias on Society: Employment:** Ethical Challenges in Mass Job Displacement-Economic Inequality and AI-driven Unemployment-Policies for a Fair Transition to an AI-enabled Workforce	2

Module No.	Topic	No. of Periods
2.3	**Healthcare:** AI-based Diagnostics and Ethical Challenges in Accuracy-**Criminal Justice**- Facial Recognition and Bias	2
3	**Fairness in AI**	
3.1	Frameworks for Fairness: Equalized Odds, Demographic Parity, and Calibration	2
3.2	Techniques for Ensuring Fairness in AI Models	2
3.3	The Trade-offs between Fairness, Accuracy, and Utility	2
4	**Accountability and Transparency in AI**	
4.1	Defining the Role of Stakeholders: Developers, Companies, Governments-Algorithmic Transparency and Explainability	2
4.2	Black-box AI Models vs. Explainable AI (XAI)	2
4.3	Regulatory Approaches: The European Union's AI Act and General Data Protection Regulation (GDPR)	2
5	**Privacy and Data Protection in AI**	
5.1	Privacy Implications of AI: Surveillance, Profiling, and Data Collection	2
5.2	Privacy-Preserving Techniques in AI: Differential Privacy, Federated Learning-Data Anonymization and Encryption	2
5.3	Data Protection Regulations: Indian Government Regulations and Their Impact on AI- Case Study: Data Breaches and AI's Role in Privacy Violations	2
6	**Building Ethical AI Systems**	
6.1	Best Practices and Tools-Best Practices for Ethical AI Development-Ethics in AI Research and Model Deployment	2
6.2	AI Ethics Frameworks: AI4People, Google's AI Principles-Tools for Auditing	2
6.3	Monitoring AI Systems for Bias and Fairness-Future Trends in AI Ethics	2
	Mini Project: Group Projects: Analysing a Real-world AI System for Ethical Issues-Presentations and Peer Review of Ethical Case Studies-Summary and Key Takeaways from the Course-Future Directions in AI Ethics	
	Total	**36**

Course Designer(s):

1. Dr.M.NirmalaDevi,Assistant Professor(Sel.Gr),CSE mnit@tce.edu
2. Dr.K.Indira,Assistant Professor,CSE kiit@tce.edu

24AM620	DEEP LEARNING FOR COMPUTER VISION	Category	L	T	P	Credits
		PC	3	0	0	3

Preamble

This course offers a comprehensive introduction to deep learning and computer vision, equipping learners with foundational knowledge and practical skills to tackle modern visual recognition tasks. From core neural network principles to advanced applications like object detection, image synthesis, and video analysis, students will explore both theoretical concepts and hands-on tools. Ethical considerations and explainability techniques are also emphasized to foster responsible AI development

Prerequisite

Machine Learning

Course Outcomes

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

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Discover the fundamentals of machine learning, deep learning, and computer vision.	TPS 2	70	75
CO2	Describe CNN architectures and its operations to solve image classification and related tasks	TPS 2	70	75
CO3	Illustrate advanced vision models including transformers and 3D CNNs.	TPS 3	70	70
CO4	Develop generative models for image synthesis and translation using GANs and VAEs.	TPS 3	70	70
CO5	Examine model decisions using explainability techniques like Grad-CAM, LRP.	TPS 3	70	70
CO6	Relate real-world Computer Vision applications with industrial tools and libraries	TPS 3	70	70

Mapping with Programme Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	M	L	-	-	L	L	-	M	M	-	-	M	L	L	M
CO2	M	L	-	-	L	L	-	M	M	M	-	M	L	L	M
CO3	S	M	L	-	L	L	-	M	M	M	-	M	M	L	M
CO4	S	M	L	-	L	L	-	M	M	M	-	M	M	L	M
CO5	S	M	L	-	M	L	-	M	M	M	-	M	M	M	M
CO6	S	M	L	L	M	L	L	M	M	M	-	M	M	M	M

S- Strong; M- Medium; L- Low

Assessment Pattern

CO	CAT1						CAT2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	20	-	-	-	-	-	-	-	-	-	-	2	5	-	-	-	-
CO2	10	20	-	-	-	-	-	-	-	-	-	-	2	5	-	-	-	-
CO3	-	20	20	-	-	-	-	-	-	-	-	-	2	5	15	-	-	-
CO4	-	-	-	-	-	-	10	20	10	-	-	-	2	5	15	-	-	-
CO5	-	-	-	-	-	-	10	20	10	-	-	-	2	5	15	-	-	-
CO6	-	-	-	-	-	-	-	10	10	-	-	-	-	5	15	-	-	-

CO	Assignment 1			Assignment 2		
TPS Scale	1	2	3	1	2	3
CO1	-	-	-	-	-	-
CO2	-	-	-	-	-	-
CO3	-	-	100	-	-	-
CO4	-	-	-	-	-	-
CO5	-	-	-	-	-	-
CO6	-	-	-	-	-	100

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Fundamentals of Deep Learning and Computer Vision: Computer Vision: Tasks (Classification, Detection, Segmentation), Applications, Challenges, Similarity indices (Superpixel between images), Tools and libraries (TensorFlow basics, OpenCV, Scikit-image, MediaPipe, Matplotlib, Blaze) - Deep learning: neurons, Activation functions, Loss functions - Training neural networks: Backpropagation, Optimization (SGD, Adam, RMSProp), Regularization

Convolutional Neural Networks (CNNs): CNN architecture and operations: Convolution, Pooling, Stride, Padding - Classical CNN models: LeNet, AlexNet, VGG, ResNet - Transfer learning and fine-tuning - Data augmentation & model evaluation – Object detection models: R-CNN, Fast R-CNN, Faster R-CNN, YOLO, SSD – Object segmentation: Semantic Vs Instance segmentation, FCNs, U-Net, Mask R-CNN – Pre-GAN Applications: Posture recognition, Gesture recognition, Anomaly detection

Advanced Vision Models, Transformers and Multimodal Systems: Advanced vision architecture - Vision Transformer (ViT), Inverse Convolution/Deconvolution, CUCN, Sparse optimization Techniques – Sequential and video Models – 3D CNN, ConvLSTM, Optimal flow, Action recognition, Video classification, Video streaming – Multimodal vision and language integration – Image captioning, BRAD model, Visual Question Answering (VQA), CLIP, BLIP

Generative Models and Image Synthesis: Autoencoders and Variational Autoencoders (VAEs) - Generative Adversarial Networks (GANs): DCGAN, CycleGAN, StyleGAN - Applications: image enhancement, super-resolution, image-to-image translation – Diffusion models: Concept, Comparison with GANs, Applications in generation and restoration

Model Explainability: Interpretability techniques: Saliency maps, Grad-CAM, Layer-wise Relevance Propagation (LRP) – Use cases of Explainability: Medical diagnosis, Autonomous vehicles comparison and Limitations of Explainability methods

Applications and Real-World Systems: Video surveillance, Satellite image processing, Digital camera pipelines, Medical imaging, Autonomous driving – Deployment considerations: Model compression, Edge inference, Efficiency: Sparse representations, Optimization – Emerging trends and research directions.

Text Books

1. Charu C. Aggarwal, Neural Networks and Deep Learning: A Textbook, Second edition, Springer, 2023.
2. Richard Szeliski, Computer Vision: Algorithms and Applications, Second edition, Springer, 2021.
3. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press, 2016.

Reference Books

1. Simon Prince, Computer Vision: Models, Learning, and Inference, First Edition, Cambridge University Press, 2012.
2. Tom Mitchell, Machine Learning, First Edition, McGraw-Hill, 2017.
3. David Forsyth, Jean Ponce, Computer Vision: A Modern Approach, Second Edition, Pearson Education, 2002.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	**Fundamentals of Deep Learning and Computer Vision**	**(6)**
1.1	Computer Vision: Tasks (Classification, Detection, Segmentation), Applications, Challenges	1
1.2	Similarity indices (Superpixel between images), Tools and libraries (TensorFlow basics, OpenCV, Scikit-image, MediaPipe, Blaze)	1
1.3	Deep learning: neurons, Activation functions, Loss functions	2
1.4	Training neural networks: Backpropagation, Optimization (SGD, Adam, RMSProp), Regularization	2
2	**Convolutional Neural Networks (CNNs)**	**(6)**
2.1	CNN architecture and operations: Convolution, Pooling, Stride, Padding	1
2.2	Classical CNN models: LeNet, AlexNet, VGG, ResNet - Transfer learning and fine-tuning	1
2.3	Data augmentation & model evaluation	1
2.4	Object Detection Models: R-CNN, Fast R-CNN, Faster R-CNN, YOLO, SSD	1
2.5	Object Segmentation: Semantic Vs Instance Segmentation, FCNs, U-Net, Mask R-CNN	1
2.6	Pre-GAN Applications: Posture Recognition, Gesture Recognition, Anomaly Detection	1
3	**Advanced Vision Models, Transformers and Multimodal Systems**	**(6)**
3.1	Advanced Vision Architecture - Vision Transformer (ViT), Inverse Convolution/Deconvolution, CUCN, Sparse Optimization Techniques	2
3.2	Sequential and Video Models - 3D CNN, ConvLSTM,	2

	Optimal Flow, Action Recognition, Video Classification, Video streaming	
3.3	Multimodal Vision and Language Integration – Image Captioning, BRAD Model, Visual Question Answering (VQA), CLIP, BLIP	2
4	**Generative Models and Image Synthesis**	**(6)**
4.1	Autoencoders and Variational Autoencoders (VAEs)	2
4.2	Generative Adversarial Networks (GANs): DCGAN, CycleGAN, StyleGAN	2
4.3	Applications: image enhancement, super-resolution, image-to-image translation – Diffusion models: Concept, Comparison with GANs, Applications in Generation and Restoration	2
5	**Model Explainability**	**(6)**
5.1	Interpretability Techniques: Saliency Maps, Grad-CAM, Layer-wise Relevance Propagation (LRP)	2
5.2	Use cases of Explainability: Medical Diagnosis, Autonomous Vehicles	2
5.3	Comparison and Limitations of Explainability Methods	2
6	**Applications and Real-World Systems**	**(6)**
6.1	Video Surveillance, Satellite Image Processing, Digital Camera Pipelines, Medical Imaging, Autonomous Driving	2
6.2	Deployment Considerations: Model Compression, Edge Inference, Efficiency: Sparse Representations, Optimization	2
6.3	Emerging Trends and Research Directions	2
Total		**36**

Course Designer(s):

1. Dr. N. Sheerin Sitara, AP, CSE, TCE nsscse@tce.edu

24AM630	HUMAN-AI INTERACTION	Category	L	T	P	Credits
		PC	3	0	0	3

Preamble

The Human-AI Interaction course introduces students to the interdisciplinary study of how humans engage with artificial intelligence systems. It emphasizes the importance of designing AI that is ethical, interpretable, user-centered, and safe, enabling students to build systems that align with human needs, values, and behaviours

Prerequisite

Basic knowledge of Artificial Intelligence (AI) concepts.

Course Outcomes

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

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Describe the fundamental understanding of Human interaction with AI Systems	TPS 2	70	75
CO2	Explain the components and design standards for interactive AI systems	TPS 2	70	75
CO3	Apply human-centric design factors in AI systems.	TPS 3	70	70
CO4	Build interaction models for human-AI collaboration using appropriate frameworks.	TPS 3	70	70
CO5	Demonstrate AI system behaviour through confidence estimation, uncertainty visualization, model customization, and user feedback mechanisms.	TPS 3	70	70
CO6	Solve real-world applications of Human-AI Interaction in domains like healthcare, autonomous vehicles, smart homes, and robotics.	TPS 3	70	70

The course outcome will be assessed through Mini-project.

Mapping with Programme Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L	-	-	L	L	-	M	M	M	-	L	L	L	M
CO2	M	L	-	-	L	L	-	M	M	M	-	L	L	L	M
CO3	S	M	L	-	M	L	-	M	M	M	-	M	M	M	M
CO4	S	M	L	-	M	L	-	M	M	M	-	M	M	M	M
CO5	S	M	L	-	M	L	-	M	M	M	-	M	M	M	M
CO6	S	M	L	L	M	L	L	M	M	M	L	M	M	M	M

S- Strong; M- Medium; L- Low

Assessment Pattern

CO	CAT1						CAT2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	20	-	-	-	-	-	-	-	-	-	-	2	5	-	-	-	-
CO2	10	20	-	-	-	-	-	-	-	-	-	-	2	5	-	-	-	-
CO3	-	20	20	-	-	-	-	-	-	-	-	-	2	5	15	-	-	-
CO4	-	-	-	-	-	-	10	15	15	-	-	-	2	5	15	-	-	-
CO5	-	-	-	-	-	-	10	15	15	-	-	-	2	5	15	-	-	-
CO6	-	-	-	-	-	-	-	10	10	-	-	-	-	5	15	-	-	-

CO	Assignment 1			Assignment 2		
TPS Scale	1	2	3	1	2	3
CO1	-	-	-	-	-	-
CO2	-	-	-	-	-	-
CO3	-	-	100	-	-	-
CO4	-	-	-	-	-	-
CO5	-	-	-	-	-	-
CO6	-	-	-	-	-	100

* Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction to Human-AI Interaction: History, Goals, Tools and Research Trends, Importance of user-oriented approach, Interface – Elements and methods, Relationship between HCI and AI.

Human-AI Interaction: Heuristics and biases in human decision making, Interaction – Principles, elements, devices, technologies and standards, Paper prototyping, Contemporary view of Human-AI Interaction.

Human-Centric Design Factors: Ethics and Safety - Ethics, Fairness and Equity, Privacy, Transparency, Bias, Reliability and Trustworthiness, Experiences -Needs, Perceptions and experiences, Human-AI Collaboration – Nature, goals and interaction patterns

Human-AI Collaboration and Interaction Design: Human-AI collaboration models: Assistive, augmentative, and co-creative AI systems - Design considerations for human-AI collaboration tools - Interaction patterns for collaborative systems: Feedback loops, shared control, and decision support

Human-AI Design: Model confidence and uncertainty, Model customization, Model behaviour, Refinement and correction, Elicitation and User feedback.

Applications: AI-based personalized systems, voice assistants,Large Language Models (LLMs) for interactive AI systems - Prompt Engineering: Concepts, strategies, and use cases -Case studies - Healthcare - Autonomous Vehicle - Smart homes - Robotics - Medical diagnostics.

Learning Resources

1. Rogers Yvonne, Sharp Helen and Preece Jenn, “Interactions Design: Beyond Human-Computer Interaction”, Sixth Edition, John Wiley & Sons, 2023.
2. Yang Li and Otmar Hilliges, “Artificial Intelligence for Human Computer Interaction: A Modern Approach”, First Edition, Springer, 2021.

3. Alan Dix, Janet E.Finlay, Gregory D.Abowd and Russell Beale, “Human- Computer Interaction”, Third Edition, Prentice-Hall, Inc, 2009.
4. Stuart Russell and Peter Norvig, “Artificial Intelligence: A Modern Approach”, Fourth Edition, Pearson, 2021.
5. Henry A.Kissinger, Eric Schmidt, Daniel Huttenlocher, “The Age of A.I and our Human Future”, First Edition, John Murray, 2021.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	**Foundations**	**(6)**
1.1	Introduction - History, Goals, Tools and Research Trends	1
1.2	Importance of user-oriented approach	2
1.3	Interface - Elements and methods	2
1.4	Relationship between HCI and AI	1
2	**Components of Human-AI Interaction**	**(6)**
2.1	Heuristics and biases in human decision making, Interaction	2
2.2	Interaction - Principles, elements, devices, technologies and standards	2
2.3	Paper prototyping, Contemporary view of Human-AI Interaction.	2
3	**Human-Centric Design Factors**	**(6)**
3.1	Ethics and Safety - Ethics, Fairness and Equity, Privacy, Transparency, Bias, Reliability and Trustworthiness	2
3.2	Experiences -Needs, Perceptions and experiences	2
3.3	Human-AI Collaboration - Nature, goals and interaction patterns	2
4	**Human-AI Collaboration and Interaction Design**	**(6)**
4.1	Human-AI collaboration models: Assistive, augmentative, and co-creative AI systems	2
4.2	Design considerations for human-AI collaboration tools	2
4.3	Interaction patterns for collaborative systems: Feedback loops, shared control, and decision support	2
5	**Human-AI Design**	**(6)**
5.1	Model confidence and uncertainty, Modelcustomization, Model behaviour	3
5.2	Refinement and correction, Elicitation and User feedback	3
6	**Ethics and Safety**	**(6)**
6.1	AI-based personalized systems, voice assistants	2
6.2	Case studies – Healthcare, Autonomous Vehicle	2
6.3	Case studies - Smart homes – Robotics - Medical diagnostics.	2
Total		**36**

Course Designer(s):

1. Dr. N. Sheerin Sitara, AP, CSE, TCE nsscse@tce.edu

24AM670	DEEP LEARNING LAB	Category	L	T	P	Credits
		PC	0	0	2	1

Preamble

The Deep Learning Lab offers students practical exposure to the core principles and applications of deep neural networks, enabling them to build, train, and evaluate models across various tasks such as image classification, object detection, and natural language processing. This lab bridges theoretical concepts with real-world implementation, fostering critical thinking and problem-solving skills essential for research and industry in the field of artificial intelligence.

Prerequisite

Basic knowledge of Problem solving and Programming concepts

Course Outcomes

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

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Demonstrate Deep Learning Models from Scratch through Backpropagation, and Optimization Techniques	TPS 3	70	70
CO2	Develop different Image Processing models using CNNs and Transfer Learning for Segmentation, Object Detection, and Classification	TPS 3	70	70
CO3	Practice Natural Language Processing tasks using with Sequence Models, Next Word Prediction, Named Entity Recognition and Ranking.	TPS 3	70	70
CO4	Implement Document Similarity, Plagiarism Detection, and Language Translation Using Deep Learning Models	TPS 3	70	70
CO5	Apply advanced techniques in Speech and Video Processing with Deep Learning Libraries	TPS 3	70	70
CO6	Perform Time-Series Forecasting and Stock Price Prediction Using GRU and Real-World Data	TPS 3	70	70

Mapping with Programme Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L	-	M	-	-	M	M	M	-	M	M	M	M
CO2	S	M	L	-	M	L	-	M	M	M	-	M	M	M	M

CO3	S	M	L	-	M	-	-	M	M	M	-	M	M	M	M
CO4	S	M	L	-	M	-	-	M	M	M	-	M	M	M	M
CO5	S	M	L	-	M	L	-	M	M	M	-	M	M	M	M
CO6	S	M	L	-	M	M	M	M	M	M	-	M	M	M	M

S- Strong; M- Medium; L- Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Model Examination	Terminal Examination
Remember	-	-
Understand	-	-
Apply	100	100
Analyse	-	-
Evaluate	-	-
Create	-	-

List of Experiments/Activities with CO Mapping

Ex.NO	Experiment	CO
1	Build a Feedforward Neural Network from Scratch and implement Backpropagation with Various Activation Functions	CO1
2	Implement binary and multiclass image classification using CNN with optimization techniques	CO1
3	Develop a python program for image segmentation, object detection and classification using transfer learning approach	CO2
4	Perform Image Generation with Generative Adversarial Networks (GANs)	CO2
5	Implement and train an autoencoder on the MNIST dataset for different latent spaces.	CO2
6	Apply an HMM to perform part-of-speech tagging on a text corpus	CO3
7	Perform next word prediction, sentence generation using LSTM, GRU and BERT	CO3
8	Illustrate similar document retrieval and plagiarism check using BERT, K-means clustering algorithm and similarity index.	CO4
9	Perform Language Translation using Sequence-to-Sequence (Seq2Seq)models and Named Entity Recognition (NER) using pre-trained language models	CO4
10	Rank a set of text documents by similarity to the given query using TF-IDF techniques.	CO4
11	Demonstrate video and speech processing related experiments using deep learning libraries.	CO5
12	Implement a GRU-based model for predicting stock prices on time-series data.	CO6

Learning Resources

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press, 2016.
2. Charu C. Aggarwal, Neural Networks and Deep Learning: A Textbook, Second edition, Springer, 2023
3. Tom Mitchell, Machine Learning, First Edition, McGraw-Hill, 2017.
4. http://www.hpc.iitkgp.ac.in/pdfs/AI_HPC.pdf
5. https://people.iitism.ac.in/~download/lab%20manuals/mathandcomp/Neural%20Networks%20and%20Deep%20Learning%20Practical.pdf

Course Designer(s):

1. Dr.N. Sheerin Sitara, AP, CSE, TCE nsscse@tce.edu

B.E. COMPUTER SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENCE and MACHINE LEARNING) DEGREE PROGRAMME

CURRICULUM AND DETAILED SYLLABI

FOR

SEVENTH & EIGHTH SEMESTERS

FOR THE STUDENTS ADMITTED IN THE

ACADEMIC YEAR 2024 - 25 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided Autonomous Institution affiliated to Anna University)
MADURAI – 625 015, TAMILNADU

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

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
BECSE(AIML) Degree Programme

COURSES OF STUDY
(For the candidates admitted from 2024-25 onwards)

SEVENTH SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			Credits
			L	T	P	
THEORY						
24AM710	Emotional Artificial Intelligence	ESC	2	-	-	2
24AM720	Autonomous Agents	PC	3	-	-	3
24AMPX0	Programme Elective	PC	3	-	-	3
24AMPX0	Programme Elective	PC	3	-	-	3
24AMPX0	Programme Elective	PC	3	-	-	3
24AMPX0	Programme Elective	PC	3	-	-	3
PRACTICAL						
24AM770	Autonomous Agents Lab	PC	-	-	2	1
22CS790	Project-III	PW	-	-	6	3
Total			**17**	**-**	**8**	**21**

EIGHTH SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			credits
			L	T	P	
THEORY						
24AMPX0	Programme Elective	PE	3	-	-	3
24AMPX0	Programme Elective	PE	3	-	-	3
PROJECT COURSE						
22CS890	Project - IV	PW	-	-	6	3
Total			**6**	**-**	**6**	**9**

BSC : Basic Science Courses
ESC : Engineering Science Courses
PC : Professional Core Courses

L : Lecture
T : Tutorial
P : Practical

Note:
1 Hour Lecture is equivalent to 1 credit
1 Hour Tutorial is equivalent to 1 credit
2 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
BECSE(AIML) Degree Programme

SCHEME OF EXAMINATIONS
(For the candidates admitted from 2024-25 onwards)

SEVENTH SEMESTER

S.No.	Course 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
THEORY								
1	24AM710	Emotional Artificial Intelligence	3	40	60	100	27	50
2	24AM720	Autonomous Agents	3	40	60	100	27	50
3	24AMPX0	Programme Elective	3	40	60	100	27	50
4	24AMPX0	Programme Elective	3	40	60	100	27	50
5	24AMPX0	Programme Elective	3	40	60	100	27	50
6	24AMPX0	Programme Elective	3	40	60	100	27	50
PRACTICAL								
7	24AM770	Autonomous Agents Lab	3	60	40	100	18	50
8	22CS790	Project-III	3	40	60	100	27	50

EIGHTH SEMESTER

S.No.	Course 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
THEORY								
1	24AMPX0	Programme Elective	3	40	60	100	27	50
2	24AMPX0	Programme Elective	3	40	60	100	27	50
PROJECT COURSE								
3	22CS890	Project - IV	3	40	60	100	27	50

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

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

24AM710	EMOTIONAL ARTIFICIAL INTELLIGENCE	Category	L	T	P	Credit
		ES	2	0	0	2

Preamble

This course delves into the interdisciplinary principles of emotional intelligence in computing, integrating concepts from affective science, psychology, machine learning, and human-computer interaction. Students will acquire the theoretical understanding and practical skills needed to design and build emotionally intelligent systems that can recognize and respond to human emotions across various modalities, including speech, facial expressions, and textual input.

Prerequisite

Knowledge of programming, psychology, and machine learning concepts.

Course Outcomes

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

	Course Outcomes	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Explain foundational theories of emotion and their relevance to human-centered system design.	TPS2	70	85
CO2	Design techniques to extract and utilize acoustic and facial features for recognizing emotions from voice and facial expressions.	TPS3	70	80
CO3	Apply emotion synthesis and facial expression recognition systems in real-world applications	TPS3	70	80
CO4	Develop text-based interactive systems capable of detecting and responding to user emotions using emotionally intelligent techniques.	TPS3	70	80
CO5	Evaluate emotional states by integrating and analyzing multimodal data using machine learning approaches.	TPS3	70	80
CO6	Design and develop emotionally empathetic agents, robots, or machines that foster natural, meaningful, and effective interactions with human users	TPS3	70	80

Mapping with Programme Outcomes

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PO1 2	PS O1	PS O2	PSO3
C(	M	L				L		M	M	L	L	L	L		L
CO2.	S	M	L		M	L		M	M	M	M	M	M	L	M
CO3.	S	M	L		M	L		M	M	M	M	M	M	L	M
CO4.	S	M	L		M	L		M	M	M	M	M	M	L	M
CO5.	S	M	L		M	M	L	M	M	M	M	M	M	M	M
CO6.	S	M	L		M	M	L	M	M	M	M	M	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						Ass 1	CAT2						Ass2	Terminal					
TPS Scale	1	2	3	4	5	6		1	2	3	4	5	6		1	2	3	4	5	6
CO1	10	10	-	-	-	-	30	-	-	-	-	-	-	-	2	5	-	-	-	-
CO2	10	10	20	-	-	-	35	-	-	-	-	-	-	-	2	5	12	-	-	-
CO3	10	10	20	-	-	-	35	-	-	-	-	-	-	-	2	5	12	-	-	-
CO4	-	-	-	-	-	-	-	5	10	20	-	-	-	20	2	5	12	-	-	-
CO5	-	-	-	-	-	-	-	5	10	15	-	-	-	20	2	5	12	-	-	-
CO6	-	-	-	-	-	-	-	10	10	15	-	-	-	60		5	12	-	-	-

*****Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Emotion Theory: Introduction to Affective Computing, Emotion Psychology, Emotion Theory, Emotion Design, Affect Elicitation.

Emotions in voice and facial expressions: Acoustic feature extraction, Feature Normalization, Speech Emotion Recognition, Emotion Speech Synthesis. Facial Expression Recognition, Facial feature extraction, group level emotion, Applications of Facial Expression Recognition.

Emotions in Text: Features – Bag-of-Words, Term Frequency, Inverse Document Frequency, Methods – Text based emotion recognition, Text representation with Learning – Word2Vec, FastText, GloVe, and Conversation based analysis.

Multimodal Emotion Recognition: Motivation, Data Collection, Feature Extraction, Stream Fusion – Late Fusion, Early Fusion, Slow Fusion, Multimodal Analysis – Audio-video emotion challenge, Feature-level fusion, Decision-level fusion.

Emotional Empathy in Agents/Machines/Robots: Empathy and Empathetic Agents, Development of Artificial Empathy, Empathy in Virtual and Robotic Agents, Empathy beyond Emotional states, Evaluation of Empathetic Response.

Text Books

1. Rosalind W. Picard, "Affective Computing", Second Edition, MIT Press, 2000.
2. Keith Davis, "Human behavior at work Human relations and Organizational Behavior", Tata McGraw Hill, New Delhi,1982.
3. Jenny Preece, Helen Sharp, Yvonne Rogers," Interaction Design: Beyond Human-Computer Interaction",Fifth Edition, Wiley,2019.

Reference Books

1. Jeanne Segal ,"The Language of Emotional Intelligence: The Five Essential Tools for Building Powerful and Effective Relationships ", McGraw-Hill Contemporary,2008.
2. The Brain and Emotional Intelligence: New Insights :Daniel Goleman HBR's 10 Must Reads on Emotional Intelligence, Harvard Business Review Press, 2015.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	**Emotion Theory**:	

Module No.	Topic	No. of Periods
1.1	Introduction to Affective Computing	1
1.2	Emotion Psychology	1
1.3	Emotion Theory	1
1.4	Emotion Design, Affect Elicitation	1
2	**Emotions in voice and facial expressions**	
2.1	Acoustic feature extraction, Feature Normalization, Applications of Facial Expression Recognition.	2
2.2	Facial feature extraction, group level emotion, Speech Emotion Recognition	1
2.3	Emotion Speech Synthesis ,Facial Expression Recognition, Facial feature extraction	1
2.4	group level emotion, Applications of Facial Expression Recognition	2
3	**Emotions in Text**	
3.1	Features – Bag-of-Words, Term Frequency	1
3.2	Inverse Document Frequency, Methods – Text based emotion recognition	1
3.3	Text representation with Learning	1
3.4	Word2Vec, FastText, GloVe, Conversation based analysis.	1
4	**Multimodal Emotion**	
4.1	Recognition Motivation, Data Collection, Feature Extraction	1
4.2	Stream Fusion – Late Fusion, Early Fusion, Slow Fusion, Multimodal Analysis	2
4.3	Audio-video emotion challenge	1
4.4	Feature-level fusion, Decision-level fusion	1
5	**Emotional Empathy in Agents/Machines/Robots**	
6.1	Empathy and Empathetic Agents, Development of Artificial Empathy	2
6.2	Empathy in Virtual and Robotic Agents	1
6.3	Empathy beyond Emotional states	1
6.4	Evaluation of Empathetic Response.	1

Module No.	Topic	No. of Periods
	Total	**24**

Course Designer(s):

1. Mrs.S.Jeniba sjacse@tce.edu

24AM720 AUTONOMOUS AGENTS

Category	L	T	P	Credit
PC	3	0	0	3

Preamble

To introduce students to autonomous and multi-agent systems, focusing on agent architectures, communication, coordination and learning. Students will apply these concepts to real-world problems and implement agent-based solutions using tools like JASON.

Prerequisite

- Basic knowledge of AI concepts such as search, reasoning, and planning.

Course Outcomes

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

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency in %	Expected Attainment Level %
CO1	Infer the core concepts and architectures of intelligent agents and their applications in expert and distributed systems.	TPS 2	70	85
CO2	Illustrate competitive and cooperative interactions within multi-agent systems	TPS 3	70	80
CO3	Design and implement agent interaction protocols to explore KQML and KIF in agent systems	TPS 3	70	80
CO4	Apply distributed planning and execution in multi-agent systems to solve constraint satisfaction problems and path-finding problems in a distributed environment.	TPS 3	70	80
CO5	Implement reinforcement learning techniques for multi-agent systems to identify challenges in multi-agent systems learning.	TPS 3	70	80
CO6	Design abstractions using the Program organizational and environmental aspects of multi-agent systems.	TPS 3	70	80

Mapping with Programme Outcomes

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
CO1	M	L			L	L	L	M	M	L		M	L	L	M
CO2	S	M	L		M	M	**L**	M	M	M		M	M	M	M
CO3	S	M	L		M	M	**L**	M	M	M		M	M	M	M
CO4	S	M	L		M	M	**L**	M	M	M		M	M	M	M
CO5	S	M	L		M	M	**L**	M	M	M		M	M	M	M
CO6	S	M	L		M	M	**L**	M	M	M	L	M	M	M	M

S- Strong; M-Medium; L-Low

Assessment Pattern

CO	CAT1						CAT2						Assignment 1						Assignment 2						Terminal					
TPS Scale	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6	1	2	3	4	5	6
CO1	10	20												30											2	6				
CO2	10	10	10											30											2	6	10			
CO3	10	10	20												40										2	6	10			
CO4							5	10	15												30				2	6	10			
CO5							5	15	15												30				2	6	10			
CO6							5	15	15												40				4	6	10			

*****Terminal examination should cover all Course Outcomes in the appropriate TPS Scale level.

Syllabus

Introduction: Agents and Objects - Expert Systems - Distributed Systems - typical application areas for agent systems - Abstract architectures for Intelligent Agent-Reasoning Agents - Agents as Reactive Systems - Hybrid Agents - Layered Agents.

Multi-Agent Systems: classifying multi-agent interactions – Multiagent Interactions – Competative and zero-sum interactions; cooperation - the Prisoner's dilemma and Axelrod's experiments;

Agent Communication: Speech acts - Knowledge Query and Manipulation Language (KQML) - Knowledge Interchange Format (KIF) – Ontologies - coordination languages - Agent Interaction Protocols

Distributed Problem Solving and Planning: Task Sharing - Result Sharing- Distributed Planning and Execution - Constraint Satisfaction Problem - Path-Finding Problem - Evaluation Criteria – Voting – Auctions

Multiagent Systems Learning: Challenges – Reinforcement Learning for Multiagent Systems – Coordination Prior to Local Planning – Local Planning Prior to Coordination - Decision-Theoretic Multiagent Planning – Multiagent Execution

MultiagentSystemsProgramming: Cognitive Primitives – Abstractions in the MAOP Paradigm – JASON - BDI Implementations – Organization and Environment Programming - Logical Background.

Text Book

1. Gerhard Weiss, "Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence", MIT Press, 2013, 2nd edition
2. Michael Woolridge, "An Introduction to Multiagent Systems", Wiley & Sons Wiley-ISTE, 2016.

Reference Books& web resources

1. Jiming Liu, "Autonomous Agents and Multi-Agent Systems Explorations in Learning,Self-Organization and Adaptive Computation" World Scientific Publishing, 2011.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Periods
1	**Introduction**	
1.1	Agents and Objects - Expert Systems	1
1.2	Distributed Systems - typical application areas for agent systems	2
1.3	Abstract architectures for Intelligent Agents	1
1.4	Reasoning Agents - Agents as Reactive Systems	2

Module No.	Topic	No. of Periods
1.5	Hybrid Agents - Layered Agents	1
2	**Multi-Agent Systems**	
2.1	Classifying multi-agent interactions- Multiagent Interactions	2
2.2	Competative and zero-sum interactions	1
2.3	Cooperation - the Prisoner's dilemma	1
2.4	Axelrod's experiments	1
3	**Agent Communication**	
3.1	Speech acts - Knowledge Query and Manipulation Language (KQML)	2
3.2	Knowledge Interchange Format (KIF)	1
3.3	Ontologies	1
3.4	Coordination languages - Agent Interaction Protocols	2
4	**Distributed Problem Solving and Planning**	
4.1	Task Sharing - Result Sharing	1
4.2	Distributed Planning and Execution	1
4.3	Constraint Satisfaction Problem	1
4.4	Path-Finding Problem - Evaluation Criteria	2
4.5	Voting – Auctions	1
5	**Multiagent Systems Learning**	
5.1	Challenges – Reinforcement Learning for Multiagent Systems	2
5.2	Coordination Prior to Local Planning	1
5.3	Local Planning Prior to Coordination	1
5.4	Decision-Theoretic Multiagent Planning	1
5.5	Multiagent Execution	1
6	**MultiagentSystemsProgramming**	
6.1	Cognitive Primitives – Abstractions in the MAOP Paradigm	2
6.2	JASON	1
6.3	BDI Implementations – Organization and Environment Programming	2
6.4	Logical Background.	1
	Total	36

Course Designer(s):

1. Mrs.S.Saradha, Assistant Professor, CSE ssacse@tce.edu,

24AM770	AUTONOMOUS AGENTS LAB	Category	L	T	P	C
		PC	0	0	2	1

Preamble

This lab courses trains students to design intelligent agents and multi-agent systems for complex problem-solving. It covers logic programming, agent architectures, communication protocols, and reinforcement learning. Students apply AI techniques to build cooperative, learning, and decision-making agents. The course prepares them for real-world challenges in software engineering and AI.

Prerequisite

Knowledge in Python and/or Prolog

ourse Outcomes

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

CO	Course Outcome	TCE Proficiency Scale	Expected Proficiency (%)	Expected Attainment Level (%)
CO1	Design and implement a simulation environment to showcase key differences in decision-making processes and analyze a distributed system with interacting agents working toward a common goal.	TPS3	70	75
CO2	Design and implement a layered agent architecture with distinct perception, action and planning modules. Apply logic programming languages like Prolog to build reasoning agents.	TPS3	70	75
CO3	Implement the Prisoner's Dilemma game, analyze various agent strategies, and design simulations for multi-agent interactions.	TPS3	70	75
CO4	Implement a simple communication protocol using speech acts or KQML/KIF and design a distributed algorithm for solving a CSP using MAS.	TPS3	70	75
CO5	Apply reinforcement learning techniques to train agents for implementing distributed algorithms in grid-world path-finding scenarios.	TPS3	70	75
CO6	Develop a small ontology for a specific domain using an appropriate language, and build a simple BDI agent using the Jason programming language.	TPS3	70	75

Mapping with Programme Outcomes

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	S	M	L		M	M	L	M	M	M	L	M	M	M	M
CO2	S	M	L		M	M	L	M	M	M	L	M	M	M	M
CO3	S	M	L		M	M	L	M	M	M	L	M	M	M	M
CO4	S	M	L		M	M	L	M	M	M	L	M	M	M	M
CO5	S	M	L		M	M	L	M	M	M	L	M	M	M	M
CO6	S	S	M	L	M	M	L	M	M	M	L	M	M	M	M

S- Strong; M-Medium; L-Low

List of Experiments/Activities with CO Mapping

Sl.No	Experiments	CO
1)	Design a simple simulation environment where an agent interacts with various objects. Analyze the differences in behaviour and decision-making between agents and objects.	CO1
2)	Simulate a multi-agent consensus mechanism using voting or weighted agreement models. Explore cooperative agent decision-making and evaluate interaction strategies.	CO1
3)	Implement a reasoning agent using Prolog or another logic programming language. Solve logic puzzles or answer questions based on a knowledge base.	CO2
4)	Design prompt-based autonomous agents using LangChain. Implement layered architecture for perception, planning, and action using Python**.**	CO2
5)	Design a simulation environment where agents can cooperate or compete for resources. Analyze the emergent behaviour and compare different interaction strategies.	CO3
6)	Implement the Prisoner's Dilemma game and analyze different strategies using Axelrod's simulations. Discuss the implications for cooperation in MAS	CO3
7)	Implement a simple communication protocol between agents using structured messages (e.g., JSON-RPC or LangChain tool usage). Implement agents that can exchange information and coordinate their actions	CO4
8)	Design and implement a distributed algorithm for solving a CSP using MAS. Analyze the communication and coordination required between agents.	CO4
9)	Implement a distributed algorithm for path-finding in a grid-world environment. Compare different algorithms and analyze their performance.	CO5
10)	Train agents using reinforcement learning techniques to achieve specific goals in a multi-agent environment. Analyze the learning process and performance.	CO5

11)	Implement a simple BDI agent using the Jason programming language. Design a scenario where the agent interacts with other agents and makes decisions based on its beliefs, desires, and intentions.	CO6
12)	Construct and query a domain-specific knowledge graph using Python (rdflib/Neo4j). Analyze the benefits of structured knowledge for agent reasoning.	CO6

Learning Resources

1. Stuart Russell and Peter Norvig. Artificial Intelligence: A Modern Approach, 4th Edition, Prentice Hall, Feb 2020.
2. Richard S. Sutton and Andrew G. Barto, “Reinforcement Learning: An Introduction,” MIT Press, 2018.

Course Designers

1. Mrs.S.Saradha, APCSE ssacse@tce.edu