SEMESTER VII

CST401	ARTIFICIAL INTELLIGENCE	CATEGORY	L	T	P	CREDIT
		PCC	2	1	0	3

Preamble: The course aims to introduce the fundamental principles of intelligent systems to students. This involves ideas about the characteristics of intelligent systems, knowledge representation schemes, logic and inference mechanisms. The course helps the learner to understand the design of self learning systems along with some of their typical applications in the emerging scenario where the business world is being transformed by the progress made in machine learning.

Prerequisite : NIL

Course Outcomes: After the completion of the course the student will be able to

CO#	CO
CO1	Explain the fundamental concepts of intelligent systems and their architecture. **(Cognitive Knowledge Level: Understanding)**
CO2	Illustrate uninformed and informed search techniques for problem solving in intelligent systems. **(Cognitive Knowledge Level: Understanding)**
CO3	Solve Constraint Satisfaction Problems using search techniques. **(Cognitive Knowledge Level: Apply)**
CO4	Represent AI domain knowledge using logic systems and use inference techniques for reasoning in intelligent systems. **(Cognitive Knowledge Level: Apply)**
CO5	Illustrate different types of learning techniques used in intelligent systems **(Cognitive Knowledge Level: Understand)**

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	✓											
CO2	✓	✓										✓
CO3	✓	✓	✓	✓								✓
CO4	✓	✓	✓	✓								✓
CO5	✓	✓			✓							✓

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	**PO7**	Environment and Sustainability
PO2	Problem Analysis	**PO8**	Ethics
PO3	Design/Development of solutions	**PO9**	Individual and team work
PO4	Conduct investigations of complex problems	**PO10**	Communication
PO5	Modern tool usage	**PO11**	Project Management and Finance
PO6	The Engineer and Society	**PO12**	Life long learning

Assessment Pattern

Bloom's Category	**Continuous Assessment Tests**		**End Semester Examination Marks (%)**
	Test 1 (%)	**Test 2 (%)**	
Remember	30	30	30
Understand	60	30	40
Apply	20	40	30
Analyze			
Evaluate			
Create			

Mark Distribution

Total Marks	**CIE Marks**	**ESE Marks**	**ESE Duration**
150	50	100	3

Continuous Internal Evaluation Pattern:

Attendance **10 marks**

Continuous Assessment Tests(Average of SeriesTests1& 2) **25 marks**

Continuous Assessment Assignment **15 marks**

Internal Examination Pattern:

Each of the two internal examinations has to be conducted out of 50 marks. First series test shall be preferably conducted after completing the first half of the syllabus and the second series test shall be preferably conducted after completing remaining part of the syllabus. There will be two parts: Part A and Part B. Part A contains 5 questions (preferably, 2 questions each from the completed modules and 1 question from the partly completed module), having 3 marks for each question adding up to 15 marks for part A. Students should answer all questions from Part A. Part B contains 7 questions (preferably, 3 questions each from the completed modules and 1 question from the partly completed module), each with 7 marks. Out of the 7 questions, a student should answer any 5.

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 full questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carries 14 marks.

Syllabus

Module – 1 (Introduction)
Introduction – What is Artificial Intelligence(AI) ? The Foundations of AI, History of AI, Applications of AI. Intelligent Agents – Agents and Environments, Good behavior: The concept of rationality, nature of Environments, Structure of Agents.

Module – 2 (Problem Solving)
Solving Problems by searching-Problem solving Agents, Example problems, Searching for solutions, Uninformed search strategies, Informed search strategies, Heuristic functions.

Module - 3 (Search in Complex environments)
Adversarial search - Games, Optimal decisions in games, The Minimax algorithm, Alpha-Beta pruning. Constraint Satisfaction Problems – Defining CSP, Constraint Propagation- inference in CSPs, Backtracking search for CSPs, Structure of CSP problems.

Module - 4 (Knowledge Representation and Reasoning)
Logical Agents – Knowledge based agents, Logic, Propositional Logic, Propositional Theorem proving, Agents based on Propositional Logic. First Order Predicate Logic – Syntax and Semantics of First Order Logic, Using First Order Logic, Knowledge representation in First Order Logic. Inference in First Order Logic – Propositional Vs First Order inference, Unification and Lifting, Forward chaining, Backward chaining, Resolution.

Module - 5 (Machine Learning)

Learning from Examples – Forms of Learning, Supervised Learning, Learning Decision Trees, Evaluating and choosing the best hypothesis, Regression and classification with Linear models.

Text Book

1. Stuart Russell and Peter Norvig. Artificial Intelligence: A Modern Approach, 3rd Edition. Prentice Hall.

References

1. Nilsson N.J., Artificial Intelligence - A New Synthesis, Harcourt Asia Pvt. Ltd.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Explain about the basic types of agent programs in intelligent systems.
2. For the following activities, give a PEAS description of the task environment and characterize it in terms of the task environment properties.

   a) Playing soccer.

   b) Bidding on an item at an auction.

Course Outcome 2 (CO2):

1. Differentiate between uninformed and informed search strategies in intelligent systems.
2. Illustrate the working of Minimax search procedure.

Course Outcome 3 (CO3):

1. Solve the following crypt arithmetic problem by hand, using the strategy of backtracking with forward checking and the MRV & least-constraining-value heuristics.

$$\begin{array}{r} T\ W\ O \\ +\ T\ W\ O \\ \hline F\ O\ U\ R \end{array}$$

Course Outcome 4 (CO4):

1. Prove, or find a counter example to, the following assertion:
   If $\alpha \models \gamma$ or $\beta \models \gamma$ (or both) then $(\alpha \wedge \beta) \models \gamma$
2. For each pair of atomic sentences, find the most general unifier if it exists:
   a) P (A, B, B), P (x, y, z).
   b) Q(y, G(A, B)), Q(G(x, x), y).

Course Outcome 5 (CO5):

1. Consider the following data set comprised of three binary input attributes (A1 , A2 , and

A3) and one binary output.

Example	A_1	A_2	A_3	Output y
$\mathbf{x}_1$	1	0	0	0
$\mathbf{x}_2$	1	0	1	0
$\mathbf{x}_3$	0	1	0	0
$\mathbf{x}_4$	1	1	1	1
$\mathbf{x}_5$	1	1	0	1

Use the DECISION-TREE-LEARNING algorithm to learn a decision tree for these data. Show the computations made to determine the attribute to split at each node.

2. What is multivariate linear regression? Explain.

Model Question Paper

QP CODE:

Reg No: _____

Name: _____ **PAGES : 4**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST401

Course Name: Artificial Intelligence

Max. Marks : 100 **Duration: 3 Hours**

PART A

Answer All Questions. Each Question Carries 3 Marks

1 What is a rational agent? Explain.

2 Describe any two ways to represent states and the transitions between them in agent programs.

3 Differentiate between informed search and uninformed search.

4 Define heuristic function? Give two examples.

5 What are the components of a Constraint Satisfaction Problem? Illustrate with an example.

6 Formulate the following problem as a CSP. Class scheduling: There is a fixed number of professors and classrooms, a list of classes to be offered, and a list of possible time slots for classes. Each professor has a set of classes that he or she can teach.

7 What is a knowledge based agent? How does it work?

8. Represent the following assertion in propositional logic:
"A person who is radical (R) is electable (E) if he/she is conservative (C), but otherwise is not electable."

9 Describe the various forms of learning?

10 State and explain Ockham's razor principle **(10x3=30)**

Part B

(Answer any one question from each module. Each question carries 14 Marks)

11. (a) Explain the structure Goal-based agents and Utility-based agents with the help of diagrams. **(8)**

(b) For the following activities, give a PEAS description of the task environment and characterize it in terms of the task environment properties. **(6)**
a) Playing soccer
b) Bidding on an item at an auction.

OR

12. (a) Explain the structure Simple reflex agents and Model-based reflex agents with the help of diagrams. **(8)**

(b) Discuss about any five applications of AI. **(6)**

13. (a) Explain Best First Search algorithm. How does it implement heuristic search? **(6)**

(b) Describe any four uninformed search strategies. **(8)**

OR

14. (a) Write and explain A* search algorithm. **(6)**

(b) Explain the components of a well defined AI problem? Write the standard formulation of 8-puzzle problem. **(8)**

15. (a) (a) Solve the following crypt arithmetic problem by hand, using the strategy of backtracking with forward checking and the MRV and least-constraining-value heuristics. **(8)**

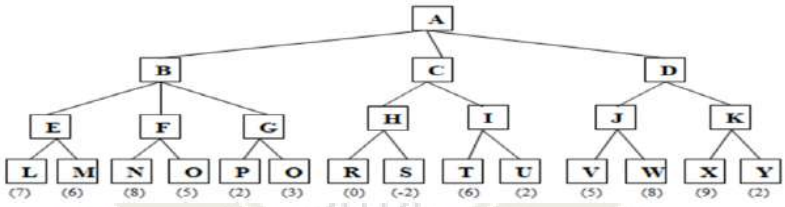
$$\begin{array}{r} T\ W\ O \\ +\ T\ W\ O \\ \hline F\ O\ U\ R \end{array}$$

(b) What is local consistency in CSP constraint propagation? Explain different types local consistencies. **(6)**

OR

16. (a) Illustrate the use of alpha-beta pruning in games. **(6)**

(b) Consider the following game tree in which static evaluation score are all from the players point of view: static evaluation score range is (+10 to -10) **(8)**



Suppose the first player is the maximizing player. What move should be chosen? Justify your answer.

17. (a) Convert the following sentences into first order logic: **(6)**

Everyone who loves all animals is loved by someone.

Anyone who kills an animal is loved by no one.

Jack loves all animals.

Either Jack or Curiosity killed the cat, who is named Tuna.

Did Curiosity kill the cat?

(b) Give a resolution proof to answer the question "Did Curiosity kill the cat? " **(8)**

OR

18. (a) Prove or find a counter example to the following assertion in propositional logic: **(6)**

If $\alpha \models (\beta \wedge \gamma)$ then $\alpha \models \beta$ and $\alpha \models \gamma$.

(b) For each pair of atomic sentences, give the most general unifier if it exists: **(8)**

Older (Father (y), y), Older (Father (x), John).

19. (a) How is best hypothesis selected from alternatives? **(8)**

(b) Explain Univariate Linear Regression. **(6)**

OR

20. (a) Consider the following data set comprised of two binary input attributes (A1 and A2) and one binary output. **(8)**

Example	A_1	A_2	Output y
x_1	1	1	1
x_2	1	1	1
x_3	1	0	0
x_4	0	0	1
x_5	0	1	0
x_6	0	1	0

Use the DECISION-TREE-LEARNING algorithm to learn a decision tree for these data. Show the computations made to determine the attribute to split at each node.

(b) Explain Linear classification with logistic regression **(6)**

Teaching Plan

No	Contents	No of Lecture Hrs (36)
Module – 1 (Introduction) (7 hrs)		
1.1	Introduction, What is Artificial Intelligence(AI)?	1
1.2	The foundations of AI, The history of AI	1
1.3	Applications of AI	1
1.4	Intelligent Agents – Agents and Environments	1
1.5	Good behavior: The concept of rationality	1
1.6	The nature of Environments	1
1.7	The structure of Agents	1
Module - 2 (Problem Solving by searching) (7 hrs)		
2.1	Solving Problems by searching-Problem solving Agents	1
2.2	Illustration of the problem solving process by agents	1
2.3	Searching for solutions	1
2.4	Uninformed search strategies:BFS, Uniform-cost search, DFS, Depth-limited search, Iterative deepening depth-first search	1
2.5	Informed search strategies: Best First search	1
2.6	Informed search strategies: A* Search	1
2.7	Heuristic functions	1
Module - 3 (Problem Solving in complex environments) (7 hrs)		
3.1	Adversarial search - Games	1
3.2	Optimal decisions in games, The Minimax algorithm	1
3.3	Alpha-Beta pruning	1
3.4	Constraint Satisfaction Problems – Defining CSP	1
3.5	Constraint Propagation- inference in CSPs	1
3.6	Backtracking search for CSPs	1
3.7	The structure of problems	1

Module - 4 (Knowledge Representation and Reasoning) (9 hrs)		
4.1	Logical Agents – Knowledge based agents and logic	1
4.2	Propositional Logic	1
4.3	Propositional Theorem proving	1
4.4	Agents based on Propositional Logic	1
4.5	First Order Predicate Logic – Syntax and Semantics of First Order Logic	1
4.6	Using First Order Logic, Knowledge representation in First Order Logic	1
4.7	Inference in First Order Logic – Propositional Vs First Order inference, Unification and Lifting	1
4.8	Forward chaining, Backward chaining	1
4.9	Resolution	1
Module - 5 (Machine Learning)(6 hrs)		
5.1	Learning from Examples – Forms of Learning	1
5.2	Supervised Learning	1
5.3	Learning Decision Trees	1
5.4	Generaliztion and overfitting	1
5.5	Evaluating and choosing the best hypothesis	1
5.6	Regression and classification with Linear models.	1

CSL411	COMPILER LAB	CATEGORY	L	T	P	CREDIT
		PCC	0	0	3	2

Preamble: This course aims to offer students hands-on experience on compiler design concepts. Students will be able to familiarize with tools such as LEX and YACC and automate different phases of a compiler. This course helps the learners to enhance the capability to design and implement a compiler.

Prerequisite: A sound knowledge in C programming, Data Structures, Formal languages and Automata Theory and Compiler design.

Course Outcomes: After the completion of the course the student will be able to

CO 1	Implement lexical analyzer using the tool LEX. **(Cognitive Knowledge Level: Apply)**
CO 2	Implement Syntax analyzer using the tool YACC. **(Cognitive Knowledge Level: Apply)**
CO 3	Design NFA and DFA for a problem and write programs to perform operations on it. **(Cognitive Knowledge Level: Apply)**
CO 4	Design and Implement Top-Down parsers. **(Cognitive Knowledge Level: Apply)**
CO 5	Design and Implement Bottom-Up parsers. **(Cognitive Knowledge Level: Apply)**
CO 6	Implement intermediate code for expressions. **(Cognitive Knowledge Level: Apply)**

Mapping of course outcomes with program outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	✓	✓	✓	✓	✓			✓		✓		✓
CO 2	✓	✓	✓	✓	✓			✓		✓		✓
CO 3	✓	✓	✓	✓				✓		✓		✓
CO 4	✓	✓	✓	✓				✓		✓		✓
CO 5	✓	✓	✓	✓				✓		✓		✓
CO 6	✓	✓	✓	✓				✓		✓		✓

Assessment Pattern

Bloom's Category	Continuous Assessment Test %	End Semester Examination %
Remember	20	20
Understand	20	20
Apply	60	60
Analyze		
Evaluate		
Create		

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	75	75	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 15 marks

Continuous Evaluation in Lab : 30 marks

Continuous Assessment Test : 15 marks

Viva-voce : 15 marks

Internal Examination Pattern: The marks will be distributed as Algorithm 30 marks, Program 20 marks, Output 20 marks and Viva 30 marks. Total 100 marks which will be converted out of 15 while calculating Internal Evaluation marks.

End Semester Examination Pattern: The marks will be distributed as Algorithm 30 marks, Program 20 marks, Output 20 marks and Viva 30 marks. Total 100 marks will be converted out of 75 for End Semester Examination.

Operating System to Use in Lab **:** Linux

Compiler/Software to Use in Lab **:** gcc, lex, yacc

Programming Language to Use in Lab **:** Ansi C

Fair Lab Record:

All Students attending the Compiler Lab should have a Fair Record. The fair record should be produced in the University Lab Examination. Every experiment conducted in the lab should be noted in the fair record. For every experiment in the fair record the right hand page should contain Experiment Heading, Experiment Number, Date of Experiment, Aim of Experiment, Details of Experiment including algorithm and Result of Experiment. The left hand page should contain a print out of the code used for the experiment and sample output obtained for a set of input.

SYLLABUS

1. Implementation of lexical analyzer using the tool LEX.
2. Implementation of Syntax analyzer using the tool YACC.
3. Application problems using NFA and DFA.
4. Implement Top-Down Parser.
5. Implement Bottom-up parser.
6. Simulation of code optimization Techniques.
7. Implement Intermediate code generation for simple expressions.
8. Implement the back end of the compiler.

PRACTICE QUESTIONS

List of Exercises/Experiments:

1. Design and implement a lexical analyzer using C language to recognize all valid tokens in the input program. The lexical analyzer should ignore redundant spaces, tabs and newlines. It should also ignore comments.
2. Implement a Lexical Analyzer for a given program using Lex Tool.
3. Write a lex program to display the number of lines, words and characters in an input text.
4. Write a LEX Program to convert the substring *abc* to *ABC* from the given input string.
5. Write a lex program to find out total number of vowels and consonants from the given input sting.
6. Generate a YACC specification to recognize a valid arithmetic expression that uses operators +, – , *,/ and parenthesis.

7. Generate a YACC specification to recognize a valid identifier which starts with a letter followed by any number of letters or digits.
8. Implementation of Calculator using LEX and YACC
9. Convert the BNF rules into YACC form and write code to generate abstract syntax tree.
10. Write a program to find ε – closure of all states of any given NFA with ε transition.
11. Write a program to convert NFA with ε transition to NFA without ε transition.
12. Write a program to convert NFA to DFA.
13. Write a program to minimize any given DFA.
14. Write a program to find First and Follow of any given grammar.
15. Design and implement a recursive descent parser for a given grammar.
16. Construct a Shift Reduce Parser for a given language.
17. Write a program to perform constant propagation.
18. Implement Intermediate code generation for simple expressions.
19. Implement the back end of the compiler which takes the three address code and produces the 8086 assembly language instructions that can be assembled and run using an 8086 assembler. The target assembly instructions can be simple move, add, sub, jump etc.

CSQ413	SEMINAR	CATEGORY	L	T	P	CREDIT
		PWS	0	0	3	2

Preamble: The course 'Seminar' is intended to enable a B.Tech graduate to read, understand, present and prepare report about an academic document. The learner shall search in the literature including peer reviewed journals, conference, books, project reports etc., and identify an appropriate paper/thesis/report in her/his area of interest, in consultation with her/his seminar guide. This course can help the learner to experience how a presentation can be made about a selected academic document and also empower her/him to prepare a technical report.

Course Objectives:

- To do literature survey in a selected area ofstudy.
- To understand an academic document from the literate and to give a presentation about it.
- To prepare a technical report.

Course Outcomes [COs] : After successful completion of the course, the students will be able to:

CO1	Identify academic documents from the literature which are related to her/his areas of interest (Cognitive knowledge level: **Apply**).
CO2	Read and apprehend an academic document from the literature which is related to her/ his areas of interest (Cognitive knowledge level: **Analyze**).
CO3	Prepare a presentation about an academic document (Cognitive knowledge level: **Create**).
CO4	Give a presentation about an academic document (Cognitive knowledge level: **Apply**).
CO5	Prepare a technical report (Cognitive knowledge level: **Create**).

Mapping of course outcomes with program outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	1	1		2	1					3
CO2	3	3	2	3		2	1					3
CO3	3	2			3			1		2		3
CO4	3				2			1		3		3
CO5	3	3	3	3	2	2		2		3		3

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	PO7	Environment and Sustainability
PO2	Problem Analysis	PO8	Ethics
PO3	Design/Development of solutions	PO9	Individual and team work
PO4	Conduct investigations of complex problems	PO10	Communication
PO5	Modern tool usage	PO11	Project Management and Finance
PO6	The Engineer and Society	PO12	Life long learning

General Guidelines

- The Department shall form an Internal Evaluation Committee (IEC) for the seminar with academic coordinator for that program as the Chairperson/Chairman and seminar coordinator & seminar guide as members. During the seminar presentation of a student, all members of IEC shall be present.
- Formation of IEC and guide allotment shall be completed within a week after the University examination (or last working day) of the previous semester.
- Guide shall provide required input to their students regarding the selection of topic/ paper.
- Choosing a seminar topic: The topic for a UG seminar should be current and broad based rather than a very specific research work. It's advisable to choose a topic for the Seminar to be closely linked to the final year project area. Every member of the project team could choose or be assigned Seminar topics that covers various aspects linked to the Project area.
- A topic/paper relevant to the discipline shall be selected by the student during the semester break.
- Topic/Paper shall be finalized in the first week of the semester and shall be submitted to the IEC.
- The IEC shall approve the selected topic/paper by the second week of the semester.
- Accurate references from genuine peer reviewed published material to be given in the report and to be verified.

Evaluation pattern

Total marks: 100, only CIE, minimum required to pass 50

Seminar Guide: 20 marks (Background Knowledge – 10 (The guide shall give deserving marks for a candidate based on the candidate's background knowledge about the topic selected), Relevance of the paper/topic selected – 10).

Seminar Coordinator: 20 marks (Seminar Diary – 10 (Each student shall maintain a seminar diary and the guide shall monitor the progress of the seminar work on a weekly basis and shall approve the entries in the seminar diary during the weekly meeting with the student), Attendance – 10).

Presentation: 40 marks to be awarded by the IEC (Clarity of presentation – 10, Interactions – 10 (to be based on the candidate's ability to answer questions during the interactive session of her/his presentation), Overall participation – 10 (to be given based on her/his involvement during interactive sessions of presentations by other students), Quality of the slides – 10).

Report: 20 marks to be awarded by the IEC (check for technical content, overall quality, templates followed, adequacy of references etc.).

CSD415	PROJECT PHASE I	CATEGORY	L	T	P	CREDIT
		PWS	0	0	6	2

Preamble: The course 'Project Work' is mainly intended to evoke the innovation and invention skills in a student. The course will provide an opportunity to synthesize and apply the knowledge and analytical skills learned, to be developed as a prototype or simulation. The project extends to 2 semesters and will be evaluated in the 7th and 8th semester separately, based on the achieved objectives. One third of the project credits shall be completed in 7th semester and two third in 8th semester. It is recommended that the projects may be finalized in the thrust areas of the respective engineering stream or as interdisciplinary projects. Importance should be given to address societal problems and developing indigenous technologies.

Course Objectives

- To apply engineering knowledge in practical problem solving.
- To foster innovation in design of products, processes or systems.
- To develop creative thinking in finding viable solutions to engineering problems.

Course Outcomes [COs] : After successful completion of the course, the students will be able to:

CO1	Model and solve real world problems by applying knowledge across domains (Cognitive knowledge level: **Apply**).
CO2	Develop products, processes or technologies for sustainable and socially relevant applications (Cognitive knowledge level: **Apply**).
CO3	Function effectively as an individual and as a leader in diverse teams and to comprehend and execute designated tasks (Cognitive knowledge level: **Apply**).
CO4	Plan and execute tasks utilizing available resources within timelines, following ethical and professional norms (Cognitive knowledge level: **Apply**).
CO5	Identify technology/research gaps and propose innovative/creative solutions (Cognitive knowledge level: **Analyze**).
CO6	Organize and communicate technical and scientific findings effectively in written and oral forms (Cognitive knowledge level: **Apply**).

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	2	1	2	2	2	1	1	1	1	2
CO2	2	2	2		1	3	3	1	1		1	1
CO3									3	2	2	1
CO4					2			3	2	2	3	2
CO5	2	3	3	1	2							1
CO6					2			2	2	3	1	1

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	PO7	Environment and Sustainability
PO2	Problem Analysis	PO8	Ethics
PO3	Design/Development of solutions	PO9	Individual and team work
PO4	Conduct investigations of complex problems	PO10	Communication
PO5	Modern tool usage	PO11	Project Management and Finance
PO6	The Engineer and Society	PO12	Lifelong learning

PROJECT PHASE I

Phase 1 Target

- Literature study/survey of published literature on the assigned topic
- Formulation of objectives
- Formulation of hypothesis/ design/ methodology
- Formulation of work plan and task allocation.
- Block level design documentation
- Seeking project funds from various agencies
- Preliminary Analysis/Modeling/Simulation/Experiment/Design/Feasibility study
- Preparation of Phase 1 report

Evaluation Guidelines & Rubrics

Total: 100 marks (Minimum required to pass: 50 marks).

- Project progress evaluation by guide: 30 Marks.
- Interim evaluation by the Evaluation Committee: 20 Marks.
- Final Evaluation by the Evaluation Committee: 30 Marks.
- Project Phase - I Report (By Evaluation Committee): 20 Marks.

(The evaluation committee comprises HoD or a senior faculty member, Project coordinator and project supervisor).

Evaluation by the Guide

The guide/supervisor shall monitor the progress being carried out by the project groups on a regular basis. In case it is found that progress is unsatisfactory it shall be reported to the Department Evaluation Committee for necessary action. The presence of each student in the group and their involvement in all stages of execution of the project shall be ensured by the guide. Project evaluation by the guide: 30 Marks. This mark shall be awarded to the students in his/her group by considering the following aspects:

Topic Selection: innovativeness, social relevance etc. (2)

Problem definition: Identification of the social, environmental and ethical issues of the project problem. (2)

Purpose and need of the project: Detailed and extensive explanation of the purpose and need of the project. (3)

Project Objectives: All objectives of the proposed work are well defined; Steps to be followed to solve the defined problem are clearly specified. (2)

Project Scheduling & Distribution of Work among Team members: Detailed and extensive Scheduling with timelines provided for each phase of project. Work breakdown structure well defined. (3)

Literature survey: Outstanding investigation in all aspects. (4)

Student's Diary/ Daily Log: The main purpose of writing daily diary is to cultivate the habit of documenting and to encourage the students to search for details. It develops the students' thought process and reasoning abilities. The students should record in the daily/weekly activity diary the day to day account of the observations, impressions, information gathered and suggestions given, if any. It should contain the sketches & drawings related to the observations made by the students. The daily/weekly activity diary shall be signed after every day/week by the guide. (7)

Individual Contribution: The contribution of each student at various stages. (7)

EVALUATION RUBRICS for PROJECT Phase I: Interim Evaluation						
No.	**Parameters**	**Marks**	**Poor**	**Fair**	**Very Good**	**Outstanding**
1-a	Topic identification, selection, formulation of objectives and/or literature survey. (Group assessment) [CO1]	10	The team has failed to come with a relevant topic in time. Needed full assistance to find a topic from the guide. They do not respond to suggestions from the evaluation committee and/or the guide. No literature review was conducted. The team tried to gather easy information without verifying the authenticity. No objectives formed yet.	The team has identified a topic. The originally selected topic lacks substance and needs to be revised. There were suggestions given to improve the relevance and quality of the project topic. Only a few relevant references were consulted/ studied and there is no clear evidence to show the team's understanding on the same. Some objectives identified, but not clear enough.	Good evidence of the group thinking and brainstorming on what they are going to build. The results of the brainstorming are documented and the selection of topic is relevant. The review of related references was good, but there is scope of improvement. Objectives formed with good clarity, however some objectives are not realistic enough.	The group has brainstormed in an excellent manner on what they were going to build. The topic selected is highly relevant, real world problem and is potentially innovative. The group shows extreme interest in the topic and has conducted extensive literature survey in connection with the topic. The team has come up with clear objectives which are feasible.
			(0 – 3 Marks)	(4 – 6 Marks)	(7 - 9 Marks)	(10 Marks)
1-b	Project Planning, Scheduling and Resource/ Tasks Identification and allocation. (Group assessment) [CO4]	10	No evidence of planning or scheduling of the project. The students did not plan what they were going to build or plan on what materials / resources to use in the project. The students do not have any idea on the budget required. The team has not yet decided on who does what. No project journal kept.	Some evidence of a primary plan. There were some ideas on the materials /resources required, but not really thought out. The students have some idea on the finances required, but they have not formalized a budget plan. Schedules were not prepared. The project journal has no details. Some evidence on task allocation among the team members.	Good evidence of planning done. Materials were listed and thought out, but the plan wasn't quite complete. Schedules were prepared, but not detailed, and needs improvement. Project journal is presented but it is not complete in all respect / detailed. There is better task allocation and individual members understand about their tasks. There is room for improvement.	Excellent evidence of enterprising and extensive project planning. Gantt charts were used to depict detailed project scheduling. A project management/version control tool is used to track the project, which shows familiarity with modern tools. All materials / resources were identified and listed and anticipation of procuring time is done. Detailed budgeting is done. All tasks were identified and incorporated in the schedule. A well-kept project journal shows evidence for all the above, in addition to the interaction with the project guide. Each member knows well about their individual tasks.
			(0 – 3 Marks)	(4 – 6 Marks)	(7 - 9 Marks)	(10 Marks)
Phase 1 Interim Evaluation Total Marks: 20						

EVALUATION RUBRICS for PROJECT Phase I: Final Evaluation						
Sl. No.	**Parameters**	**Marks**	**Poor**	**Fair**	**Very Good**	**Outstanding**
1-c	Formulation of Design and/or Methodology and Progress. (Group assessment) [CO1]	5	None of the team members show any evidence of knowledge about the design and the methodology adopted till now/ to be adopted in the later stages. The team has not progressed from the previous stage of evaluation.	The students have some knowledge on the design procedure to be adopted, and the methodologies. However, the team has not made much progress in the design, and yet to catch up with the project plan.	The students are comfortable with design methods adopted, and they have made some progress as per the plan. The methodologies are understood to a large extent.	Shows clear evidence of having a well- defined design methodology and adherence to it. Excellent knowledge in design procedure and its adaptation. Adherence to project plan is commendable.
			(0 – 1 Marks)	(2 – 3 Marks)	(4 Marks)	(5 Marks)
1-d	Individual and Teamwork Leadership (Individual assessment) [CO3]	10	The student does not show any interest in the project activities, and is a passive member.	The student show some interest and participates in some of the activities. However, the activities are mostly easy and superficial in nature.	The student shows very good interest in project, and takes up tasks and attempts to complete them. Shows excellent responsibility and team skills. Supports the other members well.	The student takes a leadership position and supports the other team members and leads the project. Shows clear evidence of leadership.
			(0 – 3 Marks)	(4 – 6 Marks)	(7 - 9 Marks)	(10 Marks)
1-e	Preliminary Analysis/ Modeling / Simulation/ Experiment / Design/ Feasibility study [CO1]	10	The team has not done any preliminary work with respect to the analysis/modeling/ simulation/experiment/desig n/feasibility study/ algorithm development.	The team has started doing some preliminary work with respect to the project. The students however are not prepared enough for the work and they need to improve a lot.	There is some evidence to show that the team has done good amount of preliminary investigation and design/ analysis/ modeling etc. They can improve further.	Strong evidence for excellent progress in the project. The team has completed the required preliminary work already and are poised to finish the phase I in an excellent manner. They have shown results to prove their progress.
			(0 – 3 Marks)	(4 – 6 Marks)	(7 - 9 Marks)	(10 Marks)

1-f	Documentation and presentation. (Individual & group assessment). [CO6]	5	The team did not document the work at all. The project journal/diary is not presented. The presentation was shallow in content and dull in appearance. The individual student has no idea on the presentation of his/her part.	Some documentation is done, but not extensive. Interaction with the guide is minimal. Presentation include some points of interest, but overall quality needs to be improved. Individual performance to be improved.	Most of the project details were documented well enough. There is scope for improvement. The presentation is satisfactory. Individual performance is good.	The project stages are extensively documented in the report. Professional documentation tools like LaTeX were used to document the progress of the project along with the project journal. The documentation structure is well-planned and can easily grow into the project report. The presentation is done professionally and with great clarity. The individual's performance is excellent.
			(0 – 1 Marks)	(2 – 3 Marks)	(4 Marks)	(5 Marks)
Total		**30**	**Phase - I Final Evaluation Marks: 30**			

EVALUATION RUBRICS for PROJECT Phase I: Report Evaluation						
Sl. No.	**Parameters**	**Marks**	**Poor**	**Fair**	**Very Good**	**Outstanding**
1-g	Report [CO6]	20	The prepared report is shallow and not as per standard format. It does not follow proper organization. Contains mostly Unacknowledged content. Lack of effort in preparation is evident.	Project report follows the standard format to some extent. However, its organization is not very good. Language needs to be improved. All references are not cited properly in the report.	Project report shows evidence of systematic documentation. Report is following the standard format and there are only a few issues. Organization of the report is good. Most of references are cited properly.	The report is exceptionally good. Neatly organized. All references cited properly. Diagrams/Figures, Tables and equations are properly numbered, and listed and clearly shown. Language is excellent and follows standard styles.
			(0 - 7 Marks)	(8 - 12 Marks)	(13 - 19 Marks)	(20 Marks)
Phase - I Project Report Marks: 20						

SEMESTER VII

PROGRAM ELECTIVE II

CST413	MACHINE LEARNING	CATEGORY	L	T	P	CREDIT	YEAR OF INTRODUCTION
		PEC	2	1	0	3	2019

Preamble: This course enables the learners to understand the advanced concepts and algorithms in machine learning. The course covers the standard and most popular supervised learning algorithms such as linear regression, logistic regression, decision trees, Bayesian learning and the Naive Bayes algorithm, basic clustering algorithms and classifier performance measures. This course helps the students to provide machine learning based solutions to real world problems.

Prerequisite: Basic understanding of probability theory, linear algebra and Python Programming

Course Outcomes: After the completion of the course the student will be able to

CO1	Illustrate Machine Learning concepts and basic parameter estimation methods. (Cognitive Knowledge Level: **Apply)**
CO2	Demonstrate supervised learning concepts (regression, linear classification). **(**Cognitive Knowledge Level: **Apply**)
CO3	Illustrate the concepts of Multilayer neural network and Support Vector Machine **(**Cognitive Knowledge Level: **Apply**)
CO4	Describe unsupervised learning concepts and dimensionality reduction techniques. (Cognitive Knowledge Level: **Apply**)
CO5	Solve real life problems using appropriate machine learning models and evaluate the performance measures (Cognitive Knowledge Level: **Apply**)

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	✓	✓	✓	✓								✓
CO2	✓	✓	✓	✓	✓							✓
CO3	✓	✓	✓	✓	✓							✓
CO4	✓	✓	✓	✓	✓							✓
CO5	✓	✓	✓	✓	✓							✓

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	**PO7**	Environment and Sustainability
PO2	Problem Analysis	**PO8**	Ethics
PO3	Design/Development of solutions	**PO9**	Individual and team work
PO4	Conduct investigations of complex problems	**PO10**	Communication
PO5	Modern tool usage	**PO11**	Project Management and Finance
PO6	The Engineer and Society	**PO12**	Life long learning

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks (%)
	Test 1 (%)	Test 2 (%)	
Remember	**30**	**30**	**30**
Understand	**30**	**30**	**30**
Apply	**40**	**40**	**40**
Analyze			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	**50**	**100**	**3**

Continuous Internal Evaluation Pattern:

Attendance	**10 marks**
Continuous Assessment Tests(Average of Internal Tests 1 & 2)	**25 marks**
Continuous Assessment Assignment	**15 marks**

Internal Examination Pattern

Each of the two internal examinations has to be conducted out of 50 marks. First series test shall be preferably conducted after completing the first half of the syllabus and the second series test shall be preferably conducted after completing remaining part of the syllabus. There will be two parts: Part A and Part B. Part A contains 5 questions (preferably, 2 questions each from the completed modules and 1 question from the partly completed module), having 3 marks for each question adding up to 15 marks for part A. Students should answer all questions from Part A. Part B contains 7 questions (preferably, 3 questions each from the completed modules and 1 question from the partly completed module), each with 7 marks. Out of the 7 questions, a student should answer any 5.

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 full questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carries 14 marks.

Syllabus

Module-1 (Overview of machine learning)

Machine learning paradigms-supervised, semi-supervised, unsupervised, reinforcement learning. Basics of parameter estimation - maximum likelihood estimation(MLE) and maximum a posteriori estimation(MAP). Introduction to Bayesian formulation.

Module-2 (Supervised Learning)

Regression - Linear regression with one variable, Linear regression with multiple variables, solution using gradient descent algorithm and matrix method, basic idea of overfitting in regression. Linear Methods for Classification- Logistic regression, Naive Bayes, Decision tree algorithm ID3.

Module-3 (Neural Networks (NN) and Support Vector Machines (SVM))

Perceptron, Neural Network - Multilayer feed forward network, Activation functions (Sigmoid, ReLU, Tanh), Backpropagation algorithm.

SVM - Introduction, Maximum Margin Classification, Mathematics behind Maximum Margin Classification, Maximum Margin linear separators, soft margin SVM classifier, non-linear SVM, Kernels for learning non-linear functions, polynomial kernel, Radial Basis Function(RBF).

Module-4 (Unsupervised Learning)

Clustering - Similarity measures, Hierarchical Agglomerative Clustering, K-means partitional clustering, Expectation maximization (EM) for soft clustering. Dimensionality reduction – Principal Component Analysis.

Module-5 (Classification Assessment)

Classification Performance measures - Precision, Recall, Accuracy, F-Measure, Receiver Operating Characteristic Curve(ROC), Area Under Curve(AUC. Bootstrapping, Cross Validation, Ensemble methods, Bias-Variance decomposition. Case Study: Develop a classifier for face detection.

Text Book

1. Ethem Alpaydin, Introduction to Machine Learning, 2nd edition, MIT Press 2010.
2. Mohammed J. Zaki and Wagner Meira, Data Mining and Analysis: Fundamental Concepts and Algorithms, Cambridge University Press, First South Asia edition, 2016.
3. Jake VanderPlas, Python Data Science Handbook, O'Reilly Media, 2016
4. Tom Mitchell, Machine Learning, McGraw-Hill, 1997.

Reference Books

1. Christopher Bishop. Neural Networks for Pattern Recognition,Oxford University Press, 1995.
2. Kevin P. Murphy. Machine Learning: A Probabilistic Perspective, MIT Press 2012.
3. Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements Of Statistical Learning, Second edition Springer 2007.
4. P. Langley, Elements of Machine Learning, Morgan Kaufmann, 1995.
5. Richert and Coelho, Building Machine Learning Systems with Python.
6. Davy Cielen, Arno DB Meysman and Mohamed Ali.Introducing Data Science: Big Data, Machine Learning, and More, Using Python Tools, Dreamtech Press 2016.

Course Level Assessment Questions

Course Outcome1 (CO1):

1. A coin is tossed 100 times and lands heads 62 times. What is the maximum likelihood estimate for θ, the probability of heads.
2. Suppose data $x_1, \ldots, x_n$ are independent and identically distributed drawn from an exponential distribution $exp(\lambda)$. Find the maximum likelihood for λ.
3. Suppose $x_1, \ldots, x_n$ are independent and identically distributed(iid) samples from a distribution with density

   $$f_X(x \mid \theta) = \begin{cases} \frac{\theta x^{\theta-1}}{3^\theta}, & 0 \le x \le 3 \\ 0, & \text{otherwise} \end{cases}$$

   Find the maximum likelihood estimate(MLE) for θ.
4. Find the maximum likelihood estimator (MLE) and maximum a posteriori (MAP) estimator for the mean of a univariate normal distribution. Assume that we have N samples, $x_1,\ldots, x_N$ independently drawn from a normal distribution with known variance σ^2 and unknown mean μ and the prior distribution for the mean is itself a normal distribution with mean v and variance β^2. What happens to the MLE and MAP estimators as the number of samples goes to infinity.

Course Outcome 2(CO2):

1. Suppose that you are asked to perform linear regression to learn the function that outputs y, given the D-dimensional input x. You are given N independent data points, and that all the D attributes are linearly independent. Assuming that D is around 100, would you prefer the closed form solution or gradient descent to estimate the regressor?
2. Suppose you have a three class problem where class label $y \in 0, 1, 2$ and each training example X has 3 binary attributes $X_1, X_2, X_3 \in 0, 1$. How many parameters (probability distribution) do you need to know to classify an example using the Naive Bayes classifier?

Course Outcome 3(CO3):

1. What are support vectors and list any three properties of the support vector classifier solution?
2. Why do you use kernels to model a projection from attributes into a feature space, instead of simply projecting the dataset directly?

3. Describe how Support Vector Machines can be extended to make use of kernels. Illustrate with reference to the Gaussian kernel $K(x, y) = e^{-z}$, where $z = (x-y)^2$.
4. Briefly explain one way in which using tanh instead of logistic activations makes optimization easier.
5. ReLU activation functions are most used in neural networks instead of the tanh activation function. Draw both activation functions and give a) an advantage of the ReLU function compared to the tanh function. b) a disadvantage of the ReLU function compared to the tanh function.

Course Outcome 4(CO4): .

1. Which similarity measure could be used to compare feature vectors of two images? Justify your answer.
2. Illustrate the strength and weakness of k-means algorithm.
3. Suppose you want to cluster the eight points shown below using **k**-means

	A_1	A_2
x_1	2	10
x_2	2	5
x_3	8	4
x_4	5	8
x_5	7	5
x_6	6	4
x_7	1	2
x_8	4	9

Assume that **k = 3** and that initially the points are assigned to clusters as follows:

$C_1 = \{x_1, x_2, x_3\}$, $C_2 = \{x_4, x_5, x_6\}$, $C_3 = \{x_7, x_8\}$. Apply the **k**-means algorithm until convergence, using the Manhattan distance.

4. Cluster the following eight points representing locations into three clusters: $A_1(2, 10)$, $A_2(2, 5)$, $A_3(8, 4)$, $A_4(5, 8)$, $A_5(7, 5)$, $A_6(6, 4)$, $A_7(1, 2)$, $A_8(4, 9)$.

   Initial cluster centers are: $A_1(2, 10)$, $A_4(5, 8)$ and $A_7(1, 2)$.

   The distance function between two points $a = (x_1, y_1)$ and $b = (x_2, y_2)$ is defined as $D(a, b) = |x_2 - x_1| + |y_2 - y_1|$

   Use **k**-Means Algorithm to find the three cluster centers after the second iteration.

Course Outcome 5(CO5):

1. What is ensemble learning? Can ensemble learning using linear classifiers learn classification of linearly non-separable sets?
2. Describe boosting. What is the relation between boosting and ensemble learning?
3. Classifier A attains 100% accuracy on the training set and 70% accuracy on the test set. Classifier B attains 70% accuracy on the training set and 75% accuracy on the test set. Which one is a better classifier. Justify your answer.
4. What are ROC space and ROC curve in machine learning? In ROC space, which points correspond to perfect prediction, always positive prediction and always negative prediction? Why?
5. Suppose there are three classifiers A,B and C. The (FPR, TPR) measures of the three classifiers are as follows – A (0, 1), B (1, 1) , C (1,0.5). Which can be considered as a perfect classifier? Justify your answer.

Model Question Paper

QP CODE:

Reg No: _____

Name: _____ **PAGES : 4**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST413

Course Name: Machine Learning

Max. Marks : 100 **Duration: 3 Hours**

PART A

Answer All Questions. Each Question Carries 3 Marks

1. Define supervised learning? Name special cases of supervised learning depending on whether the inputs/outputs are categorical, or continuous.

2. Differentiate between Maximum Likelihood estimation (MLE) and Maximum a Posteriori (MAP) estimation?

3. What is overfitting and why is it a problem?

4. Specify the basic principle of gradient descent algorithm.

5. Suppose that you have a linear support vector machine(SVM) binary classifier. Consider a point that is currently classified correctly, and is far away from the decision boundary. If you remove the point from the training set, and re-train the classifier, will the decision boundary change or stay the same? Justify your answer.

6. Mention the primary motivation for using the kernel trick in machine learning algorithms?

7. Expectation maximization (EM) is designed to find a maximum likelihood setting of the parameters of a model when some of the data is missing. Does the algorithm converge? If so, do you obtain a locally or globally optimal set of parameters?

8. Illustrate the strength and weakness of k-means algorithm.

9. Classifier A attains 100% accuracy on the training set and 70% accuracy on the test set. Classifier B attains 70% accuracy on the training set and 75% accuracy on the test set. Which one is a better classifier. Justify your answer.

10. How does bias and variance trade-off affect machine learning algorithms?

(10x3=30)

Part B

(Answer any one question from each module. Each question carries 14 Marks)

11. (a) Suppose that X is a discrete random variable with the following probability mass function: where $0 \leq \theta \leq 1$ is a parameter. The following 10 independent observations **(7)**

X	0	1	2	3
$P(X)$	$2\theta/3$	$\theta/3$	$2(1-\theta)/3$	$(1-\theta)/3$

were taken from such a distribution: (3, 0, 2, 1, 3, 2, 1, 0, 2, 1). What is the maximum likelihood estimate of θ.

(b) Suppose you have a three class problem where class label $y \in 0, 1, 2$ **(7)**

and each training example X has 3 binary attributes $X_1, X_2, X_3 \in 0, 1$.

How many parameters (probability distribution) do you need to know

to classify an example using the Naive Bayes classifier?

OR

12. (a) Consider the geometric distribution, which has p.m.f $P(X = k) = (1 - \theta)^{k-1}\theta$. Assume that n i.i.d data are drawn from that distribution. **(7)**

i. Write an expression for the log-likelihood of the data as a function of the parameter θ.

ii. Find the maximum likelihood estimate for θ?

ii. Let θ has a beta prior distribution. What is the posterior distribution of θ?

(b) Find the maximum a posteriori (MAP) estimator for the mean of a univariate normal distribution. Assume that we have N samples, $x_1,\ldots, x_N$ independently drawn from a normal distribution with known variance σ^2 and unknown mean μ and the prior distribution for the mean is itself a normal distribution with mean ν and variance β^2. **(7)**

13. (a) Consider the hypothesis for the linear regression $h_\theta(x) = \theta_0 + \theta_1 x$, and the cost function $J(\theta_0, \theta_1) = 1/2m \sum_1^m (h_\theta(x^{(i)}) - y^{(i)})^2$ where m is the number of training examples. Given the following set of training examples. **(7)**

x	y
3	2
1	2
0	1
4	3

Answer the following questions :

1) Find the value of h_θ (2) if $\theta_0 = 0$ and $\theta_1 = 1.5$

2) Find the value of J(0,1)

3) Suppose the value of $J(\theta_0, \theta_1) = 0$. What can be inferred from this.

(b) Assume we have a classification problem involving 3 classes: professors, students, and staff members. There are 750 students, 150 staff members and 100 professors. All professors have blond hair, 50 staff members have blond hair, and 250 students have blond hair. Compute the information gain of the test "hair color = blond" that returns true or false. **(3)**

(c) Explain the significance of regularization. How do Ridge differs from Lasso regularization? **(4)**

OR

14. (a) The following dataset can be used to train a classifier that determines whether a given person is likely to own a car or not. There are three features: education level (primary, secondary, or university); residence (city or country); gender (female, male). **(7)**

education	residence	gender	has car?
sec	country	female	yes
univ	country	female	yes
prim	city	male	no
univ	city	male	no
sec	city	female	no
sec	country	male	yes
prim	country	female	yes
univ	country	male	yes
sec	city	male	yes
prim	city	female	no
univ	city	female	no
prim	country	male	yes

Use ID3 Algorithm and find the best attribute at the root level of the tree

(b) Consider a linear regression problem $y = w1x + w0$, with a training set having m examples $(x_1, y_1), \ldots (x_m, y_m)$. Suppose that we wish to minimize the mean 5^{th} degree error (loss function) given by $1/m \, \Sigma_1^m (y_i - w_1 x_i - w_0)^5$. **(7)**

1. Calculate the gradient with respect to the parameter w_1.
2. Write down pseudo-code for on-line gradient descent on w_1.
3. Give one reason in favor of on-line gradient descent compared to batch-gradient descent, and one reason in favor of batch over on-line.

15. (a) Consider a support vector machine whose input space is 2-D, and the inner products are computed by means of the kernel $K(x, y) = (x.y + 1)^2 - 1$, where $x.y$ denotes the ordinary inner product. Show that the mapping to feature space that is implicitly defined by this kernel is the mapping to 5-D given by **(8)**

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \rightarrow \phi(\mathbf{x}) = \begin{bmatrix} x_1^2 \\ x_2^2 \\ \sqrt{2}\, x_1 x_2 \\ \sqrt{2}\, x_1 \\ \sqrt{2}\, x_2 \end{bmatrix}.$$

(b) Consider a neuron with four inputs, and weight of edge connecting the inputs are 1, 2, 3 and 4. Let the bias of the node is zero and inputs are 2, 3, 1, 4. If the activation function is linear f(x)=2x, compute the output of the neuron. **(3)**

(c) Compare ReLU with Sigmoid function **(3)**

OR

16. (a) State the mathematical formulation to express Soft Margin as a constraint optimization problem. **(10)**

(b) What is the basic idea of back propagation algorithm **(4)**

17. (a) Suppose that we have the following data (one variable). Use single linkage Agglomerative clustering to identify the clusters. **(8)**

Data: (2, 5, 9, 15, 16, 18, 25, 33, 33, 45).

(b) Given two objects represented by the tuples (22, 1, 42, 10) and (20, 0, 36, 8): **(6)**

(i) Compute the Euclidean distance between the two objects.

(ii) Compute the Manhattan distance between the two objects.

(iii) Compute the Minkowski distance between the two objects, using p = 3

OR

18. (a) Suppose that we have the following data: **(8)**

(2, 0), (1, 2), (2, 2), (3, 2), (2, 3), (3, 3), (2, 4), (3, 4), (4, 4), (3, 5)

Identify the cluster by applying the k-means algorithm, with k = 2. Try using initial cluster centers as far apart as possible

(b) Describe EM algorithm for Gaussian Mixtures **(8)**

19. (a) Suppose the dataset had 9700 cancer-free images from 10000 images from cancer patients. Find precision, recall and accuracy ? Is it a good classifier? Justify. **(7)**

Actual Class\Predicted class	cancer = yes	cancer = no	Total
cancer = yes	**90**	**210**	300
cancer = no	**140**	**9560**	9700
Total	230	9770	10000

(b) What is Principal Component Analysis (PCA)? Which eigen value indicates the direction of largest variance? **(7)**

OR

20. (a) Assume you have a model with a high bias and a low variance. What are the characteristics of such a model? **(6)**

(b) What are ROC space and ROC curve in machine learning? In ROC space, which points correspond to perfect prediction, always positive prediction and always negative prediction? Why? **(8)**

Teaching Plan

No	Contents	No. of Lecture Hours (37 hrs)
	Module -1 (Overview of machine learning) (7 hours)	
1.1	Supervised, semi-supervised, unsupervised learning, reinforcement learning (Text Book (TB) 1: Chapter 1)	1 hour
1.2	Maximum likelihood estimation(MLE) (TB 1: Section 4.2)	1 hour
1.3	Maximum likelihood estimation (MLE)- example (TB 1: Section 4.2)	1 hour
1.4	Maximum a posteriori estimation(MAP) (TB 4: Section 6.2)	1 hour
1.5	Maximum a posteriori estimation(MAP)-example (TB 4: Section 6.2)	1 hour
1.6	Bayesian formulation (TB 1: Section 14.1, 14.2)	1 hour
1.7	Bayesian formulation -example (TB 1: Section 14.1, 14.2)	1 hour
	Module-2 (Supervised Learning) (7 hours)	
2.1	Linear regression with one variable (TB 1: Section 2.6)	1 hour
2.2	Multiple variables, Solution using gradient descent algorithm and matrix method (No derivation required) (TB 1: Section 5.8)	1 hour
2.3	Overfitting in regression, Lasso and Ridge regularization	1 hour
2.4	Logistic regression	1 hour
2.5	Naive Bayes (TB 2: Section 18.2)	1 hour
2.6	Decision trees (TB 2: Chapter 19)	1 hour
2.7	Decision trees- ID3 algorithm (TB 2: Chapter 19)	1 hour
	Module-3 (Neural Networks and Support Vector Machines) (9 hours)	
3.1	Perceptron, Perceptron Learning	1 hour
3.2	Multilayer Feed forward Network, Activation Functions (Sigmoid, ReLU, Tanh)	1 hour
3.3	Back Propagation Algorithm	1 hour
3.4	Illustrative Example for Back Propagation	1 hour
3.5	Introduction, Maximum Margin Hyperplane,	1 hour
3.6	Mathematics behind Maximum Margin Classification	1 hour
3.7	Formulation of maximum margin hyperplane and solution	1 hour

3.8	Soft margin SVM, Solution of Soft margin SVM	1 hour
3.9	Non-linear SVM , Kernels for learning non-linear functions, Examples - Linear, RBF, Polynomial	1 hour
Module-4 (Unsupervised Learning) (7 hours)		
4.1	Similarity measures- Minkowski distance measures(Manhattan, Euclidean), Cosine Similarity	1 hour
4.2	Clustering - Hierarchical Clustering (TB 2: Chapter 14)	1 hour
4.3	K-means partitional clustering (TB 2: Chapter 13)	1 hour
4.4	Expectation maximization (EM) for soft clustering (TB 2: Chapter 13)	1 hour
4.5	Expectation maximization (EM) for soft clustering (TB 2: Chapter 13)	1 hour
4.6	Dimensionality reduction – Principal Component Analysis (TB 1: Section 6.3)	1 hour
4.7	Dimensionality reduction – Principal Component Analysis (TB 1: Section 6.3)	1 hour
Module-5 (Classification Assessment) (7 hours)		
5.1	Performance measures - Precision, Recall, Accuracy, F-Measure, ROC, AUC. (TB 2: Chapter 22.1)	1 hour
5.2	Boot strapping, Cross validation	1 hour
5.3	Ensemble methods- bagging, boosting	1 hour
5.4	Bias-Variance decomposition (TB 2: Chapter 22.3)	1 hour
5.5	Bias-Variance decomposition (TB 2: Chapter 22.3)	1 hour
5.6	Face detection (TB 3: Chapter 5 Section Application: A Face Detection Pipeline)	1 hour
5.7	Face detection (TB 3: Chapter 5 Section Application: A Face Detection Pipeline)	1 hour

CST423	CLOUD COMPUTING	CATEGORY	L	T	P	CREDIT	YEAR OF INTRODUCTION
		PEC	2	1	0	3	2019

Preamble: This course helps the learners to understand cloud computing concepts. This course includes basic understanding of virtualization, fundamentals of cloud security, cloud computing based programming techniques and different industry popular cloud computing platforms. This course enables the student to suggest cloud based solutions to real world problems.

Prerequisite: Basic understanding of computer networks and operating systems.

Course Outcomes: After the completion of the course the student will be able to

CO1	Explain the various cloud computing models and services. **(Cognitive Knowledge Level: Understand)**
CO2	Demonstrate the significance of implementing virtualization techniques. **(Cognitive Knowledge Level: Understand)**
CO3	Explain different cloud enabling technologies and compare private cloud platforms **(Cognitive Knowledge Level: Understand)**
CO4	Apply appropriate cloud programming methods to solve big data problems. (Cognitive Knowledge Level: **Apply**)
CO5	Describe the need for security mechanisms in cloud (**Cognitive Knowledge Level: Understand**)
CO6	Compare the different popular cloud computing platforms **(Cognitive Knowledge Level: Understand)**

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	✓											✓
CO2	✓	✓	✓									✓
CO3	✓											✓
CO4	✓	✓	✓	✓	✓							✓
CO5	✓	✓										✓
CO6	✓				✓							✓

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	**PO7**	Environment and Sustainability
PO2	Problem Analysis	**PO8**	Ethics
PO3	Design/Development of solutions	**PO9**	Individual and team work
PO4	Conduct investigations of complex problems	**PO10**	Communication
PO5	Modern tool usage	**PO11**	Project Management and Finance
PO6	The Engineer and Society	**PO12**	Life long learning

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks
	Test1 (Percentage)	Test2 (Percentage)	
Remember	30	30	30
Understand	40	40	40
Apply	30	30	30
Analyze			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	**50**	**100**	**3 hours**

Continuous Internal Evaluation Pattern:

Attendance : 10 marks

Continuous Assessment Tests : 25 marks

Continuous Assessment Assignment : 15 marks

Internal Examination Pattern:

Each of the two internal examinations has to be conducted out of 50 marks.

First Internal Examination shall be preferably conducted after completing the first half of the syllabus and the Second Internal Examination shall be preferably conducted after completing the remaining part of the syllabus.

There will be two parts: Part A and Part B. Part A contains 5 questions (preferably, 2 questions each from the completed modules and 1 question from the partly covered module), having 3 marks for each question adding up to 15 marks for part A. Students should answer all questions from Part A. Part B contains 7 questions (preferably, 3 questions each from the completed modules and 1 question from the partly covered module), each with 7 marks. Out of the 7 questions in Part B, a student should answer any 5.

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which a student should answer any one. Each question can have a maximum of 2 subdivisions and carries 14 marks.

Syllabus

Module 1: Fundamental Cloud Computing (7 Hours)

Traditional computing- Limitations. Overview of Computing Paradigms-Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Cloud Computing. NIST reference Model-Basic terminology and concepts. Cloud characteristics, benefits and challenges, Roles and Boundaries. Cloud delivery (service) models-Infrastructure-as-a-Service (IaaS), Platform-as-a-Service(PaaS),Software-as-a-Service (SaaS), XaaS (Anything-as-a-service)-Cloud deployment models- Public cloud, Community cloud, Private cloud, Hybrid cloud.

Module 2: Virtualization (7 Hours)

Introduction to virtualization-Virtualizing physical computing resources, Virtual Machines (Machine virtualization), non-virtualized v/s virtualized machine environments. Types of VMs-process VM v/s system VM, Emulation, interpretation and binary translation. Hardware-level virtualization- Hypervisors/VMM. Types of Hypervisors. Full Virtualization, Para-Virtualization, Hardware-assisted virtualization, OS level virtualization. Basics of Network Virtualization, Storage Virtualization and Desktop Virtualization, Pros and cons of virtualization. Case Study- Xen: Para-virtualization, VMware: full virtualization.

Module 3: Cloud-Enabling Technologies, Private cloud platforms and programming (7 Hours)

Broadband networks and internet architecture- Internet Service Providers (ISPs), Data center technology, Web technology, Multitenant technology, Service technology. Resource provisioning techniques-static and dynamic provisioning.

Open-source software platforms for private cloud-OpenStack, CloudStack, Basics of Eucalyptus, Open Nebula, Nimbus.

Cloud Programming- Parallel Computing and Programming Paradigms. Map Reduce – Hadoop Library from Apache, HDFS, Pig Latin High Level Languages, Apache Spark.

Module 4: Fundamental Cloud Security (7 Hours)

Basic terms and concepts in security- Threat agents, Cloud security threats/risks, Trust. Operating system security-Virtual machine security- Security of virtualization- Security Risks Posed by Shared Images, Security Risks Posed by Management OS. Infrastructure security-Network Level Security, Host Level Security, Application level security, Security of the Physical Systems. Identity & Access Management- Access Control.

Module 5: Popular Cloud Platforms (9 Hours)

Amazon Web Services(AWS):- AWS ecosystem- Computing services, Amazon machine images, Elastic Compute Cloud (EC2), Advanced compute services. Storage services-Simple Storage System (Amazon S3), Elastic Block Store (Amazon EBS), Database Services, Amazon CDN Services and Communication services.

Google Cloud Platform:- IaaS Offerings: Compute Engine (GCE), Cloud Storage, PaaS Offerings: Google App Engine (GAE), Storage services, Application services, Compute services, Database Services, SaaS Offerings: Gmail, Docs, Google Drive.

Microsoft Azure: Azure Platform Architecture, Hyper-V, Azure Virtual Machine, Compute services, Storage services.

Text Books

1. Thomas, E., Zaigham M., Ricardo P "Cloud Computing Concepts, Technology & Architecture.", (2013 Edition). Prentice Hall.
2. Buyya, R., Vecchiola, C., & Selvi, S. T. “Mastering cloud computing: foundations and applications programming”, (2017 Edition), Morgan Kaufmann.
3. Bhowmik, S., “Cloud computing”, (2017 Edition). Cambridge University Press.

References

1. Marinescu, D. C., "Cloud computing: theory and practice.", (2017 Edition). Morgan Kaufmann.
2. Buyya, R., Broberg, J., & Goscinski, A. M., "Cloud computing: Principles and paradigms" (2011 Edition). John Wiley & Sons.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. "*A hybrid cloud is a combination of two or more other cloud deployment models*". Justify the statement with an example.
2. What are the main characteristics of a Platform-as-a-Service solution?
3. How does cloud computing help to reduce the time to market for applications and to cut down capital expenses?
4. Differentiate public and private clouds in terms of flexibility.

Course Outcome 2 (CO2):

1. Define virtualization. What is the role of VMM in virtualization?
2. Explain various implementation levels of Virtualization.
3. State the differences between a traditional computer and a virtual machine.

Course Outcome 3 (CO3):

1. Differentiate between on-premise and cloud-based internetworking.
2. What are the benefits of Data Center Technologies?
3. What are the characteristics of Multi-tenant technology?
4. How can virtualization be implemented at the hardware level?

Course Outcome 4 (CO4):

1. Write a Hadoop MapReduce program that counts the number of occurrences of each character in a file.
2. Write a Hadoop MapReduce program to find the maximum temperature in the weather dataset.

Course Outcome 5 (CO5):

1. Why is it harder to establish security in the cloud?
2. Explain in detail about the security issues one should discuss with a cloud-computing vendor.
3. List and Explain major cloud security challenges.

Course Outcome 6 (CO6):

1. Explain the cloud based databases.
2. With a neat diagram, write about Google App Engine for PaaS applications.
3. Differentiate between amazon SimpleDB and Amazon RDS.
4. *"Storage services in the cloud are offered in two different forms as IaaS and as SaaS".* Explain.

Model Question Paper

QP Code: **Total Pages : 3**

Reg No:_____ **Name :**_____

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST423
Course Name: Cloud Computing

Duration: 3 Hrs **Max. Marks :100**

PART A
Answer all Questions. Each question carries 3 Marks (10 x 3 = 30 Marks)

1. Is the IT outsourcing model of traditional computing similar to cloud computing? Justify.

2. Why is grid computing considered as the predecessor of cloud computing? Explain.

3. What is virtualization and what are its benefits?

4. Explain why a hypervisor is also called a virtual machine monitor?

5. Differentiate between multi-tenancy and virtualization.

6. "*The field of service technology is a keystone foundation of cloud computing*". Explain.

7. Discuss any two identity management techniques used in cloud computing.

8. Differentiate between mandatory access control (MAC) and discretionary Access Control (DAC).

9. Differentiate between Amazon S3 and Amazon EBS.

10. Explain the database service offered by google cloud.

(10 x3 =30)

PART B

Answer any one Question from each Module. Each question carries 14 Marks

11. **(a)** Discuss the cloud computing reference model. **(8)**

(b) Which are the basic components of an IaaS-based solution for cloud computing? Also provide some examples of IaaS implementations. **(6)**

OR

12. **(a)** List down the characteristics and challenges of cloud computing. **(6)**

(b) Classify the various types of clouds. **(8)**

13. **(a)** List and discuss various types of virtualization. **(8)**

(b) Differentiate between full virtualization and paravirtualization. **(6)**

OR

14. **(a)** What is Xen? Discuss its elements for virtualization. **(8)**

(b) Explain the design requirements for Virtual Machine Monitor (VMM). **(6)**

15. **(a)** Explain the broadband networks and internet architecture. **(8)**

(b) List and explain the technologies and components of data centers. **(6)**

OR

16. **(a)** What are the major functions of the MapReduce framework? Explain the logical data flow of MapReduce function using a suitable example . **(8)**

(b) Write a Hadoop MapReduce program that counts the number of occurrences of each word in a file. **(6)**

17. **(a)** Explain common threats and vulnerabilities in cloud-based environments with suitable examples. **(8)**

(b) Discuss the security risks posed by shared images with suitable examples. **(6)**

OR

18. **(a)** Explain the operating system security in cloud computing. **(8)**

(b) What do you mean by threat agents?. Explain different types of threat agents. **(6)**

19. **(a)** Describe Amazon EC2 and its basic features. **(8)**

(b) Illustrate the architecture of Amazon S3. **(6)**

OR

20. **(a)** Describe the core components of Google AppEngine. **(8)**

(b) Explain the architecture of Windows Azure. **(6)**

Teaching Plan

No	Contents	No. of Lecture Hours (37 hrs)
	Module 1 (Fundamental Cloud Computing) (6 hours)	
1.1	Traditional computing: Limitations.	1
1.2	Overview of Computing Paradigms: Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Cloud Computing.	1
1.3	NIST reference Model, Basic terminology and concepts.	1
1.4	Cloud characteristics and benefits, challenges. Roles and Boundaries.	1
1.5	Cloud delivery (service) models: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), Software-as-a-Service (SaaS), XaaS (Anything-as-a-service).	1
1.6	Cloud deployment models: Public cloud, Community cloud, Private cloud, Hybrid cloud.	1

Module 2(Virtualization)(7 Hours)		
2.1	Introduction to virtualization, Virtualizing physical computing resources Virtual Machines (Machine virtualization):- non-virtualized v/s virtualized machine environments.	1
2.2	Types of VMs: process VM v/s system VM, Emulation, interpretation and binary translation.	1
2.3	Hardware-level virtualization: Hypervisors/VMM, Types of Hypervisors.	1
2.4	Full Virtualization, Para-Virtualization, Hardware-assisted virtualization, OS level virtualization.	1
2.5	Basics of Network Virtualization, Storage Virtualization and Desktop Virtualization, Pros and cons of virtualization.	1
2.6	Case Study: Xen: Para-virtualization.	1
2.7	Case Study: VMware: full virtualization.	1
Module 3 (Cloud-Enabling Technologies, Private cloud platforms and programming) (9 Hours)		
3.1	Broadband networks and internet architecture: Internet Service Providers (ISPs), Data center technology, Web technology, Multitenant technology, Service technology.	1
3.2	Resource provisioning techniques: static and dynamic provisioning.	1
3.3	Open-source software platforms for private cloud: OpenStack, CloudStack.	1
3.4	Basics of Eucalyptus, Open-Nebula, Nimbus.	1
3.5	Cloud Programming: Parallel Computing and Programming Paradigms.	1
3.6	Map Reduce.	1
3.7	Hadoop Library from Apache, HDFS.	1
3.8	Pig Latin High Level Languages	1
3.9	Apache Spark.	1

Module 4 (Fundamental Cloud Security) (7 Hours)		
4.1	Basic terms and concepts in security, Threat agents.	1
4.2	Cloud security threats/risks, Trust.	1
4.3	Operating system security, Virtual machine security.	1
4.4	Security of virtualization.	1
4.5	Security Risks posed by Shared Images, Security Risks posed by Management OS.	1
4.6	Infrastructure security: - Network Level Security, Host Level Security, Application level security, Security of the Physical Systems.	1
4.7	Identity & Access Management, Access Control.	1
Module 5 (Popular Cloud Platforms) (8 Hours)		
5.1	Amazon Web Services(AWS):- AWS ecosystem, Computing services: Amazon machine images, Elastic Compute Cloud (EC2).	1
5.2	Advanced computing services, Storage services: Simple Storage System (Amazon S3), Elastic Block Store (Amazon EBS).	1
5.3	Database Services, Amazon CDN Services and Communication services.	1
5.4	Google Cloud Platform:- IaaS Offerings: Compute Engine (GCE), Cloud Storage.	1
5.5	PaaS Offerings: Google App Engine (GAE), Storage services, Application services, Compute services.	1
5.6	Database Services, SaaS Offerings: Gmail, Docs, Google Drive.	1
5.7	Microsoft Azure: Azure Platform Architecture, Hyper-V, Azure Virtual Machine.	1
5.8	Azure Compute services, Storage services.	1

CST433	SECURITY IN COMPUTING	CATEGORY	L	T	P	CREDIT	YEAR OF INTRODUCTION
		PEC	2	1	0	3	2019

Preamble: This course helps the learners to explore various algorithms to offer confidentiality, integrity, authentication &non-repudiation services and different attacks on system security with their countermeasures. It covers classical encryption techniques, symmetric and public key crypto-system, key distribution techniques, authentication functions, intruders, malicious software, and DDoS attacks. The concepts covered in this course enable the learners in effective use of cryptographic algorithms and appropriate countermeasures for securing real life applications.

Prerequisite: A fundamental knowledge in mathematical foundations of security.

Course Outcomes: After the completion of the course, the student will be able to

CO1	Identify the security services provided against different types of security attacks. **(Cognitive Knowledge Level: Understand)**
CO2	Illustrate classical encryption techniques for information hiding. **(Cognitive Knowledge Level: Apply)**
CO3	Illustrate symmetric/asymmetric key cryptosystems for secure communication. **(Cognitive Knowledge Level: Apply)**
CO4	Explain message integrity and authentication methods in a secure communication scenario. **(Cognitive Knowledge Level: Understand)**
CO5	Interpret public/secret key distribution techniques for secure communication. **(Cognitive Knowledge Level: Understand)**
CO6	Identify the effects of intruders, malicious software and distributed denial of service attacks on system security. **(Cognitive Knowledge Level: Understand)**

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	✓	✓	✓									✓
CO2	✓	✓	✓									✓
CO3	✓	✓	✓		✓							✓
CO4	✓	✓	✓			✓						✓
CO5	✓	✓	✓									✓
CO6	✓	✓	✓			✓		✓				✓

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	**PO7**	Environment and Sustainability
PO2	Problem Analysis	**PO8**	Ethics
PO3	Design/Development of solutions	**PO9**	Individual and team work
PO4	Conduct investigations of complex problems	**PO10**	Communication
PO5	Modern tool usage	**PO11**	Project Management and Finance
PO6	The Engineer and Society	**PO12**	Life long learning

Assessment Pattern

Bloom's Category	Test 1 (%)	Test 2 (%)	End Semester Examination (%)
Remember	30	30	30
Understand	40	40	40

Apply	30	30	30
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3

Continuous Internal Evaluation Pattern:

Attendance : **10 marks**

Continuous Assessment Test : **25 marks**

Continuous Assessment Assignment : **15 marks**

Internal Examination Pattern:

Each of the two internal examinations has to be conducted out of 50 marks. First series test shall be preferably conducted after completing the first half of the syllabus and the second series test shall be preferably conducted after completing remaining part of the syllabus. There will be two parts: Part A and Part B. Part A contains 5 questions (preferably, 2 questions each from the completed modules and 1 question from the partly completed module), having 3 marks for each question adding up to 15 marks for part A. Students should answer all questions from Part A. Part B contains 7 questions (preferably, 3 questions each from the completed modules and 1 question from the partly completed module), each with 7 marks. Out of the 7 questions, a student should answer any 5.

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which a student should answer any one. Each question can have maximum 2 sub-divisions and carries 14 marks.

Syllabus

Module-1 (Basics of Security and Traditional Cryptosystems)
OSI security architecture – Security attacks, Services, Mechanisms. Cryptography vs Cryptanalysis. Classical encryption techniques – Symmetric cipher model. Substitution ciphers – Monoalphabetic vs Polyalphabetic ciphers, Caesar cipher, Affine cipher, Playfair cipher, Vigenere cipher, Hill cipher. Transposition ciphers – Keyless, Keyed, Double transposition.

Module-2 (Modern Symmetric Key Cryptosystems)
Symmetric key ciphers – Block vs Stream ciphers, Block cipher components, Product ciphers, Feistel and Non-Feistel ciphers. Data Encryption Standard (DES) – Structure, Key generation, Design criteria, Weaknesses, Double DES, Triple DES. Advanced Encryption Standard (AES) – Structure, Key expansion. Block cipher modes of operation – Electronic Codebook Mode (ECB), Cipher Block Chaining Mode (CBC), Cipher Feedback Mode (CFB), Output Feedback Mode (OFB), Counter Mode (CTR). Stream ciphers – Structure, RC4.

Module-3 (Public Key Cryptosystems)
Introduction to public key cryptosystems – Principles, Applications, Requirements, Conventional vs Public key cryptosystems. RSA cryptosystem – Algorithm, Security, Attacks. ElGamal cryptosystem – Algorithm. Diffie-Hellman key exchange – Algorithm, Man-in-the-middle attack. Elliptic Curve Cryptography (ECC) – ElGamal ECC, Key exchange using ECC.

Module-4 (Message Integrity and Authentication)
Hash functions – Security requirements, Secure Hash Algorithm (SHA-512). Message Authentication Code (MAC) – Requirements, Uses, Hash-based MAC (HMAC), Cipher-based MAC (CMAC). Digital signatures – Attacks, Forgeries, Requirements, Direct vs Arbitrated digital signatures, RSA digital signature, ElGamal digital signature, Digital Signature Standard (DSS).

Module-5 (Key Distribution and System Security)
Key management – Distribution of secret keys using symmetric and asymmetric encryption, Distribution of public keys. System security – Intruders, Intrusion detection techniques, Password management. Malicious software – Viruses, Related threats, Countermeasures. Distributed Denial of Service (DDoS) attacks – Types, Countermeasures.

Text Books
1. William Stallings, Cryptography and Network Security Principles and Practice, 4/e, Pearson Ed.
2. Behrouz A Forouzan, Cryptography and Network Security, 3/e, Tata McGraw-Hill.

References

1. Charles P Pfleeger, Shari Lawrence Pfleeger, Jonathan Margulies, Security in Computing, 5/e, Prentice Hall.
2. G.A. Jones & J.M. Jones, Elementary Number Theory, Springer UTM, 2007.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Define the type of security attack in the following case: A student breaks into a teacher's office to obtain a copy of the next day's exam question paper.
2. Which security mechanism is provided in the following case: A bank requires the customer's signature for a withdrawal.

Course Outcome 2 (CO2):

1. Alice wishes to send the message "COME BACK EARLY" to Bob, using Playfair cipher. The key to be used is "SAFFRON". Show the process of encryption.
2. Using Affine cipher, encrypt "HOT" and decrypt "JDG". Key is (7, 3).
3. Implement the Vigenere cipher method in a suitable programming language. (Assignment)

Course Outcome 3 (CO3):

1. If the DES key with parity bit is 0123 ABCD 2562 1456, find the first round key.
2. In RSA, given p=19, q=23, public key(e)=3, find n, $\phi(n)$ and private key(d).
3. Implement any two symmetric/asymmetric encryption techniques in a suitable programming language. (Assignment)

Course Outcome 4 (CO4):

1. Describe the steps involved in generating a Hash-based MAC.
2. Using ElGamal scheme, generate the signatures for the message M=400 with p=881, d=700 and r=17.
3. A company wishes to implement a secure authentication mechanism for communication. As a system security admin suggest any two ways of implementing such a mechanism. (Assignment)

Course Outcome 5 (CO5):

1. List any two ways in which secret keys can be distributed to two communicating parties.
2. Explain the significance of a public-key authority in the distribution of public keys.

Course Outcome 6 (CO6):

1. What are false positives and negatives in the context of Intrusion Detection Systems? How can we reduce these two?
2. Distinguish between a direct DDoS attack and a reflector DDoS attack.
3. Bob works as a network administrator in ABC & Co. On a day of his absence, he shared his admin password with one of his colleagues, John, to manage a network issue. Later John started misusing this privilege by launching DoS attacks in the network. Describe the ethical issues in this scenario and how can this be avoided? (Assignment)

Model Question Paper

QP CODE: **PAGES: ___**

Reg No:_____
Name:_____

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST433
Course Name: SECURITY IN COMPUTING

Max Marks: 100 **Duration: 3 Hours**

PART A
(Answer All Questions. Each question carries 3 marks)

1. Differentiate between passive attack and active attack.

2. Use an Affine cipher to encrypt the message “SECURITY” with the key pair(7,2) in modulus 26.

3. Compare stream cipher and Block cipher with example.

4. Differentiate between diffusion and confusion.

5. Define the elliptic curve logarithm problem.

6. Consider an ElGamal scheme with a common prime q = 71 and a primitive root α = 7. If B has a public key $Y_B = 3$ and A chose the random number k = 2, what is the ciphertext of the message M = 30?

7. Give the requirements of MAC function.

8. Specify the different types of forgery in digital signature.

9. List three different classes of intruders.

10. Mention the phases of operation of a virus. **(10x3=30)**

Part B

(Answer any one question from each module. Each question carries 14 Marks)

11. (a) Illustrate the two approaches to attack a conventional encryption scheme. **(4)**

(b) A Hill cipher is setup with the key matrix $\begin{bmatrix} 9 & 4 \\ 5 & 7 \end{bmatrix}$. **(10)**

Encrypt the text "COMPUTER". Show the calculations for the corresponding decryption of the ciphertext to recover the original text back.

OR

12. (a) Encrypt the text "this is an exercise and complete it" using transposition cipher with the key (3,2,1,4,5). Show decryption of the ciphertext to recover the original text back. **(6)**

(b) Encrypt the message "the house is being sold tonight"using the following ciphers. Ignore the space between words. **(8)**

i) Vigenere cipher with key = "largest".

ii) Autokey system of Vigenere cipher with key ="largest".

13. (a) How is round key generated in DES? **(4)**

(b) Illustrate AES encryption in detail. **(10)**

OR

14. (a) Explain the construction of S-box in AES. **(5)**

(b) Summarize the primitive operations in RC4 algorithm. **(9)**

15. (a) Compare the Cipher Block Chaining Mode (CBC) and Cipher Feedback Mode (CFB) of block ciphers. **(6)**

(b) Explain RSA cryptosystem. In an RSA cryptosystem a participant A uses two prime numbers p=13 and q=17 to generate public key and private key. The public key of A is 35. Find the private key of A. **(8)**

OR

16. (a) Illustrate ElGamal cryptosystem. **(6)**

(b) Consider a Diffie–Hellman scheme with a common prime q=11 and a primitive root α=2. **(8)**

i) Show that 2 is a primitive root of 11.

ii) If User A has public key Y_A= 9, what is A's private key X_A?

iii) If User A has public key Y_B= 3, what is the shared secret key K, shared with A?

17. (a) Describe different types of arbitrated digital signature techniques. **(6)**

(b) Explain Cipher – Based Message Authentication Code. **(8)**

OR

18. (a) Explain the attacks on digital signature. **(5)**

(b) Describe the working of SHA-512 with diagrams. **(9)**

19. (a) Explain four techniques used to avoid guessable passwords. **(6)**

(b) Describe the different techniques for public key distribution. **(8)**

OR

20. (a) Explain different types of Simple DDoS attack and its countermeasures. **(6)**

(b) Differentiate between statistical anomaly detection and rule-based intrusion detection. **(8)**

Teaching Plan

No	Contents	No.of Lecture Hours (35Hrs)
Module-1 (Basics of Security and Traditional Cryptosystems) (6 hrs)		
1.1	OSI security architecture – Security attacks, Services, Mechanisms	1
1.2	Cryptography vs Cryptanalysis. Classical encryption techniques – Symmetric cipher model	1
1.3	Substitution ciphers – Monoalphabetic vs Polyalphabetic ciphers, Caesar cipher, Affine cipher	1
1.4	Playfair cipher, Vigenere cipher	1
1.5	Hill cipher	1
1.6	Transposition ciphers – Keyless, Keyed, Double transposition	1
Module-2 (Modern Symmetric Key Cryptosystems) (9hrs)		
2.1	Symmetric key ciphers – Block vs Stream ciphers, Block cipher components, Product ciphers, Feistel and Non-Feistel ciphers	1
2.2	Data Encryption Standard (DES) – Structure, Key generation	1
2.3	Design criteria, Weaknesses	1
2.4	Double DES, Triple DES	1
2.5	Advanced Encryption Standard (AES) – Overall Structure	1
2.6	Stages of encryption/decryption	1
2.7	Key expansion	1
2.8	Block cipher modes of operation – Electronic Codebook Mode (ECB), Cipher Block Chaining Mode (CBC), Cipher Feedback Mode (CFB), Output Feedback Mode (OFB), Counter Mode (CTR).	1
2.9	Stream ciphers – Structure, RC4	1
Module-3 (Public Key Cryptosystems)(7hrs)		
3.1	Public key cryptosystems – Principles, Applications, Requirements, Conventional vs Public key cryptosystems	1

3.2	RSA cryptosystem – Algorithm	1
3.3	RSA Security, Attacks	1
3.4	ElGamal cryptosystem – Algorithm	1
3.5	Diffie-Hellman key exchange – Algorithm, Man-in-the-middle attack	1
3.6	Elliptic Curve Cryptography (ECC) – ElGamal ECC	1
3.7	Key exchange using ECC	1
Module-4 (Message Integrity and Authentication) (6 hrs)		
4.1	Hash functions – Security requirements, Secure Hash Algorithm (SHA-512)	1
4.2	Message Authentication Code (MAC) – Requirements, Uses	1
4.3	Hash-based MAC (HMAC), Cipher-based MAC (CMAC)	1
4.4	Digital signatures – Attacks, Forgeries, Requirements, Direct Vs Arbitrated digital signatures	1
4.5	RSA digital signature, ElGamal digital signature	1
4.6	Digital Signature Standard (DSS)	1
Module-5 (Key Distribution and System Security) (7hrs)		
5.1	Key management – Distribution of secret keys using symmetric and asymmetric encryption	1
5.2	Distribution of public keys	1
5.3	System security – Intruders, Intrusion detection techniques	1
5.4	Password management	1
5.5	Malicious software – Viruses, Related threats	1
5.6	Virus countermeasures	1
5.7	Distributed Denial of Service (DDoS) attacks – Types, Countermeasures	1

CST443	MODEL BASED SOFTWARE DEVELOPMENT	CATEGORY	L	T	P	CREDIT	YEAR OF INTRODUCTION
		PEC	2	1	0	3	2019

Preamble: The objective of the course is to familiarize learners about the concepts and advantages of using model based software development. This course covers the methodologies in developing the model of a software, perform analysis on the model and automatic generation of code from the model. The OSATE framework and its plugins using the Architecture Analysis and Design Language(AADL) language is used in the course to demonstrate the end-to-end concept of MBSD which helps the learners to get a hands on experience.

Prerequisite: Software Engineering

Course Outcomes: After the completion of the course the student will be able to

CO1	Explain the relevance of model based software development in the software development process. (**Cognitive Knowledge level: Understand**)
CO2	Explain Model Driven Architecture with Computation Independent Model (CIM), Platform Independent Model(PIM), Platform Specific Model (PSM). (**Cognitive Knowledge level: Apply**)
CO3	Illustrate software modeling with Architecture Analysis and Design Language (AADL). (**Cognitive Knowledge level: Apply**)
CO4	Explain error annex using error modelling concepts and illustrate error modelling in AADL. (**Cognitive Knowledge level: Understand**)
CO5	Illustrate the process of code generation from an AADL model. (**Cognitive Knowledge level: Understand**)

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	✓	✓	✓									✓
CO2	✓	✓	✓	✓								✓
CO3	✓	✓	✓	✓								✓
CO4	✓	✓	✓									✓
CO5	✓	✓	✓									✓

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	**PO7**	Environment and Sustainability
PO2	Problem Analysis	**PO8**	Ethics
PO3	Design/Development of solutions	**PO9**	Individual and team work
PO4	Conduct investigations of complex problems	**PO10**	Communication
PO5	Modern tool usage	**PO11**	Project Management and Finance
PO6	The Engineer and Society	**PO12**	Life long learning

Assessment Pattern

Bloom's Category	Test 1 (Marks in percentage)	Test 2 (Marks in percentage)	End Semester Examination Marks
Remember	30	30	30
Understand	50	50	50
Apply	20	20	20
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test : 25 marks
Continuous Assessment Assignment : 15 marks

Internal Examination Pattern:

Each of the two internal examinations has to be conducted out of 50 marks. The first series test shall be preferably conducted after completing the first half of the syllabus and the second series test shall be preferably conducted after completing the remaining part of the syllabus. There will be two parts: Part A and Part B. Part A contains 5 questions (preferably, 2 questions each from the completed modules and 1 question from the partly completed module), having 3 marks for each

question adding up to 15 marks for part A. Students should answer all questions from Part A. Part B contains 7 questions (preferably, 3 questions each from the completed modules and 1 question from the partly completed module), each with 7 marks. Out of the 7 questions, a student should answer any 5.

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which a student should answer any one. Each question can have a maximum of 2 sub-divisions and carries 14 marks.

Syllabus

Module - 1 (Introduction to Model Based Software Development)

Software faults, Introduction to Model checking, Introduction to Automated Testing, Model Based Software Development (MBSD) – Need, MBSD Approach, Learning MBSD from the perspective of Architecture Analysis and Design Language (AADL).

Module - 2 (More on MBSD)

MBSD based software development – Requirements, Analysis, Design and Implementation. Model-Driven Architecture - Definitions and Assumptions, Overview of MBSD methodology, The modeling levels-Computation Independent Model (CIM), Platform Independent Model (PIM), Platform Specific Model (PSM). Introduction to AADL, Basic Comparison of AADL with other modeling languages - Comparison with UML.

Module -3 (Modeling using AADL)

Modeling: Developing a Simple Model - Define the components - Explain with example (powerboat autopilot system), Develop a top-level model - Use example Powerboat Autopilot (PBA) system.

AADL: Components - Software, Hardware, Composite, Runtime semantics, Language syntax, AADL declarations, AADL classifiers, AADL system models and specifications

Case Study: Powerboat Autopilot System.

Module - 4 (Model Analysis)

Safety Analysis -Fault tree analysis, Minimal cutsets. Error Modeling in AADL-Error Model Libraries and Subclause Annotations, Error Types and Common Type Ontology, Error Sources and Their Impact, Component Error Behavior, Compositional Abstraction of Error Behavior, Use of Properties in Architecture Fault Models, Error modeling example.

Module - 5 (Code Generation)

Need for code generation, Categorization, Code Generation Techniques, Code Generation in AADL Model – Ocarina.

Text Books

1. Marco, Brambilla, Jordi Cabot, Manuel Wimmer, Model-Driven Software Engineering in Practice, 2/e, Synthesis Lectures on Software Engineering, 2017.
2. Christel Baier and Joost-Pieter Katoen, Principles of model checking, The MIT Press.
3. Thomas Stahl and Markus Volter, Model-Driven Software Development, Wiley, 2006.
4. David P. Gluch, Peter H. Feiler, Model-Based Engineering with AADL: An Introduction to the SAE Architecture Analysis & Design Language, Adison-Wesley, 2015.

References:

1. Automated software testing : http://www2.latech.edu
2. Peter H. Feiler, David P. Gluch, John J. Hudak.The Architecture Analysis & Design Language(AADL): An Introduction.
3. de Niz, Dionisio, Diagrams and Languages for Model-Based Software Engineering of EmbeddedSystems: UML and AADL
4. FAA System Safety Handbook, Chapter 8: Safety Analysis/Hazard Analysis Tasks
5. Enno Ruijters, Marielle Stoelinga, Fault tree analysis: A survey of the state-of-the-art in modeling, analysis and tools.
6. Larson, Brian &Hatcliff, John & Fowler, Kim &Delange, Julien. (2013). Illustrating the AADL error modeling annex (v.2) using a simple safety-critical medical device. ACM SIGAda Ada Letters. 33. 65-84. 10.1145/2527269.2527271.
7. Delange, Julien&Feiler, Peter &Hudak, John &Gluch, Dave. (2016). Architecture Fault Modeling and Analysis with the Error Model Annex, Version 2. 10.13140/RG.2.1.4224.7927.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Justify the need of model based software development?
2. Explain the advantages of model based software development?

Course Outcome 2 (CO2):

1. Explain infrastructure of model driven architecture.
2. Describe about MDA modeling levels.

Course Outcome 3 (CO3):

1. Illustrate the basic components of an AADL Model.

2. Assume we have a system to regulate the fuel valve of a boiler by monitoring the steam flow and steam pressure. Identify the basic components of this system and design its AADL model.

Course Outcome 4 (CO4):

1. Suppose we have an isolette system which ensures the temperature is within a specified temperature range with following components:

   i) temperature sensor - detects air temperature.

   iii) heat source - supply hot air to maintain temperature.

   iv) operator interface - specify target temperature range(lower desired temperature, upper desired temperature.)

   iv) thermostat - takes as input an air temperature value from a temperature sensor and controls a heat source to produce an air temperature within a target range.

   Model the error flows, error propagations, component error behaviour and error properties for the value error in the isolette system.

Course Outcome 5 (CO5):

1. Illustrate code generation from an AADL model.

Model Question Paper

QP CODE:

Reg No: _____

Name: _____ **PAGES : 4**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST443

Course Name: Model Based Software Development

Max. Marks : 100 **Duration: 3 Hours**

PART A

Answer All Questions. Each Question Carries 3 Marks

1. List any three advantages of automated software testing.

2. Specify the steps and their purpose in the model checking process.

3. Compare Analysis And Design Language (AADL) with Unified modeling language (UML).

4. Describe the design phase in the model based software development process.

5. Represent interface component with an out data port and an out event port in AADL.

a) textual b)graphical

6. Give the textual top level model of a powerboat autopilot system in AADL.

7. What is an error type? Mention any two pre-declared timing and value errors in AADL.

8. Define : (i) Fault Tree Analysis (ii) Minimal cutsets

9. Explain templates and filtering code generation technique.

10. How does automated code generation help to deal with faults in a software system?

(10x3=30)

Part B

(Answer any one question from each module. Each question carries 14 Marks)

11. (a) Explain model based software development approach. **(12)**

(b) Why is model based software development important? **(2)**

OR

12. (a) What are software faults? Mention any three software faults and its consequences. **(5)**

(b) Explain two approaches for ensuring software reliability? **(9)**

(i) Model Checking

(ii) Automated Testing

13. (a) Illustrate model based software development process. **(8)**

(b) Explain infrastructure of model driven architecture. **(6)**

OR

14. (a) What is AADL? Compare AADL and UML. **(6)**

(b) Explain in detail about MDA modeling levels. **(8)**

15. (a) Illustrate the components of an AADL model. **(12)**

(b) What is the AADL language syntax? **(2)**

OR

16. (a) Explain the following:

i) AADL classifiers **(2)**

ii) AADL declarations **(2)**

(b) Design an AADL model which controls the speed of a vehicle. Also describe the basic components of the designed model. **(10)**

17. (a) Illustrate how value error can be modelled using AADL in the isolette system. **(10)**

(b) With a diagram explain error propagation, termination and transformation in AADL models. **(4)**

OR

18. (a) Illustrate error state machines in AADL using proper textual representations. **(8)**

(b) Suppose we have a train door controller system with following components **(6)**

i) door_controller - ensures safe opening of the door.

ii) train_controller - sends train speed and transit status to the door_controller.

iii) alarm - triggered when an emergency occurs in other components.

Model the error flows, error propagations, component error behaviour and error properties for the value error in the component door_controller.

19. (a) Explain templates and meta model type code generation? **(4)**

(b) Illustrate how the code can be generated from an AADL model. **(10)**

OR

20. (a) Describe any four code generation techniques. **(10)**

(b) Explain the advantages of automatic code generation. **(4)**

Teaching Plan

Sl No	Contents	Number of Lecture Hours (35)
	Module 1 (Introduction) (7 Hours)	
1.1	Software faults	1
1.2	Introduction to Model Checking	1
1.3	Introduction to Automated Testing (Lecture 1)	1
1.4	Introduction to Automated Testing (Lecture 2)	1
1.5	Need for MBSD, MBSD Approach	1
1.6	Architecture centric model driven software development	1
1.7	AADL and architecture-centric model-based software systems	1
	Module 2 (Model Based Software Development) (7 Hours)	
2.1	Model based software development process	1
2.2	Overview of MBSD methodology	1
2.3	Model Driven Architecture	1
2.4	MDA Definitions and Assumptions	1
2.5	The modeling levels	1
2.6	Introduction to AADL	1
2.7	Comparison of AADL with other modeling languages	1
	Module 3 (Modeling using AADL) (7 Hours)	
3.1	Modeling in detail: AADL components	1
3.2	Modeling in detail: Developing a simple model	1

3.3	Modeling in detail: Define top level model with an example	1
3.4	AADL in detail: Explain AADL components, Language syntax	1
3.5	AADL declarations and classifiers	1
3.6	AADL system models and specifications	1
3.7	Case study: Power boat auto pilot system	1
	Module 4 (Model Analysis)(7 Hours)	
4.1	Introduction to safety analysis	1
4.2	Fault tree analysis, minimal cutsets	1
4.3	Error modeling with AADL - Error Model Libraries and Subclause Annotations	1
4.4	Error modeling with AADL - Error Types and Common Type Ontology,	1
4.5	Error modeling with AADL - Error Sources and Their Impact, Component Error Behavior	1
4.6	Error modelling with AADL - Compositional Abstraction of Error Behavior, Use of Properties in Architecture Fault Models	1
4.7	Illustrate isolette error model	1
	Module 5 (Code Generation) (7 Hours)	
5.1	Code generation and its advantages	1
5.2	Categorization	1
5.3	Code generation techniques - Templates + filtering, Template + metamodel, Frame processors	1
5.4	Code generation techniques - API-based generators, In-line generation, Code attributes	1
5.5	Code generation techniques - Code weaving Commonalities and Differences Between the Different Code generation Approaches	1
5.6	Code generation in AADL - Ocarina	1
5.7	Illustration of code generation using AADL model	1

CST463	WEB PROGRAMMING	CATEGORY	L	T	P	CREDIT	YEAR OF INTRODUCTION
		PEC	2	1	0	3	2019

Preamble: This course helps the learners to understand the web programming concepts. It includes the essential frontend and backend technologies needed for the development of web applications. The learners will have an opportunity to gain necessary web development skills such as HTML, CSS, JavaScript, PHP, MySQL integration, JSON and Laravel framework.

Prerequisite: Knowledge of Programming is required.

Course Outcomes: After the completion of the course the student will be able to

CO1	Use HyperText Markup Language (HTML) for authoring web pages and understand the fundamentals of WWW. **(Cognitive Knowledge Level: Understand)**
CO2	Construct and visually format responsive, interactive web pages using CSS and JavaScript (JS) **(Cognitive Knowledge Level: Apply)**
CO3	Construct websites using advanced sever side programming tool PHP **(Cognitive Knowledge Level: Apply)**
CO4	Develop dynamic web applications using PHP and perform MySQL database operations. **(Cognitive Knowledge Level: Apply)**
CO5	Explain the importance of object exchange formats using JSON and the MVC based web application development frameworks (Laravel) **(Cognitive Knowledge Level: Understand)**

Mapping of course outcomes with program outcomes

	PO 1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	✓				✓							✓
CO2	✓	✓	✓		✓							✓
CO3	✓	✓	✓	✓	✓							✓
CO4	✓	✓	✓	✓	✓							✓

CO5	✓	✓			✓							✓

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	**PO7**	Environment and Sustainability
PO2	Problem Analysis	**PO8**	Ethics
PO3	Design/Development of solutions	**PO9**	Individual and team work
PO4	Conduct investigations of complex problems	**PO10**	Communication
PO5	Modern tool usage	**PO11**	Project Management and Finance
PO6	The Engineer and Society	**PO12**	Life long learning

Assessment Pattern

Bloom's Category	**Continuous Assessment Tests**		**End Semester Examination Marks (%)**
	Test 1 (%)	**Test 2 (%)**	
Remember	**20**	**20**	**20**
Understand	**40**	**40**	**40**
Apply	**40**	**40**	**40**
Analyze			
Evaluate			
Create			

Mark Distribution

Total Marks	**CIE Marks**	**ESE Marks**	**ESE Duration**
150	**50**	**100**	**3**

Continuous Internal Evaluation Pattern:

Attendance	**10 marks**
Continuous Assessment Tests (Average of Internal Tests 1 & 2)	**25 marks**
Continuous Assessment Assignment	**15 marks**

Internal Examination Pattern

Each of the two internal examinations has to be conducted out of 50 marks.

First series test shall be preferably conducted after completing the first half of the syllabus and the second series test shall be preferably conducted after completing the remaining part of the syllabus. There will be two parts: Part A and Part B. Part A contains 5 questions (preferably, 2 questions each from the completed modules and 1 question from the partly completed module), having 3 marks for each question adding up to 15 marks for part A. Students should answer all questions from Part A. Part B contains 7 questions (preferably, 3 questions each from the completed modules and 1 question from the partly completed module), each with 7 marks. Out of the 7 questions, a student should answer any 5.

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 full questions from each module of which student should answer any one. Each question can have a maximum of 2 subdivisions and carries 14 marks.

Syllabus

Module – 1 (WWW, HTML)

Introduction to the Internet & WWW: Evolution of Internet & World Wide Web- Web Basics, URI's & URL-MIME.

Introduction to HTML5: Structuring & editing an HTML5 document, Fundamentals of HTML - Headings-Hyper Links- Images - Special Characters & Horizontal Rules-Lists- Tables -Forms - Internal Linking- Meta Elements-HTML5 Form input types -Input and Data List Elements and autocomplete attribute- Page Structure Elements -Multimedia-HTML5 Audio & video elements..

Module -2 (CSS, JavaScript)

Introduction to Stylesheets : Introduction to CSS-Basic syntax and structure-Inline Styles, Embedded Style Sheets, Conflict Resolution, Linking External Style Sheets-Exploring CSS Selectors-Properties, values, Positioning Elements: Absolute Positioning, Relative Positioning -

Backgrounds-List Styles-Element Dimensions- Table Layouts-Box Model and Text Flow-div and span -Basics of Responsive CSS, Media port & Media Queries.

Introduction to JavaScript : Introduction to Scripting- Programming fundamentals of JavaScript -Obtaining User Input with prompt Dialogs-Arithmetic-Decision Making -Control Statements -Functions -Arrays -Objects -Document Object Model (DOM) -Form processing

Module- 3 (PHP Basics)

PHP Language Structure: Introduction- **B**uilding blocks of PHP-Variables, Data Types -simple PHP program-Converting between Data Types- Operators and Expressions -Flow Control functions - Control statements- Working with Functions- Initialising and Manipulating Arrays--Objects- String Comparisons-String processing with Regular Expression

Module -4 (PHP- MySQL, JSON)

Advanced PHP: Form processing and Business Logic-Cookies- Sessions & MySQL Integration-Connecting to MySQL with PHP- Performing CREATE, DELETE, INSERT, SELECT and UPDATE operations on MySQL table -Working with MySQL data-Reading from Database-Dynamic Content.

Module- 5 (JSON, **Laravel)**

JSON Data Interchange Format: Syntax, Data Types, Object, JSON Schema, Manipulating JSON data with PHP

Web Development Frameworks: Laravel Overview-Features of Laravel-Setting up a Laravel Development Environment-Application structure of Laravel-Routing -Middleware-Controllers-Route Model Binding-Views-Redirections-Request and Responses.

Text Books

1. Paul J. Deitel, Harvey M. Deitel, Abbey Deitel, Internet & World Wide Web How to Program 5th Edition **[Module 1,2,3,4]**
2. Lindsay Bassett, Introduction to JavaScript Object Notation: A To-the-Point Guide to JSON 1st Edition, O'Reilly **[Module 5]**
3. Julie C. Meloni, Pearson -PHP, MySQL & JavaScript All in One, Sams Teach Yourself,5th Ed [**Module 4]**
4. Matt Stauffer," LARAVEL up and Running, A framework for building modern PHP apps"1st Edition, O'REILLY **[Module 5]**

Reference Books

1. Robert W Sebesta, Programming the World Wide Web, 7/e, Pearson Education Inc,8th Edition
2. Larry Ullman, Pearson- PHP 6 and MySQL 5 for Dynamic Web Sites: Visual QuickPro Guide
3. Eric van der Vlist, Danny Ayers, Erik Bruchez, Joe Fawcett, Alessandro Vernet",Wrox-Professional Web 2.0 Programming, Wiley-India edition
4. Web Technologies Black Book 2018(As per Mumbai University Syllabus) HTML, CSS3, JavaScript, iQuery, AJAX,PHP,XML,MVC and Laravel DT Editorial Services (ISBN: 9789386052490)

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Construct a valid HTML document for your personal Profile registration page for a Job Site www.123Jobs.com. Add relevant HTML elements in a table, to accept a minimum of 10 different fields which includes your name, address, phone, email address, your picture, your college; your branch, fields for your personal history (Minimum 3 fields), favourite theory and practical subjects (Checkbox), Username, Password(password)
2. What is MIME? Give the expansion of MIME. List four examples for MIME types. State the reason why MIME type specification is necessary in a request-response transaction between a browser and server.
3. What is codec? Recognize the role of controls attribute in <video> & <audio> tag in HTML. Use the COVID vaccination promotional video 'MySafety.mp4' in a web page with suitable HTML code, 'autoplay' option enabled and displayed in a standard dimension 750 X500.

Course Outcome 2 (CO2):

1. Organize a sample web page for the event 'Raagam2021' at your campus and use embedded Style sheets to apply a minimum 5 styles. State the Style Specification format of embedded style sheets.
2. Write CSS style rules to implement the following in a web page:
   a. to display the content of hyperlinks with yellow background color and in italics
   b. to display the contents of unordered lists in bold and in Arial font
   c. to display a background image titled "birds.jpg" with no tiling.
3. Write the code for an HTML document with embedded JavaScript scripts, which initially displays a paragraph with text "Welcome" and a button titled "Click". When the button is clicked, the message "Hello from JavaScript" in bold should replace the paragraph text

Course Outcome 3 (CO3):

1. Write a PHP program to store the name and roll no of 10 students in an Associative Array and Use foreach loop to process the array and Perform asort, rsort and ksort in the array. Illustrate with suitable output data
2. Design an HTML page which enters a given number, write a PHP program to display a message indicating, whether the number is odd or even, when clicking on the submit button.
3. Write a PHP program to compute the sum of the positive integers up to 100 using do while.

Course Outcome 4 (CO4):

1. Write a PHP form handling program to verify the user authentication credentials of a web page using MySQL connection and store the userid value as a Session variable if the userid is valid.
2. Create a valid HTML document for yourself, including your name, address, and email address. Also add your college; your major and the course. Perform form handling in PHP and process the output using POST method.
3. Write an embedded PHP script which displays the factorial of all numbers from 1 to 10 in a table in the web page. The factorial should be calculated and returned from a function. The table headings should be "Number" and "Factorial"

Course Outcome 5 (CO5):

1. What is Route Model Binding in Laravel? Which types of route model binding are supported in Laravel?
2. Explain how laravel performs route handling using routes calling controller methods?
3. List the data types used in JSON? Explain the use of parse () and stringify() functions in JSON with examples.

Model Question Paper

QP CODE:

Reg No: _____

Name: _____ **PAGES : 4**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST463

Course Name: Web Programming

Max. Marks : 100 **Duration: 3 Hours**

PART A

Answer All Questions. Each Question Carries 3 Marks

1. Define WWW. List any two examples of web server & web browser. Differentiate between URL and a domain?

2. Write the syntax of the URL? Rewrite the default URL of your university website by adding a subdomain named 'Research' and a web page named 'FAQ.html'. Also link this URL through the logo of 'kturesearch.png' placed in a web page. The FAQ page should be opened in a new window.

3. Illustrate the implementation of a JavaScript function greeting () using external .js file, to display a welcome message, when you click on a Button in an HTML page.

4. What are different ways of adjusting spacing in a text with suitable example.

5. Discuss the various CSS style sheet levels with suitable examples. How are conflicts resolved when multiple style rules apply to a single web page element?

6. Describe how input from an HTML form is retrieved in a PHP program, with an example

7. Write a PHP program to check whether a number is prime number or not.

8. Discuss the various steps for establishing PHP-MySQL connection with a MySQL

database ?

9. Describe the schema of a document implemented in JSON with suitable examples

10. Explain the role of Resource controllers in Laravel.

(10x3=30)

Part B

(Answer any one question from each module. Each question carries 14 Marks)

11. (a) Design a webpage that displays the following table. **(6)**

Food Item	Recommended Intake			
	age<15		age>15	
	gm	Kcal	gm	Kcal
Cerials	1000	2000	750	1760
NonCerials	450	800	350	600

(b) What is the difference between radio buttons and checkboxes when implemented using HTML? Write HTML code to implement a form which has the following elements: **(8)**

i. A textbox which can accept a maximum of 25 characters
ii. Three radio buttons with valid Label, Names and values
iii. Three check boxes buttons with valid Label, Names and values
iv. A selection list containing four items, two which are always visible
v. A submit button clicking on which will prompt the browser to send the form data to the server "http://www..mysite.com/reg.php" using "POST" method and reset button to clear its contents. You can use any text of your choice to label the form elements.

OR

12. (a) Write the equivalent HTML code to implement the following in a web page: (i) An image titled "birds.jpg" with a height of 100 pixels and width of 200 pixels. If the image cannot be accessed, a message "No image available" should be displayed (ii) A hyperlink to the URL "www.mysite.com/birds.jpg". The hyperlink should have the label "Click Here". **(6)**

(b) Create a static HTML document for your portfolio, which includes the following contents: your name, address, Mobile Number and email address. Also add the details about your college, university, your major and the batch **(8)**

of study. Include a picture of yourself and at least one other image (friend/pet/role model) to the document with a short description about that. Add three paragraphs about your personal history, with links to your social media profile. Also create an ordered list for describing your Skill Set & an unordered list showing your Strengths & Weaknesses.

13. (a) Illustrate the usage of JavaScript DOM in event handling and explain any three methods with example. **(8)**

(b) Write CSS and the corresponding HTML code for the following: **(6)**

i. Set the background color for the hover and active link states to "green"
ii. Set the list style for unordered lists to "square".
iii. Set "Flower.png" as the background image of the page and set 3% margin for the pages
iv. Set dashed border for left and right and double border for top & bottom of a table with 2 rows.

OR

14. (a) List the order of precedence of style levels. Organize a sample web page for providing 'KTU BTech Honours Regulation 19' for KTU and use embedded Style sheet to apply minimum 5 styles for list, tables and pages. **(6)**

(b) Illustrate the different ways of Array declaration in JavaScript. Describe the function of the following JavaScript Array object methods with examples. **(8)**
(i) join (ii) slice

15. (a) Explain any six string handling functions used in PHP with example. **(6)**

(b) How does a PHP array differ from an array in C? List the different ways to create an array in PHP with an example. Explain any 4 functions that deals with PHP array. **(8)**

OR

16. (a) During the process of fetching a web page from a web server to a client browser, at what point does an embedded PHP script get executed. What are the two modes that the PHP processor operates in? Explain **(6)**

(b) Why is PHP considered to be dynamically typed? Distinguish between **(8)**

implode and explode function in PHP with suitable examples.

17. (a) Write equivalent PHP statements corresponding to the following: **(8)**

i. Declare an associative array named "ages" to store the key-value pairs ("Alice", 30), ("Bob", 30), ("Harry", 35), ("Mary", 32).

ii. Modify the value associated with the key "Mary" to 28.

iii. Sort the array according to values maintaining the key-value relationships and print the sorted key-value pairs.

iv. The entry identified by the key "Bob"

(b) What are the uses of cookies in web pages? Describe syntax for setting cookies in PHP. How can you access and delete the cookie using setcookie() function? **(6)**

OR

18. (a) Write a PHP form handling program to perform the user registration of any website with a minimum of 5 different fields and insert the data into a MySQL table after establishing necessary connections with the DB, **(8)**

(b) Design the HTML page which enters a given number and embed the PHP code to display a message indicating, whether the number is odd or even, when clicking on the 'CHECK NUMBER' button. **(6)**

19. (a) With a neat diagram, explain about Laravel MVC Framework. **(6)**

(b) Discuss in detail about Laravel's Routing mechanisms. **(8)**

OR

20. (a) Enumerate the data types in JSON. Illustrate the document definition of a 'Student document 'using JSON Schema. **(8)**

(b) Discuss the following in Laravel Views **(6)**

i. Creating & Rendering Views

ii. Passing Data to Views

iii. Sharing Data with All Views

Teaching Plan

No	Contents	No of Lecture Hrs (35 hrs)
	Module 1 (7 hours)	
	Introduction to Internet and WWW	
1.1	Evolution of Internet &World Wide Web- Web Basics URI's & URL -MIME [Book 1 - Chapter 1]	1
	Introduction to HTML5	
1.2	Structuring & editing an HTML5 document- Fundamentals of HTML, Headings-Images [Book 1 - Chapter 2]	1
1.3	Hyper Links, Internal Linking- Lists [Book 1 - Chapter 2]	1
1.4	Special Characters & Horizontal Rules- meta Elements- div and span [Book 1 - Chapter 2]	1
1.5	Tables- Forms [Book 1 - Chapter 2]	1
1.6	HTML5 Form input types, input and data list Elements and autocomplete attributes-Page Structure Elements [Book 1 - Chapter 3]	1
1.7	Multimedia-HTML5 Audio & video elements [Book 1 - Chapter 9]	1
	Module 2 (10 hours)	
	Introduction to Cascading Style Sheets(CSS)	
2.1	Introduction to CSS3-Basic syntax and structure-Inline Styles [Book 1 - Chapter 4]	1
2.2	Embedded Style Sheets-Linking External Style Sheets [Book 1 - Chapter 4]	1
2.3	Exploring CSS Selectors-Properties-values [Book 1 - Chapter 4]	1
2.4	Positioning Elements: Absolute Positioning- Relative Positioning -Backgrounds-List Styles- Table Layouts [Book 1 - Chapter 4]	1

2.5	Box Model and Text Flow, Basics of Responsive CSS-Media port & Media Queries [Book 1 - Chapter 4]	1
Introduction to JavaScript		
2.6	Introduction to Scripting- Programming fundamentals of JavaScript -Obtaining User Input with prompt Dialogs [Book 1 - Chapter 6]	1
2.7	Arithmetic-Decision Making [Book 1 - Chapter 6]	1
2.8	Control Statements [Book 1 - Chapter 7]- Functions [Book 1 - Chapter 9]	1
2.9	Arrays [Book 1 - Chapter 10] - Objects [Book 1 - Chapter 11]	1
2.10	Document Object Model (DOM)- Form processing [Book 1 - Chapter 12,13]	1
Module 3 (6 hours)		
Introduction to PHP		
3.1	Building blocks of PHP-Variables, Data Types simple PHP program [Book 3- Chapters 4]	1
3.2	Converting between Data Types, Operators and Expressions -Flow Control functions [Book 1- Chapters 19]	1
3.3	Control Statements -Working with Functions [Book 3- Chapters 6]	1
3.4	Initialising and Manipulating Arrays- Objects [Book 1- Chapters 19]	1
3.5	Working with Strings-String processing with Regular expression, Pattern Matching [Book 1- Chapters 19]	1
3.6	Form processing and Business Logic [Book 1- Chapters 19]	1
Module 4 (6 hours)		
PHP -MYSQL		
4.1	Cookies- Sessions [Book 1- Chapters 19]	1
4.2	PHP& MySQL Integration-Connecting to MySQL with PHP . [Book 4- Chapters 18]	1

4.3	Working with MySQL data [Book 4- Chapters 18]	1
4.4	Performing CREATE, DELETE, INSERT operations on MySQL table from PHP Program. [Book 4- Chapters 16]	1
4.5	Performing SELECT and UPDATE operations on MySQL table from PHP Program. [Book 4- Chapters 16]	1
4.6	Building Dynamic Content in PHP application [Book1- Chapter19]	1
Module 5 (6 hours)		
	JSON	
5.1	JSON Data Interchange Format -Syntax, Data Types, Object [Book 2 - Chapters 1-2]	1
5.2	JSON Schema, Manipulating JSON data with PHP [Book 2 - Chapter 3,4]	1
	LARAVEL	
5.3	Laravel Overview- Design Pattern- Laravel Features [Book 4- Chapters 1] Setting up a Laravel Development Environment-Application structure of Laravel [Book 4- Chapters 2]	1
5.4	**Laravel Basics** Routing -middleware - Controllers [Book 4- Chapters 3]	1
5.5	Route Model Binding-Views-Redirections [Book 4- Chapters 3]	1
5.6	**Blade Templating-**echoing data, control structures [Book 4- Chapters 4]	1

CST473	NATURAL LANGUAGE PROCESSING	CATEGORY	L	T	P	CREDIT	YEAR OF INTRODUCTION
		PEC	2	1	0	3	2019

Preamble: This course enables the learners to understand the concepts of Natural Language Processing. The course covers basic pre-processing steps, language models, text classification using machine learning algorithms, information and relation extraction methods, Information Retrieval, Question Answer Systems and Machine Translation models. This course enables the students to apply techniques and methods to solve challenging real-world problems in NLP.

Prerequisite: Nil.

Course Outcomes: After the completion of the course the student will be able to

CO1	Summarize basic concepts and learning methods for NLP **(Cognitive Knowledge Level: Understand)**
CO2	Demonstrate the relevance of pre-processing methods on text data**(Cognitive Knowledge Level: Apply)**
CO3	Compare different language modelling techniques**(Cognitive Knowledge Level: Apply)**
CO4	Make use of NLP techniques in Text Classification and Information Retrieval**(Cognitive Knowledge Level: Apply)**
CO5	Explain Information Extraction, Relation Detection, QA Systems and Machine Translation**(Cognitive Knowledge Level: Understand)**

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	✓											✓
CO2	✓	✓	✓		✓							✓
CO3	✓	✓	✓		✓							✓
CO4	✓	✓	✓									✓
CO5	✓											✓

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	**PO7**	Environment and Sustainability
PO2	Problem Analysis	**PO8**	Ethics
PO3	Design/Development of solutions	**PO9**	Individual and team work
PO4	Conduct investigations of complex problems	**PO10**	Communication
PO5	Modern tool usage	**PO11**	Project Management and Finance
PO6	The Engineer and Society	**PO12**	Life long learning

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks (%)
	Test 1 (%)	Test 2 (%)	
Remember	**30**	**30**	**30**
Understand	**40**	**40**	**40**
Apply	**30**	**30**	**30**
Analyze			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	**50**	**100**	**3**

Continuous Internal Evaluation Pattern:

Attendance	**10 marks**
Continuous Assessment Tests (Average of Internal Tests 1 & 2)	**25 marks**
Continuous Assessment Assignment	**15 marks**

Internal Examination Pattern

Each of the two internal examinations has to be conducted out of 50 marks. First series test shall be preferably conducted after completing the first half of the syllabus and the second series test shall be preferably conducted after completing remaining part of the syllabus. There will be two parts: Part A and Part B. Part A contains 5 questions (preferably, 2 questions each from the completed modules and 1 question from the partly completed module), having 3 marks for each question adding up to 15 marks for part A. Students should answer all questions from Part A. Part B contains 7 questions (preferably, 3 questions each from the completed modules and 1 question from the partly completed module), each with 7 marks. Out of the 7 questions, a student should answer any 5.

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 full questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carries 14 marks.

Syllabus

Module – 1 (Introduction to NLP)

NLP Tasks and Applications, Language-Building Blocks, Challenges of NLP, Machine Learning for NLP – Naïve Bayes Classifier, Logistic Regression, Support Vector Machines, Approaches to NLP-- Heuristics-Based NLP, Machine Learning-based NLP.

Module – 2 (Pre-processing and Representation Models)

NLP System Pipeline--Steps--Data Acquisition, Text Extraction and Clean-up, Pre-processing, Feature Engineering, Modelling, Evaluation, Post-Modelling Phases

Text Representation--Vector Space Models--Basic Vectorization Approaches--One-Hot Encoding, Bag of Words, Bag of N-Grams TF-IDF; Distributed Representations-- Word Embeddings, Doc2Vec.

Module - 3 (Classification and Information Extraction)

Text Classification--Text classification applications – Pipeline for building text classification systems, Naïve Bayes for Sentiment Classification – Naïve Bayes Classifier Training – Optimizing for Sentiment Analysis, Logistic Regression, Support Vector Machine for Text Classification

Information Extraction(IE)—IE Applications – The General Pipeline for IE - Named Entity Recognition(NER), Ambiguity in Named Entity Recognition – NER as Sequence Labeling – Evaluation of NER.

Module - 4 (Relation Detection and Information Retrieval)

Relation Detection and Classification – Supervised Learning Approaches to Relation Analysis – Lightly Supervised Approaches to Relation Analysis – Evaluation of Relation Analysis systems

Information Retrieval – Term weighting and document scoring – Inverted Index – Evaluation of Information Retrieval Systems.

Module - 5 (QA Systems and Machine Translation)

Question-Answering Systems – Factoid Question Answering – Question Processing – Passage Retrieval – Answer Processing – Evaluation of Factoid Answers

Machine Translation – Why Machine Translation is Hard – Classical Machine Translation – Direct Translation – Transfer – Statistical Machine Translation- The Phrase based Translation model – Alignment in MT – Training Alignment Models – Symmetrizing Alignments for Phrase-based MT – Decoding for Phrase-based Statistical MT

Text Books

1. Daniel Jurafsky, James H. Martin , "Speech and Language Processing"(2nd and 3rd editions), Pearson Prentice Hall
2. Sowmya Vajjala, Bodhisattwa Majumder, Anuj Gupta, Harshit Surana," Practical Natural Language Processing: A Comprehensive Guide to Building Real-World NLP Systems " June 2020 Publisher(s): O'Reilly Media, Inc. ISBN: 9781492054054.

Reference Books

1. James Allen, "Natural Language Understanding", Second Edn , Pearson.
2. Christopher Manning and Hinrich Schutze, Statistical Natural Language Processing, MIT Press.

Course Level Assessment Questions

Course Outcome1 (CO1):

1. Explain the fundamental tasks that make up an NLP system.
2. Why is NLP considered a challenging problem domain?
3. The following table shows data about the profile of customers and whether they purchase computers or not. Given this data, use Naïve Bayes Classifier to classify the customer ***X*** (*age = youth, income = medium, student = yes, credit rating = fair*)

RID	age	income	student	credit_rating	Class: buys_computer
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

4. Illustrate how linearly inseparable data can be made linearly separable by suitable mapping using kernel functions.

Course Outcome 2(CO2):

1. Mention two issues associated with sentence segmentation.
2. Show how is lemmatization done using Python Library.
3. Given a dataset of tweets, prepare the data for sentiment analysis by doing the following operations: conversion to lower casing, removal of punctuations, removal of stop-words, stemming, lemmatization, removal of emojis and removal of URLs. (Assignment Question)

Course Outcome 3(CO3):

1. Compare Bag-of-Words model and Bag-of-n-gram model.
2. Illustrate how TF-IDF model is used to represent text. Mention the advantage of TF-IDF over other models.
3. A corpus of data is given below :

D1	Dog bites man.
D2	Man bites dog.
D3	Dog eats meat.
D4	Man eats food.

Use one hot-encoding and Bag-of-words models to represent “dog bites man”.

Using the toy corpus given above, represent the sentence “Dog and Man eat meat” with TF-IDF model. Use python code for implementation. (Assignment Question)

Course Outcome 4(CO4): .

1. Given the following data about documents and contents, use tf-idf document scoring method to retrieve the document for the query "best game"

Doc 1	The game was so exciting. The players excelled in every department of the game.
Doc 2	It was an excellent game.
Doc 3	The game was not good. The moves were boring

2. A corpus of data is available from a social media platform that represents review of books. How can Naïve Bayes Classifier be used for sentiment analysis of the reviews? What changes can be made to this classifier to make it tuned for sentiment analysis.
3. Use python library to implement sentiment analysis of review of a book, given a toy corpus data set given below. (Assignment Question)

Document	Category
just plain boring	Negative
entirely predictable and lacks energy	Negative
no surprises and very few laughs	Negative
very powerful book	Positive
the best book of the summer	Positive

Course Outcome 5(CO5):

1. Explain lightly supervised approaches to relational analysis.
2. Explain a statistical algorithm for word alignment in Machine Translation.

Model Question Paper

QP CODE:

Reg No: _____

Name: _____ **PAGES : 4**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST473

Course Name: Natural Language Processing

Max. Marks : 100 **Duration: 3 Hours**

PART A

Answer All Questions. Each Question Carries 3 Marks

1. Differentiate information extraction and information retrieval.
2. State Bayes' Theorem.
3. List three preprocessing steps that are necessary for an HTML file.
4. Differentiate CBOW and Skipgram models
5. Explain the role of support vectors in SVM Classification.
6. Explain challenges in Name Entity Recognition.
7. How is a Relational Analysis System evaluated?
8. Explain the need for an inverted index in an information retrieval system. Are there any more efficient data structures that serve the same purpose.
9. How do you extract answers to DEFINITION questions?
10. What are the components that make up a noisy channel model of statistical Machine Translation?

(10x3=30)

Part B

(Answer any one question from each module. Each question carries 14 Marks)

11. (a) How is classification done by SVM on linearly separable data? **(8)**

(b) What is a kernel function? What is the need for a kernel function? Can a kernel function be replaced by an ordinary mapping function? **(4)**

(c) Explain Heuristic-based NLP. **(2)**

OR

12. (a) Illustrate the steps involved in classification in Naïve Bayes Classifier. **(8)**

(b) Explain the fundamental tasks that make up an NLP system. **(6)**

13. (a) Supposing that a set of social media posts' dataset is available to do sentiment analysis. What pre-processing steps need to be done in order to use the data for generating a language model? Illustrate. **(8)**

(b) Illustrate Bag-of-ngrams model with an example. **(6)**

OR

14. (a) Explain the concept of word embeddings as a model for text representation. **(6)**

(b) Compare word embeddings model with vectorization approaches. **(4)**

(c) Explain the concept of feature engineering in NLP Systems. **(4)**

15. (a) 1. Given the following data about movie review and its classification, classify "predictable with no fun" to one of the classes using Naïve Bayes Classifier. **(10)**

Document	Category
just plain boring	Negative
entirely predictable and lacks energy	Negative
no surprises and very few laughs	Negative
very powerful	Positive
the most fun film of the summer	Positive

(b) Explain challenges in Name Entity Recognition. **(4)**

OR

16. (a) Explain Logistic Regression for Text Classification. **(6)**

(b) Explain Name Entity Recognition using Sequence Labeling. **(8)**

17. (a) Explain supervised approach to relation analysis. What are its limitations? **(10)**

(b) How is term selection done for indexing? **(4)**

OR

18. (a) Given the following data about documents and contents, use tf-idf document scoring method to retrieve the document for the query "sweet love". **(10)**

Doc 1	Sweet sweet nurse! Love
Doc 2	Sweet sorrow
Doc 3	How sweet is love?
Doc 4	Nurse!

(b) Explain the approaches to evaluate a relation analysis system. **(4)**

19. (a) Explain the phases of a factoid question-answering system. **(8)**

(b) Give an algorithm for word alignment in Machine Translation. **(6)**

OR

20. (a) How is decoding done in a Phrase-based Statistical Machine Translation System? **(10)**

(b) Explain the concept of Mean Reciprocal Rank. **(4)**

Teaching Plan

No	Contents	No of Lecture Hrs: 35
Module 1 : Introduction to NLP (7 hours)		
1.1	Introduction to NLP – Tasks and Applications	1
1.2	Language – Building Blocks, Challenges of NLP	1
1.3	Approaches to NLP - Heuristics-Based NLP, Machine Learning for NLP	1
1.4	Machine Learning for NLP – Naïve Bayes Classifier	1
1.5	Logistic Regression	1
1.6	Support Vector Machines – Linearly Separable Data	1
1.7	Support Vector Machines – Linearly Inseparable Data	1
Module 2 : Pre-processing and Representation Models(7 hours)		
2.1	NLP System Pipeline – Stages – Overview, Data Acquisition	1
2.2	NLP System Pipeline – Text Extraction and Cleanup	1
2.3	NLP System Pipeline – Preprocessing - Sentence segmentation, Word tokenization, Stemming and lemmatization	1
2.4	Feature Engineering, Model Building, Evaluation – Metrices, Post-modeling phase	1
2.5	Text Representation – Vector Space Model, Vectorization Approaches – One hot encoding, Bag of words	1
2.6	Bag of n-grams, TF-IDF	1
2.7	Word Embeddings – Word2Vec- CBOW, SkipGram models	1
Module 3: Classification and Information Extraction(7 hours)		
3.1	Text Classification--Text classification applications – Pipeline for building text classification systems	1
3.2	Sentiment Analysis using Naïve Bayes Classifier	1
3.3	Case Studies for Text Classification using Logistic Regression and	1

	Support Vector Machines	
3.4	Information Extraction (IE) and Applications, IE Tasks and the IE Pipeline	1
3.5	Named Entity Recognition (NER) – Ambiguity in NER	1
3.6	NER as Sequence Labeling	1
3.7	Evaluation of NER, Practical NER Systems	1
Module 4 : Relation Detection and Information Retrieval(5 hours)		
4.1	Relation Detection and Classification – Supervised Learning Approaches to Relation Analysis	1
4.2	Relation Detection and Classification – Lightly Supervised Approaches to Relation Analysis	1
4.3	Relation Detection and Classification -Evaluation of Relation Analysis systems	1
4.4	Information Retrieval – Term weighting and document scoring	1
4.5	Inverted Index, Evaluation of Information-Retrieval Systems	1
Module 5 : QA Systems and Machine Translation (9 hours)		
5.1	Question-Answering Systems – Factoid Question Answering, Question Processing	1
5.2	Passage Retrieval	1
5.3	Answer Processing, Evaluation of Factoid Answers	1
5.4	Machine Translation – Why Machine Translation is Hard	1
5.5	Classical Machine Translation	1
5.6	Statistical Machine Translation	1
5.7	The Phrase based Translation model	1
5.8	Alignment in Machine Translation	1
5.9	Decoding for Phrase-based Statistical MT	1

SEMESTER VII

OPEN ELECTIVE

CST415	INTRODUCTION TO MOBILE COMPUTING	CATEGORY	L	T	P	CREDIT	YEAR OF INTRODUCTION
		OEC	2	1	0	3	2019

Preamble: The purpose of this course is to prepare learners to understand the functionalities and design considerations of mobile computing. The course content is designed to cover the mobile computing architecture, features of different communication systems and major elements of mobile security and next generation computer systems. This course enables the learners to acquire advanced concepts on mobile and ad-hoc networks.

Prerequisite: A good knowledge of data communication and computer networks.

Course Outcomes: After the completion of the course the student will be able to

CO#	Course Outcomes
CO1	Describe the mobile computing applications, services, design considerations and architectures(**Cognitive knowledge: Understand**)
CO2	Identify the technology trends for cellular wireless networks**(Cognitive knowledge:Understand)**
CO3	Summarize the Short Messaging Service and General Packet Radio Service **(Cognitive knowledge: Understand)**
CO4	Outline the LAN technologies used in mobile communication **(Cognitive knowledge: Understand)**
CO5	Describe the security protocols and apply suitable security algorithm to secure the communication **(Cognitive knowledge: Apply)**
CO6	Explain the fundamental concepts of next generation mobile networks**(Cognitive knowledge: Understand)**

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	✓	✓										✓
CO2	✓	✓	✓									✓
CO3	✓	✓	✓									✓
CO4	✓	✓	✓									✓
CO5	✓	✓	✓									✓
CO6	✓	✓	✓									✓

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	**PO7**	Environment and Sustainability
PO2	Problem Analysis	**PO8**	Ethics
PO3	Design/Development of solutions	**PO9**	Individual and team work
PO4	Conduct investigations of complex problems	**PO10**	Communication
PO5	Modern tool usage	**PO11**	Project Management and Finance
PO6	The Engineer and Society	**PO12**	Life long learning

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination (%)
	Test 1 (%)	Test 2 (%)	
Remember	30	30	30
Understand	50	50	50
Apply	20	20	20
Analyse			
Evaluate			

Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3

Continuous Internal Evaluation Pattern:

Attendance : **10 marks**

Continuous Assessment Test : **25 marks**

Continuous Assessment Assignment : **15 marks**

Internal Examination Pattern:

Each of the two internal examinations shall be conducted for 50 marks. First series test shall be conducted preferably after completing the first half of the syllabus and the second series test shall be conducted preferably after completing the remaining part of the syllabus. There shall be two parts for the question paper: Part A and Part B. Part A shall contain five questions (preferably, two questions each from the fully completed modules and one question from the partly covered module), having three marks for each question adding up to 15 marks for part A. A student is expected to answer all questions from Part A. Part B shall contain seven questions (preferably, three questions each from the fully completed modules and one question from the partially completed module), each having seven marks. Out of the seven questions, a student is expected to answer any five.

End Semester Examination Pattern:

There shall be two parts; Part A and Part B. Part A shall contain 10 questions with 2 questions from each module, having 3 marks for each question. A student is expected to answer all questions from Part A. Part B shall contain 2 questions from each module, out of which a student is expected to answer any one. Each question shall have a maximum of two sub-divisions and shall carry 14 marks.

Syllabus

Module-1 (Mobile Computing Architecture)

Introduction to mobile computing – Functions, Devices, Middleware and gateways, Applications and services, Limitations. Mobile computing architecture – Internet: The ubiquitous network, Three-tier architecture, Design considerations for mobile computing.

Module-2 (Communication Systems)

Mobile computing through telephony - Evolution of telephony, Multiple access procedures - Frequency Division Multiple Access (FDMA), Time Division Multiple Access (TDMA), Code Division Multiple Access (CDMA), Space Division Multiple Access (SDMA). Satellite communication systems – Basics, Applications, Geostationary Earth Orbit (GEO), Low Earth Orbit (LEO), Medium Earth Orbit (MEO), Satellite phones. Mobile computing through telephone – Interactive Voice Response (IVR) architecture, Overview of voice software, Developing an IVR application. Global System for Mobile Communication (GSM) - Introduction, Architecture, Entities, Call routing, Mobility management, Frequency allocation, Authentication and security.

Module-3 (Short Messaging Service and General Packet Radio Service)

Short Message Service (SMS) – Strengths, Architecture, Value added services, Accessing the SMS bearer. General Packet Radio Service (GPRS) – Architecture, Network operations, Data services, Applications, Limitations, Billing and charging.

Module-4 (Wireless Local Area Networks)

Wireless Local Area Network (WLAN) - Advantages, Evolution, Applications, Architecture, Mobility, Security, Deploying WLAN. Wireless Local Loop (WLL) – Architecture. High Performance Radio Local Area Network (HIPERLAN). WiFi Vs 3G.

Module-5 (Mobile Security and Next Generation Networks)

Security issues in mobile computing - Information security, Security techniques and algorithms, Security protocols. Next generation networks – The Converged Scenario, Narrowband to broadband, Orthogonal Frequency Division Multiplexing (OFDM), Multi Protocol Label Switching (MPLS), Wireless Asynchronous Transfer Mode (WATM), Multimedia broadcast services.

Text Books

1. Asoke K. Talukder, Hasan Ahmad, Roopa R Yavagal, Mobile Computing Technology- Application and Service Creation, 2nd Edition, McGraw Hill Education.
2. Schiller J., Mobile Communications, 2/e, Pearson Education, 2009.

Reference Books

1. Andrew S. Tanenbaum, Computer Networks, 6/e, PHI.
2. Theodore S. Rappaport, Wireless Communications Principles and Practice, 2/e, PHI, New Delhi, 2004.
3. Curt M. White, Fundamentals of Networking and Communication 7/e, Cengage learning.

Course Level Assessment Questions

Course Outcome 1 CO1):

1. Describe the design considerations in mobile computing.
2. Give five examples of mobile computing applications.

Course Outcome 2 (CO2):

1. Draw a call flow diagram for a theatre ticket booking system.
2. Illustrate the GSM architecture with figure.

Course Outcome 3 (CO3):

1. Illustrate the billing and charging services in GPRS.
2. Describe the SMS architecture.

Course Outcome 4 (CO4):

1. Compare IEEE 802.11, HIPERLAN with respect to their ad-hoc capabilities.
2. Discuss the security mechanism used in WLAN.

Course Outcome 5 (CO5):

1. With the help of a suitable example, show the working of Diffie-Hellman key exchange algorithm.
2. Bob chooses 7 and 11 as two prime numbers and chooses e as 13. Find an appropriate value for d and decrypt the plaintext 5 send by Alice to Bob.
3. Describe the security issues in mobile computing.

Course Outcome 6 (CO6):

1. Describe WATM and Multimedia broadcast services.
2. Describe the significance of Orthogonal Frequency Division Multiplexing (OFDM) in next generation networks.

Model Question Paper

QP CODE: **PAGES: 3**

Reg No:_____
Name:_____

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST415

Course Name : INTRODUCTION TO MOBILE COMPUTING

Max Marks: 100 **Duration: 3 Hours**

PART-A
(Answer All Questions. Each question carries 3 marks)

1. Explain the different types of middleware and gateways required in mobile computing.
2. List any six limitations of mobile computing.
3. Compare and contrast the satellite systems – GEO, LEO and MEO.
4. How is frequency allocation done in GSM?
5. What are the various strengths of SMS?
6. How is billing and charging done in GPRS?
7. What are the different types of Wireless LANs?
8. Describe the architecture of a Wireless Local Loop.
9. Explain the key features of TLS protocol.
10. How are attacks classified?

(10x3=30)

Part B
(Answer any one question from each module. Each question carries 14 Marks)

11. (a) Describe any four mobile computing functions. **(4)**

    (b) Explain the three-tier architecture of mobile computing with figure. **(10)**

OR

12. (a) Describe the significance and functions of core, edge and access network. **(6)**

    (b) Explain the terms (i) Client Context Manager (ii) Policy Manager (iii) Security Manager (iv) Adaptability Manager **(8)**

13. (a) Why is multiple access important? With the help of suitable examples, explain the various multiple access techniques. **(7)**

    (b) Describe the different algorithms used for security and authentication in GSM. **(7)**

OR

14. (a) Show how call routing is done in GSM. Give an example. **(7)**

    (b) Explain the process of handover. How does handover differ from roaming? **(7)**

15. (a) With the help of neat sketches, explain the difference between Short Message Mobile Terminated (SM MT) and Short Message Mobile Originated (SM MO) messages. **(6)**

    (b) Explain the network operations in GPRS. **(8)**

OR

16. (a) How does operator-centric pull differ from operator-independent push and pull? **(7)**

    (b) Describe the data services and applications of GPRS. **(7)**

17. (a) Compare the HIPERLAN and OSI layered architecture. **(4)**

    (b) Explain the 802.11 architecture. **(10)**

OR

18. (a) Compare 3G and WiFi. **(7)**

    (b) Explain the HIPERLAN communication models with suitable diagrams. **(7)**

19. (a) Given p = 7, q = 17 and e = 5. Find the value of d and also encrypt the message P = 65 using RSA. **(7)**

    (b) Explain the role of MPLS in service provisioning. **(7)**

OR

20. (a) With the help of a suitable example, show the working of Diffie-Hellman key exchange algorithm. **(7)**

    (b) Explain the features of any three multimedia broadcast services. **(7)**

TEACHING PLAN

No	Contents	No.of Lecture Hrs (35 hrs)
Module-1 (Mobile Computing Architecture) (6 hrs)		
1.1	Introduction to mobile computing – Functions, Devices, Middleware and gateways	1
1.2	Applications, services, limitations, Internet: The ubiquitous network	1
1.3	Three-tier architecture (Lecture 1)	1
1.4	Three-tier architecture (Lecture 2)	1
1.5	Design considerations for mobile computing (Lecture 1)	1
1.6	Design considerations for mobile computing (Lecture 2)	1
Module-2 (Communication Systems) (7hrs)		
2.1	Evolution of telephony, Multiple access procedures – FDMA, TDMA, CDMA, SDMA	1
2.2	Satellite communication systems – GEO, MEO, LEO, Satellite phones	1
2.3	Interactive Voice Response (IVR) architecture, Overview of voice software, Developing an IVR application (Call flow diagram)	1
2.4	Introduction to GSM,Architecture	1
2.5	GSM entities, Call routing	1
2.6	Mobility management	1
2.7	Frequency allocation, Authentication and security	1
Module-3 (Short Messaging Service and General Packet Radio Service) (8hrs)		
3.1	SMS Strengths, Architecture, Short Message Mobile Terminated (SM MT) and Short Message Mobile Originated (SM MO) messages	1
3.2	SMS Architecture - Operator-centric pull, operator-	1

	independent push/pull, Value added services	
3.3	Accessing the SMS bearer (Lecture 1)	1
3.4	Accessing the SMS bearer (Lecture 2)	1
3.5	GPRS architecture	1
3.6	Network operations	1
3.7	Data services, Applications	1
3.8	Limitations, Billing and charging	1
Module-4 (Wireless Local Area Networks) (7 hrs)		
4.1	WLAN Advantages, Evolution, Applications	1
4.2	WLAN Architecture (Lecture 1)	1
4.3	WLAN Architecture (Lecture 2)	1
4.4	Mobility, Security	1
4.5	Deploying WLAN	1
4.6	WLL Architecture, HIPERLAN	1
4.7	WiFi Vs 3G	1
Module-5 (Mobile Security and Next Generation Networks) (7hrs)		
5.1	Information security – Attacks, Components	1
5.2	Security techniques and algorithms – Stream Vs Block cipher, Symmetric Vs Asymmetric cryptography	1
5.3	Security techniques and algorithms – RSA, Diffie Hellman Key exchange	1
5.4	Security protocols – Secure Socket Layer, Transport Layer Security, Wireless Transport Layer Security	1
5.5	The Converged Scenario, Narrowband to broadband	1
5.6	Orthogonal Frequency Division Multiplexing (OFDM) and Multi Protocol Label Switching (MPLS)	1
5.7	Wireless Asynchronous Transfer Mode (WATM) and Multimedia broadcast services	1

CST425	INTRODUCTION TO DEEP LEARNING	CATEGORY	L	T	P	CREDIT	YEAR OF INTRODUCTION
		OEC	2	1	0	3	2019

Preamble: This course aims to introduce the learner to an overview of the concepts and algorithms involved in deep learning. Basic concepts and application areas of machine learning, deep networks, convolutional neural network and recurrent neural network are covered in this course. This is a foundational program that will help students understand the capabilities, challenges, and consequences of deep learning and prepare them to participate in the development of leading-edge AI technology. They will be able to gain the knowledge needed to take a definitive step in the world of AI.

Prerequisite: Basics of linear algebra and probability.

Course Outcomes: After the completion of the course the student will be able to

CO1	Demonstrate basic concepts in machine learning.(**Cognitive Knowledge Level: Understand)**
CO2	Illustrate the validation process of machine learning models using hyper-parameters and validation sets. **(Cognitive Knowledge Level: Understand)**
CO3	Demonstrate the concept of the feed forward neural network and its training process. **(Cognitive Knowledge Level: Apply)**
CO4	Build CNN and Recurrent Neural Network (RNN) models for different use cases. **(Cognitive Knowledge Level: Apply)**
CO5	Use different neural network/deep learning models for practical applications. **(Cognitive Knowledge Level: Apply)**

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	✓	✓										✓
CO2	✓	✓										✓
CO3	✓	✓	✓	✓								✓
CO4	✓	✓	✓	✓								✓
CO5	✓	✓	✓	✓								✓

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	**PO7**	Environment and Sustainability
PO2	Problem Analysis	**PO8**	Ethics
PO3	Design/Development of solutions	**PO9**	Individual and team work
PO4	Conduct investigations of complex problems	**PO10**	Communication
PO5	Modern tool usage	**PO11**	Project Management and Finance
PO6	The Engineer and Society	**PO12**	Lifelong learning

Assessment Pattern

Bloom's Category	**Continuous Assessment Tests**		End Semester Examination Marks (%)
	Test 1 (%)	**Test 2 (%)**	
Remember	**30**	**30**	**30**
Understand	**30**	**30**	**30**
Apply	**40**	**40**	**40**
Analyze			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks

Continuous Assessment Tests : 25 marks

Continuous Assessment Assignment : 15 marks

Internal Examination Pattern:

Each of the two internal examinations has to be conducted out of 50 marks

First Internal Examination shall be preferably conducted after completing the first half of the syllabus and the Second Internal Examination shall be preferably conducted after completing remaining part of the syllabus.

There will be two parts: Part A and Part B. Part A contains 5 questions (preferably, 2 questions each from the completed modules and 1 question from the partly covered module), having 3 marks for each question adding up to 15 marks for part A. Students should answer all questions from Part A. Part B contains 7 questions (preferably, 3 questions each from the completed modules and 1 question from the partly covered module), each with 7 marks. Out of the 7 questions in Part B, a student should answer any 5.

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which a student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Syllabus

Module-1 (Introduction)

Key components - Data, models, objective functions, optimization algorithms, Learning algorithms. Supervised learning- regression, classification, tagging, web search, page ranking, recommender systems, sequence learning, Unsupervised learning, Reinforcement learning, Historical Trends in Deep Learning. Other Concepts - overfitting, underfitting , hyperparameters and validation sets, estimators, bias and variance.

Module- 2 (Optimization and Neural Networks)

Neural Networks –Perceptron, Gradient Descent solution for Perceptron, Multilayer perceptron, activation functions, architecture design, chain rule, back propagation, gradient based learning. Introduction to optimization– Gradient based optimization, linear least squares. Stochastic gradient descent, Building ML algorithms and challenges.

Module -3 (Convolutional Neural Network)

Convolutional Neural Networks – convolution operation, motivation, pooling, Convolution and Pooling as an infinitely strong prior, variants of convolution functions, structured outputs, data types, efficient convolution algorithms.

Module- 4 (Recurrent Neural Network)

Recurrent neural networks – Computational graphs, RNN design, encoder – decoder sequence to sequence architectures, deep recurrent networks, recursive neural networks, modern RNNs LSTM and GRU, Practical use cases for RNNs.

Module-5 (Application Areas)

Applications – computer vision, speech recognition, natural language processing. Research Areas – Autoencoders, Representation learning, Boltzmann Machines, Deep belief networks.

Text Book

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT Press 2015 ed.
2. Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola, Dive into Deep Learning, August 2019.
3. Neural Networks and Deep Learning: A Textbook by Charu C. Aggarwal. Springer.1st edition, 2018.

Reference Books

1. Neural Smithing: Supervised Learning in Feed forward Artificial Neural Networks by Russell Reed, Robert J MarksII, 1st edition, 1999, MIT Press.
2. Practical Convolutional Neural Networks by Mohit Sewak, Md. Rezaul Karim, Pradeep Pujari, 1st edition, 2018, Packt Publishing Ltd.
3. Hands-On Deep Learning Algorithms with Python by Sudharsan Ravichandran, 1st edition, 2019, Packt Publishing Ltd.
4. Deep Learning with Python by Francois Chollet, 2nd edition, 2018, Manning Publications Co.

Sample Course Level Assessment Questions

Course Outcome1(CO1):

1. Compare regression and classification.
2. Define supervised learning? Distinguish between regression and classification.
3. Discuss the different learning approaches used in machine learning.
4. You train an initial model that achieves a 90% accuracy on the training dataset. What kind of problems your model is experiencing, and suggest a possible solution.
5. How does splitting a dataset into train, validation and test sets help identify overfitting?
6. Consider solving a classification task. You first train your network on 20 samples. Training converges, but the training loss is very high. You then decide to train this network on 10,000 examples. Is your approach to fixing the problem correct? If yes, explain the most likely results of training with 10,000 examples. If not, give a solution to this problem.

7. Describe one advantage of using mini-batch gradient descent instead of full-batch gradient descent.
8. Sketch the typical learning curves for the training and validation sets, for a setting where overfitting occurs at some point. Assume that the training set and the validation set are of the same size

Course Outcome 2(CO2):

1. What are hyperparameters? Why are they needed?
2. What issues are to be considered while selecting a model for applying machine learning in a given problem?
3. Update the parameters V11 in the given MLP using back propagation with learning rate as 0.5 and activation function as sigmoid. Initial weights are given as V11= 0.2, V12=0.1, V21=0.1, V22=0.3, V11=0.2, W11=0.5, W21=0.2
4. Draw the architecture of a multi-layer perceptron.
5. Derive update rules for parameters in the multi-layer neural network through the gradient descent.
6. Why is it important to place non-linearities between the layers of neural networks?
7. You design a fully connected neural network architecture where all activations are sigmoids. You initialize the weights with large positive numbers. Is this a good idea? Explain your answer.
8. You are doing full batch gradient descent using the entire training set (not stochastic gradient descent). Is it necessary to shuffle the training data? Explain your answer.
9. Consider training a fully-connected neural network with 5 hidden layers, each with 10 hidden units. The input is 20-dimensional and the output is a scalar. What is the total number of trainable parameters in your network?
10. Consider building a 10-class neural network classifier. Given a cat image, you want to classify which of the 10 cat breeds it belongs to. What loss function do you use? Introduce the appropriate notation and write down the formula of the loss function.
11. Why is the sigmoid activation function susceptible to the vanishing gradient problem?

Course Outcome 3 (CO3):

1. Give two benefits of using convolutional layers instead of fully connected ones for visual tasks.
2. Suppose that a CNN was trained to classify images into different categories. It performed well on a validation set that was taken from the same source as the training set but not on a testing set. What could be the problem with the training of such a CNN? How will you ascertain the problem? How can those problems be solved?
3. You are given a dataset of 10 x 10 grayscale images. Your goal is to build a 5-class classifier. You have to adopt one of the following two options: a) the input is flattened into a 100-dimensional vector, followed by a fully-connected layer with 5

neurons, b) the input is directly given to a convolutional layer with five 10 x 10 filters. Explain which one you would choose and why.

4. Weight sharing allows CNNs to deal with image data without using too many parameters. Does weight sharing increase the bias or the variance of a model?
5. Why do the layers in a deep architecture need to be non-linear?
6. A convolutional neural network has 4 consecutive layers as follows:
   3 x 3 conv (stride 2) - 2 x 2 Pool - 3 x 3 conv (stride 2) - 2 x 2 Pool
   How large is the set of image pixels which activate a neuron in the 4th non-image layer of this network?
7. Consider a convolution layer. The input consists of 6 feature maps of size 20 x 20. The output consists of 8 feature maps, and the filters are of size 5 x 5. The convolution is done with a stride of 2 and zero padding, so the output feature maps are of size 10 x 10. Determine the number of weights in this convolution layer

Course Outcome 4(CO4):

1. Explain how the cell state is updated in the LSTM model from Ct-1 to Ct
2. Show the steps involved in an LSTM to predict stock prices.
3. Illustrate the workings of the RNN with an example of a single sequence defined on a vocabulary of four words.
4. If we have a recurrent neural network (RNN), we can view it as a different type of network by "unrolling it through time". Briefly explain what that means.
5. Briefly explain how “unrolling through time” is related to “weight sharing” in convolutional networks.
6. Explain how the cell state is updated in the LSTM model from Ct-1 to Ct
7. Show the steps involved in an LSTM to predict stock prices.Give one advantage of using an RNN rather than a convolutional network.

Course Outcome 5 (CO5):

1. Development a deep learning solution for problems in the domain i) natural language processing or ii Computer vision (Assignment)
2. Is an autoencoder for supervised learning or for unsupervised learning? Explain briefly.
3. Sketch the architecture of an autoencoder network.
4. Describe how to train an autoencoder network.
5. Write down the formula for the energy function (E) of a Restricted Boltzmann Machine (RBM).

Model Question Paper

QP CODE:

Reg No: _____

Name: _____ **PAGES : 4**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

EIGHTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST425

Course Name: Introduction To Deep Learning

Max. Marks : 100 **Duration: 3 Hours**

PART A

Answer All Questions. Each Question Carries 3 Marks

1. Distinguish between supervised learning and Reinforcement learning. Illustrate with an example.
2. Differentiate classification and regression.
3. Compare overfitting and underfitting. How it can affect model generalization.
4. Why does a single perceptron cannot simulate simple XOR function? Explain how this limitation is overcome?
5. Illustrate the strengths and weaknesses of convolutional neural networks.
6. Illustrate convolution and pooling operation with an example
7. How many parameters are there in AlexNet? Why the dataset size (1.2 million) is important for the success of AlexNet?
8. Explain your understanding of unfolding a recursive or recurrent computation into a computational graph.
9. Illustrate the use of deep learning concepts in Speech Recognition.
10. What is an autoencoder? Give one application of an autoencoder

(10x3=30)

Part B

(Answer any one question from each module. Each question carries 14 Marks)

11. (a) "A computer program is said to learn from experience E with respect to some class oftasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E." What is your understanding of the terms task, performance and experience. Explain with two example **(10)**

    (b) "How does bias and variance trade-off affect machine learning algorithms? **(4)**

OR

12. (a) Illustrate the concepts of Web search, Page Ranking, Recommender systems with suitable examples. **(10)**

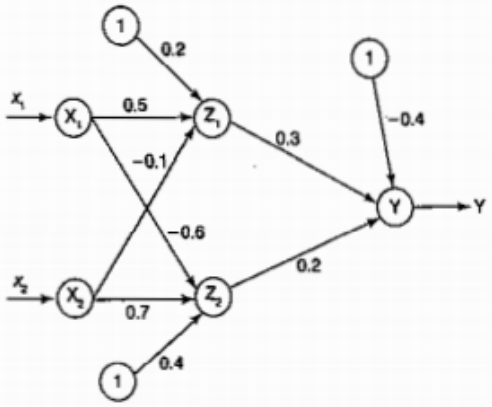
(b) List and discuss the different hyper parameters used in fine tuning the traditional machine learning models **(4)**

13. (a) How multilayer neural networks learn and encode higher level features from input features. **(7)**

(b) Explain gradient decent and delta rule? Why stochastic approximation to gradient descent is needed? **(7)**

OR

14. (a) Find the new weights for the network using backpropogation algorithm, the network is given with a input pattern[-1,1] and target output as +1, Use learning rate of alpha=0.3 and bipolar sigmoid function. **(7)**



(b) Write an algorithm for backpropagation which uses stochastic gradient descent method. Comment on the effect of adding momentum to the network. **(7)**

15. (a) Input to CNN architecture is a color image of size 112x112x3. The first convolution layer comprises of 64 kernels of size 5x5 applied with a stride of 2 and padding 0. What will be the number of parameters? **(5)**

(b) Let X=[-1, 0, 3, 5] W=[.3 ,.5 .2,.1] be the the input of i^{th} layer of a neural network and to apply softmax function. What should be the output of it? **(4)**

(c) Draw and explain the architecture of convolutional network **(5)**

OR

16. (a) Explain the concept behind i) Early stopping ii) dropout iii) weight decay **(9)**

(b) How backpropagation is used to learn higher-order features in a convolutional Network? **(5)**

17. (a) Explain the working of RNN and discuss how backpropagation through time is used in recurrent networks. **(8)**

(b) Describe the working of a long short term memory in RNNs. **(6)**

OR

18. (a) What is the vanishing gradient problem and exploding gradient problem? **(8)**

(b) Why do RNNs have a tendency to suffer from exploding/vanishing gradient? How to overcome this challenge? **(6)**

19. (a) Explain any two word embedding techniques **(8)**

(b) Explain the merits and demerits of using Auto encoders in Computer Vision. **(6)**

OR

20. (a) Illustrate the use of representation learning in object classification. **(7)**

(b) Compare Boltzmann Machine with Deep Belief Network. **(7)**

Teaching Plan

No	Contents	No. of Lecture Hours (37 hrs)
Module 1 : Introduction (8 hours)		
1.1	Key components - Data, models, objective functions, optimization algorithms. (TB2: Section 1.1-1.2)	1
1.2	Learning algorithm (TB1: Section 5.1), Supervised learning- regression, classification, tagging, web search, page ranking (TB2: Section 1.3.1)	1
1.3	Recommender systems, Sequence learning, Unsupervised learning, Reinforcement learning(TB2: Section 1.3.2-1.3.4)	1
1.4	Historical Trends in Deep Learning (TB1: Section 1.2).	1
1.5	Concepts: overfit, underfit, hyperparameters and validation sets. (TB1: Section 5.2-5.3)	1
1.6	Concepts: Estimators, bias and variance. (TB1: Section 5.4)	1
1.7	Demonstrate the concepts of supervised learning algorithms using a suitable platform.	1
1.8	Demonstrate the concepts of unsupervised using a suitable platform.	1
Module 2 : Optimization and Neural Networks (9 hours)		
2.1	Perceptron, Stochastic Gradient descent, Gradient descent solution for perceptron (TB3: Section 1.1 - 1.2.1)	1
2.2	Multilayer perceptron (TB3: Section 1.2.2), (TB1: Section 6.1,6.3)	1
2.3	Activation functions- Sigmoid, tanh, Softmax, ReLU, leaky ReLU (TB3: Section 1.2.1.3 - 1.2.1.5)	1
2.4	Architecture design (TB1: Section 6.4, TB3: Section 1.6)	1
2.5	Chain rule, back propagation (TB3: Section 1.3)	1
2.6	Gradient based learning (TB1: Section 6.2)	1
2.7	Gradient based optimization (TB1: Section 4.3)	1
2.8	Linear least squares using a suitable platform. (TB1: Section 4.5)	1
2.9	Building ML Algorithms and Challenges (TB3: 1.4, TB1: 5.10-5.11)	1
Module 3 :Convolution Neural Network (8 hours)		
3.1	Convolution operation (TB1:Section 9.1)	1
3.2	Motivation, pooling (TB1:Section 9.2-9.3)	1

3.3	Convolution and Pooling as an infinitely strong prior (TB1: Section 9.4)	1
3.4	Variants of convolution functions – multilayer convolutional network, tensors, kernel flipping, downsampling, strides and zero padding. (TB1: Section 9.5)	1
3.5	Variants of convolution functions - unshared convolutions, tiled convolution, training different networks. (TB1: Section 9.5)	1
3.6	Structured outputs, data types (TB1: Section 9.6-9.7)	1
3.7	Efficient convolution algorithms. (TB1: Section 9.8,9.10)	1
3.8	Case Study: AlexNet, VGG, ResNet. (TB3: Section 8.4.1, 8.4.3, 8.4.5)	1
Module 4 :Recurrent Neural Network (7 hours)		
4.1	Computational graphs (TB1: Section 10.1)	1
4.2	RNN (TB1: Section 10.2-10.3)	1
4.3	Encoder – decoder sequence to sequence architectures. (TB1: Section 10.4)	1
4.4	Deep recurrent networks (TB1: Section 10.5)	1
4.5	Recursive neural networks , Modern RNNs, LSTM and GRU (TB1: Section 10.6, 10.10)	1
4.6	Practical use cases for RNNs. (TB1: Section 11.1-11.4)	1
4.7	Demonstrate the concepts of RNN using a suitable platform.	1
Module 5 : Applications and Research (5 hours)		
5.1	Computer vision. (TB1: Section 12.2)	1
5.2	Speech recognition. (TB1: Section 12.3)	1
5.3	Natural language processing. (TB1: Section 12.4)	1
5.4	Brief introduction on current research areas- Autoencoders, Representation learning. (TB1: Section 14.1-14.2, TB3: 9.3)	1
5.5	Brief introduction on current research areas- Boltzmann Machines, Deep belief networks. (TB1: Section 20.1, 20.3)	1

CST435	COMPUTER GRAPHICS	CATEGORY	L	T	P	CREDIT	YEAR OF INTRODUCTION
		OEC	2	1	0	3	2019

Preamble: This course helps the learners to make awareness about strong theoretical concept in computer graphics. It covers the three-dimensional environment representation in a computer, transformation of 2D/3D objects and basic mathematical techniques and algorithms used to build applications. This course enables the learners to develop the ability to create image processing frameworks for different domains and develop algorithms for emerging display technologies.

Prerequisite: A sound knowledge of Mathematics and concepts of any programming language.

Course Outcomes: After the completion of the course the student will be able to

CO#	CO
CO1	Describe the working principles of graphics devices**(Cognitive Knowledge level: Understand)**
CO2	Illustrate line drawing, circle drawing and polygon filling algorithms**(Cognitive Knowledge level: Apply)**
CO3	Demonstrate geometric representations and transformations on 2D & 3D objects **(Cognitive Knowledge level: Apply)**
CO4	Demonstrate the working of line and polygon clipping algorithms**(Cognitive Knowledge level: Apply)**
CO5	Summarize visible surface detection methods and illustrate projection algorithms. (**Cognitive Knowledge level: Apply)**

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	✓	✓										✓
CO2	✓	✓	✓	✓	✓							✓
CO3	✓	✓	✓	✓	✓							✓
CO4	✓	✓	✓	✓	✓							✓
CO5	✓	✓										✓

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	**PO7**	Environment and Sustainability
PO2	Problem Analysis	**PO8**	Ethics
PO3	Design/Development of solutions	**PO9**	Individual and team work
PO4	Conduct investigations of complex problems	**PO10**	Communication
PO5	Modern tool usage	**PO11**	Project Management and Finance
PO6	The Engineer and Society	**PO12**	Life long learning

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks (%)
	Test 1 (%)	Test 2 (%)	
Remember	30	30	30
Understand	30	30	30
Apply	40	40	40
Analyze			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3

Continuous Internal Evaluation Pattern:

Attendance **10 marks**

Continuous Assessment Tests (Average of SeriesTests1&2) **25 marks**

Continuous Assessment Assignment **15 marks**

Internal Examination Pattern:

Each of the two internal examinations has to be conducted out of 50 marks. The first series test shall be preferably conducted after completing the first half of the syllabus and the second series test shall be preferably conducted after completing the remaining part of the syllabus. There will be two parts: Part A and Part B. Part A contains 5 questions (preferably, 2 questions each from the completed modules and 1 question from the partly completed module), having 3 marks for each question adding up to 15 marks for part A. Students should answer all questions from Part A. Part B contains 7 questions (preferably, 3 questions each from the completed modules and 1 question from the partly completed module), each with 7 marks. Out of the 7 questions, a student should answer any5.

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 full questions from each module of which student should answer any one full question. Each question can have maximum 2 sub-divisions and carries 14 marks.

Syllabus

Module – 1(Basics of Computer graphics)

Basics of Computer Graphics and its applications. Video Display devices- Refresh Cathode Ray Tubes(CRT), Random Scan Displays and systems, Raster scan displays and systems, Color CRT displays, Flat panel display and its categories.

Module – 2 (Line drawing, Circle drawing and Filled Area Primitives)

Line drawing algorithms- DDA, Bresenham's algorithm. Circle drawing algorithms- Midpoint Circle generation algorithm, Bresenham's algorithm. Filled Area Primitives- Scan line polygon filling, Boundary filling and flood filling.

Module - 3 (Geometric transformations)

Two dimensional transformations-Translation, Rotation, Scaling, Reflection and Shearing, Composite transformations, Matrix representations and homogeneous coordinates. Basic 3D transformations.

Module - 4 (Clipping)

Window to viewport transformation. Cohen Sutherland and Midpoint subdivision line clipping algorithms, Sutherland Hodgeman and Weiler Atherton Polygon clipping algorithms.

Module - 5 (Three dimensional graphics)

Three dimensional viewing pipeline. Projections- Parallel and Perspective projections. Visible surface detection algorithms- Back face detection, Depth buffer algorithm, Scan line algorithm, A buffer algorithm

Text Book

1. Zhigang Xiang and Roy Plastock, Computer Graphics (Schaum's outline Series), McGraw Hill, 2019.
2. Donald Hearn and M. Pauline Baker, Computer Graphics, PHI, 2e, 1996

References

1. William M. Newman and Robert F. Sproull, Principles of Interactive Computer Graphics. McGraw Hill, 2001
2. David F. Rogers , Procedural Elements for Computer Graphics, Tata McGraw Hill,2001.
3. Donald Hearn, M. Pauline Baker and Warren Carithers, Computer Graphics with OpenGL, PHI, 4e, 2013

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Compare the working principle of raster scan systems and random scan systems.
2. How much time is spent scanning across each row of pixels during screen refresh on a raster system with resolution of 1280*1024 and a refresh rate of 60 frames per second?

Course Outcome 2 (CO2):

1. Rasterize the line using Bresenham's line drawing algorithm with end points (2,3) and (5,8) accepted from the user and implement it using any appropriate programming language. (Assignment)
2. Illustrate how the 4-connected boundary filling approach differs from 8-connected boundary filling and implement it using any appropriate programming language. (Assignment)

Course Outcome 3 (CO3):

1. Rotate a triangle ABC 45 degree counter clockwise about the pivot point (10,3) , where the position vector of the coordinates is given as A(4,1), B(5,2) and C(4,3).
2. Implement the above transformation using any appropriate programming language with user inputs. (Assignment)
3. Illustrate the steps required for a general 3D rotation if the rotation axis is not parallel to any one of the principal axis. The rotation axis is defined by the points P1(x1,y1,z1) and P2(x2,y2,z2). Give its composite matrix representation.

Course Outcome 4 (CO4):

1. Given a clipping window A(20,20), B(60,20), C(60,40) and D(20,40). Using Cohen Sutherland algorithm, find the visible portion of the line segment joining the points P(40,80) and Q(120,30).

2. Implement Cohen Sutherland clipping algorithm using any appropriate programming language with user inputs. (Assignment)

Course Outcome 5 (CO5):

1. Explain scan line algorithm for detecting visible surfaces in an object.
2. Derive the matrix for performing perspective projection and parallel projection.

Model Question Paper

QP CODE:

Reg No: _____

Name: _____ **PAGES : 3**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST435

Course Name: Computer Graphics

Max. Marks : 100 **Duration: 3 Hours**

PART A

Answer All Questions. Each Question Carries 3 Marks

1. Describe Flat Panel display and its categories.
2. Consider a raster system with a resolution of 1024*1024. Compute the size of the raster needed to store 4 bits per pixel? How much storage is needed if 8 bits per pixel are to be stored?
3. Justify the usage of integer arithmetic in Bresenham's line drawing algorithm.
4. How 8-way symmetry of circle can be used for developing circle drawing algorithms?
5. Show that two successive reflections about either of the coordinate axes is equivalent to a single rotation about the coordinate origin.
6. Determine a sequence of basic transformations that is equivalent to x-direction shearing.
7. Find the window to viewport normalization transformation with window lower left corner at (1,1) and upper right corner at (2,6).

8. How does Cohen Sutherland algorithm determine whether a line is visible, invisible or a candidate for clipping based on the region codes assigned to the end points of the line?

9. Define the terms (i) Centre of projection (ii) Principal vanishing point

10. Differentiate between the object space and image space method for the hidden surface removal of an image. **(10x3=30)**

Part B

(Answer any one question from each module. Each question carries 14 Marks)

11. (a) Explain the working principle of beam penetration method and shadow mask method with suitable illustrations. **(8)**

(b) Draw the architecture of raster scan display systems and explain its working principle. **(6)**

OR

12. (a) Explain the working principle of a Refresh CRT monitor with suitable diagrams. **(8)**

(b) Describe random graphics system with suitable illustrations. **(6)**

13. (a) Differentiate between boundary fill and flood fill algorithms. **(5)**

(b) Derive the initial decision parameter of Bresenham's line drawing algorithm and rasterize a line with endpoints (2,2) and (10,10). **(9)**

OR

14. (a) Write Midpoint circle drawing algorithm and identify the points in the circle with radius as 20 and center at (50,30) using the algorithm. **(8)**

(b) Illustrate the working principle of scan line polygon filling algorithm. **(6)**

15. (a) Reflect a triangle ABC about the line 3x-4y+8=0, where the coordinates of the triangle are given as A(4,1), B(5,2) and C(4,3). **(8)**

(b) A diamond shaped polygon is located at P(-1,0), Q(0,-2), R(1,0) and S(0,2). Find the transformation matrix which would rotate the triangle by 90 degree counter clockwise about the point Q. Using the transformation matrix, find the coordinates of the rotated polygon**.** **(6)**

OR

16. (a) Describe the steps required for a general 3D rotation if the rotation axis is not parallel to any one of the principal axis. The rotation axis is defined by the points P1(x1,y1,z1) and P2(x2,y2,z2). Give its composite matrix representation. **(8)**

(b) Consider a triangle at (2,2), (10,2), (2,10). Perform the following 2D transformations in succession and find the resultant vertices. **(6)**

i) Scale with respect to (2,2) by scaling factors (2,2) along x and y directions.

ii) Rotate by 90 degree counter clockwise direction.

iii) Reflection based on y=x

17. (a) Illustrate Weiler – Atherton polygon clipping algorithm. **(6)**

(b) Explain Cohen-Sutherland line clipping algorithm. Use the algorithm to clip line with end points P1 (70, 20) and P2(100,10) against a window with lower left hand corner (50,10) and upper right hand corner (80,40). **(8)**

OR

18. (a) Describe Sutherland Hodgeman polygon clipping algorithm and list out its limitations. **(7)**

(b) Explain the steps involved in clipping a line using Mid point Subdivision algorithm. **(7)**

19. (a) Explain how visible surfaces can be detected using depth buffer algorithm. **(7)**

(b) Define parallel projection. Describe orthographic and oblique parallel projection. **(7)**

OR

20. (a) Illustrate the scan line method used in visible surface detection. **(7)**

(b) Derive the matrix needed for performing perspective projections. **(7)**

TEACHING PLAN

No	Contents	No of Lecture Hrs (35 hrs)
Module – 1 (Basics of Computer Graphics) (6 hrs)		
1.1	Basics of Computer Graphics and applications	1
1.2	Refresh Cathode Ray Tubes	1
1.3	Random Scan Displays and systems	1
1.4	Raster scan displays and systems	1
1.5	Color CRT displays	1
1.6	Flat panel display and its categories.	1
Module - 2 (Line drawing, Circle drawing and Filled Area Primitives) (7 hrs)		
2.1	DDA Line drawing Algorithm	1
2.2	Bresenham's line drawing algorithm	1
2.3	Midpoint Circle generation algorithm	1
2.4	Bresenham's Circle generation algorithm	1
2.5	Illustration of line drawing and circle drawing algorithms	1
2.6	Scan line polygon filling	1
2.7	Boundary filling and flood filling	1
Module - 3 (Geometric transformations) (8 hrs)		
3.1	Basic 2D transformations-Translation and Rotation	1
3.2	Basic 2D transformations- Scaling	1
3.3	Reflection and Shearing	1
3.4	Illustration of 2D Transformations	1
3.5	Composite transformations	1
3.6	Matrix representations and homogeneous coordinates	1
3.7	Basic 3D transformations	1
3.8	Illustration of basic 3D transformations	1
Module - 4 (2D Clipping) (6 hrs)		
4.1	Window to viewport transformation	1
4.2	Cohen Sutherland Line clipping algorithm	1
4.3	Midpoint subdivision Line clipping algorithm	1
4.4	Sutherland Hodgeman Polygon clipping algorithm	1
4.5	Weiler Atherton Polygon clipping algorithm	1
4.6	Practice problems on Clipping algorithms	1
Module - 5 (Three dimensional graphics)(8 hrs)		
5.1	Three dimensional viewing pipeline, Projections-Parallel projections	1

5.2	Projections- Perspective projections	1
5.3	Visible surface detection algorithms- Back face detection.	1
5.4	Depth buffer algorithm	1
5.5	Depth buffer algorithm	1
5.6	Scan line visible surface detection algorithm	1
5.7	Scan line visible surface detection algorithm	1
5.8	A buffer algorithm	1

CST445	PYTHON FOR ENGINEERS	CATEGORY	L	T	P	CREDIT	YEAR OF INTRODUCTION
		OEC	2	1	0	3	2019

Preamble: The objective of the course is to provide learners an insight into Python programming in a scientific computation context and develop programming skills to solve engineering problems. It covers programming environment, important instructions, data representations, intermediate level features, Object Oriented Programming and file data processing of Python. This course lays the foundation to scientific computing, develop web applications, Machine Learning, and Artificial Intelligence-based applications and tools, Data Science and Data Visualization applications.

Prerequisite: NIL

Note : ***Students who have successfully completed CST 283 - Python for Machine Learning (Minor) are not eligible to opt this course.***

Course Outcomes: After the completion of the course the student will be able to

CO1	Write, test and debug Python programs **(Cognitive Knowledge level: Apply)**
CO2	Illustrate uses of conditional (if, if-else, if-elif-else and switch-case) and iterative (while and for) statements in Python programs **(Cognitive Knowledge level: Apply)**
CO3	Develop programs by utilizing the modules Lists, Tuples, Sets and Dictionaries in Python **(Cognitive Knowledge level: Apply)**
CO4	Implement Object Oriented programs with exception handling **(Cognitive Knowledge level: Apply)**
CO5	Analyze, Interpret, and Visualize data according to the target application (**Cognitive Knowledge level: Apply**)
CO6	Develop programs in Python to process data stored in files by utilizing the modules Numpy, Matplotlib, and Pandas **(Cognitive Knowledge level: Apply)**

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	✓	✓										✓
CO2	✓	✓										✓
CO3	✓	✓										✓
CO4	✓	✓	✓		✓							✓
CO5	✓	✓	✓		✓							✓
CO6	✓	✓	✓		✓							✓

Abstract POs defined by National Board of Accreditation

#PO	Broad PO	#PO	Broad PO
PO1	Engineering Knowledge	PO7	Environment and Sustainability
PO2	Problem Analysis	PO8	Ethics
PO3	Design/Development of solutions	PO9	Individual and team work
PO4	Conduct investigations of complex problems	PO10	Communication
PO5	Modern tool usage	PO11	Project Management and Finance
PO6	The Engineer and Society	PO12	Life long learning

Assessment Pattern

Bloom's Category	Test 1 (*Marks in percentage*)	Test 2 (*Marks in percentage*)	End Semester Examination Marks
Remember	20	20	20
Understand	30	30	30
Apply	50	50	50
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3

Continuous Internal Evaluation Pattern:

Attendance	: 10 marks
Continuous Assessment Test	: 25 marks
Continuous Assessment Assignment	: 15 marks

Internal Examination Pattern:

Each of the two internal examinations has to be conducted out of 50 marks. The first series test shall be preferably conducted after completing the first half of the syllabus and the second series test shall be preferably conducted after completing the remaining part of the syllabus. There will be two parts: Part A and Part B. Part A contains 5 questions (preferably, 2 questions each from the completed modules and 1 question from the partly completed module), having 3 marks for each question adding up to 15 marks for part A. Students should answer all questions from Part A. Part B contains 7 questions (preferably, 3 questions each from the completed modules and 1 question from the partly completed module), each with 7 marks. Out of the 7 questions, a student should answer any 5.

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which a student should answer any one. Each question can have a maximum of 2 sub-divisions and carries 14 marks.

Syllabus

Module 1 (Basics of Python)

Getting Started with Python Programming - Running code in the interactive shell, Editing, Saving, and Running a script. Using editors - IDLE, Jupyter. Basic coding skills - Working with data types, Numeric data types and Character sets, Keywords, Variables and Assignment statement, Operators, Expressions, Working with numeric data, Type conversions, Comments in the program, Input Processing, and Output, Formatting output. How Python works. Detecting and correcting syntax errors. Using built in functions and modules in math module. Control statements - Selection structure - if-else, if-elif-else. Iteration structure - for, while. Testing the control statements. Lazy evaluation.

Module 2 (Functions and Python Data Structures)

Functions - Hiding redundancy and complexity, Arguments and return values, Variable scopes and parameter passing, Named arguments, Main function, Working with recursion, Lambda functions. Strings - String function. Lists - Basic list Operations and functions, List of lists, Slicing, Searching and sorting list, List comprehension. Work with tuples. Sets. Dictionaries - Dictionary functions, dictionary literals, adding and removing keys, accessing and replacing values, traversing dictionaries, reverse lookup.

Module 3 (Object Oriented Programming)

Design with classes - Objects and Classes, Methods, Instance Variables, Constructor, Accessors and Mutators. Structuring classes with Inheritance and Polymorphism. Abstract Classes. Exceptions - Handle a single exception, Handle multiple exceptions.

Module 4 (Visualization and File handling)

Plotting - An Interactive Session with PyPlot, Basic Plotting, Logarithmic Plots, More Advanced Graphical Output, Plots with multiple axes, Mathematics and Greek symbols, The Structure of matplotlib, Contour and Vector Field Plots. File Processing - The os and sys modules, Introduction to file I/O, Reading and writing text files, Working with CSV files.

Module 5 (Scientific Computing)

Numerical Routines. SciPy and NumPy - Basics, Creating arrays, Arithmetic, Slicing, Matrix Operations, Special Functions, Random Numbers, Linear Algebra, Solving Nonlinear Equations, Numerical Integration, Solving ODEs. Data Manipulation and Analysis – Pandas : Reading Data from Files Using Pandas, Data Structures: Series and DataFrame, Extracting Information from a DataFrame, Grouping and Aggregation.

Text Books:

1. Kenneth A Lambert., Fundamentals of Python : First Programs, 2/e, Cengage Publishing, 2016
2. David J. Pine, Introduction to Python for Science and Engineering, CRC Press, 2021

Reference Books:

1. Wes McKinney, Python for Data Analysis, 2/e, Shroff / O'Reilly Publishers, 2017
2. Allen B. Downey, Think Python: How to Think Like a Computer Scientist, 2/e, Schroff, 2016
3. Michael Urban and Joel Murach, Python Programming, Shroff/Murach, 2016
4. David M.Baezly, Python Essential Reference. Addison-Wesley Professional; 4/e, 2009.
5. Charles Severance. Python for Informatics: Exploring Information,
6. http://swcarpentry.github.io/python-novice-gapminder/

Sample Course Level Assessment Questions

Course Outcome1(CO1):

1. What is type conversion? How is it done in Python?

Course Outcome 2(CO2):

1. Given is a list of of words, *wordlist*, and a string, *name.* Write a Python function which takes *wordlist* and *name* as input and returns a tuple. The first element of the output tuple is the number of words in the *wordlist* which have *name* as a substring in it. The second element of the tuple is a list showing the index at which the *name* occurs in each of the words of the *wordlist* and a 0 if it doesn't occur.

Course Outcome 3(CO3):

1. Write a Python program to implement the addition, subtraction, and multiplication of complex numbers using classes. Use constructors to create objects. The input to the program consist of real and imaginary parts of the complex numbers.

Course Outcome 4(CO4):

1. Plot the function $y = 3x^2$ for $-1 \leq x \leq 3$ as a continuous line. Include enough points so that the curve you plot appears smooth. Label the axes x and y

Course Outcome 5(CO5):

1. Given a file "auto.csv" of automobile data with the fields *index, company, body-style, wheel-base, length, engine-type, num-of-cylinders, horsepower, average-mileage,* and *price***,** write python code to
   i. Clean and Update the CSV file
   ii. Print total cars of all companies
   iii. Find the average mileage of all companies
   iv. Find the highest priced car of all companies.

Model Question Paper

QP CODE: PAGES:

Reg No:_____
Name:_____

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR
Course Code: CST445

Course name : PYTHON FOR ENGINEERS

Max Marks: 100 **Duration: 3 Hours**

PART-A

(Answer All Questions. Each question carries 3 marks)

1. Explain the basic data types available in Python, with examples.
2. Write a Python program to reverse a number and also find the sum of digits of the number. Prompt the user for input.
3. Compare tuples, lists, and dictionaries.
4. Explain the concept of scope and lifetime of variables in Python programming language, with a suitable example.
5. What is polymorphism? Give an example in the context of OOP in Python.
6. How is exception handling accomplished in Python programs?
7. Describe the characteristics of the CSV format.

8. Plot the function $y = 3x^2$ for $-1 \leq x \leq 3$ as a continuous line. Include enough points so that the curve you plot appears smooth. Label the axes x and y

9. Describe random number generation using Python

10. How can a generalized eigen value problem can be solved using Python?

PART-B
(Answer any one full question from each module)

Module -1

11. (a) Compare and contrast interpreted languages and compiled languages. How does it affect the quality of program development and execution of the program? (6)

(b) What are the possible errors in a Python program. Write a Python program to print the value of $2^{2n}+n+5$ for n provided by the user. (8)

OR

12. (a) Describe Arithmetic operators, Assignment operators, Comparison operators, Logical operators, and Bitwise operators in detail with examples. (6)

(b) Input 4 integers (+ve and −ve). Write a Python code to find the sum of negative numbers, positive numbers, and print them. Also, find the averages of these two groups of numbers and print (8)

Module -2

13. (a) Write a Python code to create a function called *list_of_frequency* that takes a string and prints the letters in non-increasing order of the frequency of their occurrences. Use dictionaries. (5)

(b) Write a Python program to read a list of numbers and sort the list in a non-decreasing order without using any built in functions. Separate function should be written to sort the list wherein the name of the list is passed as the parameter. (9)

OR

14. (a) Illustrate the following Set methods with an example. (8)

i. *intersection( )* ii. *Union( )* iii. *Issubset( )* iv. *Difference( )* v. *update( )* vi. *discard( )*

(b) Write a Python program to check the validity of a password given by the user. (6)

The Password should satisfy the following criteria:

1. Contains at least one letter between a and z
2. Contains at least one number between 0 and 9
3. Contains at least one letter between A and Z

4. Contains at least one special character from $, #, @

5. Minimum length of password: 6

Module -3

15. (a) How can a class be instantiated in Python? Write a Python program to express the instances as return values to define a class RECTANGLE with parameters *height, width, corner_x,* and *corner_y* and member functions to find center, area, and perimeter of an instance. (5)

(b) Explain inheritance in Python. Give examples for each type of inheritance. (9)

OR

16. (a) Write a Python class named Circle constructed by a radius and two methods which will compute the area and the perimeter of a given circle (6)

(b) Define a class in Python to store the details of a ship (name, source,destination) with the following methods: (8)

i) *get_details( )* - to assign values to class attributes

ii) *print_details( )* - to display the attribute values

Create an object of the class and invoke the methods

Module -4

17. (a) Plot the functions *sin x* and *cos x* vs *x* on the same plot with x going from $-\pi$ to π. Make sure the limits of the *x*-axis do not extend beyond the limits of the data. Plot *sin x* in the color orange and *cos x* in the color green and include a legend to label the two curves. Place the legend within the plot, but such that it does not cover either of the sine or cosine traces. Draw thin gray lines behind the curves, one horizontal at $y = 0$ and the other vertical at $x = 0$. (10)

(b) Explain semi-log plots and log-log plots along with the functions used in creating such plots. (4)

OR

18. (a) Explain how *matplotlib* can be used to create dimensional contour plots and vector field plots. (6)

(b) Given a file "*auto.csv*" of automobile data with the fields *index, company, body-style, wheel-base, length, engine-type, num-of-cylinders, horsepower, average-mileage,* and *price,* write Python codes using Pandas to (8)

1) Clean and Update the CSV file
2) Print total cars of all companies
3) Find the average mileage of all companies
4) Find the highest priced car of all companies.

Module -5

19. (a) Write python program to solve the following system of equations (4)

$x_1 - 2x_2 + 9x_3 + 13x_4 = 1$

$-5x_1 + x_2 + 6x_3 - 7x_4 = -3$

$4x_1 + 8x_2 - 4x_3 - 2x_4 = -2$

$8x_1 + 5x_2 - 7x_3 + x_4 = 5$

(b) Given the sales information of a company as CSV file with the following fields *month_number, facecream, facewash, toothpaste, bathingsoap, shampoo, moisturizer, total_units, total_profit*. Write Python codes to visualize the data as follows (10)

1) Toothpaste sales data of each month and show it using a scatter plot
2) Face cream and face wash product sales data and show it using the bar chart

Calculate total sale data for last year for each product and show it using a Pie chart.

OR

20. (a) Write Python program to write the data given below to a CSV file. (9)

SN	Name	Country	Contribution	Year
1	Linus Torvalds	Finland	Linux Kernel	1991
2	Tim Berners-Lee	England	World Wide Web	1990
3	Guido van Rossum	Netherlands	Python	1991

(b) Explain how integration is performed with SciPy. Illustrate the same with the two sample integrals using SciPy function. (5)

Teaching Plan

Sl No	Contents	Number of Hours (35 Hrs)
Module 1: Basics of Python (8 hours)		
1.1	Getting Started with Python Programming: Running code in the interactive shell Editing, Saving, and Running a script	1 hour
1.2	Using editors: IDLE, Jupyter	1 hour
1.3	Basic coding skills: Working with data types, Numeric data types and Character sets, Keywords, Variables and Assignment statement, Operators, Expressions,	1 hour
1.4	Working with numeric data, Type conversions, Comments in the program, Input Processing, and Output. Formatting output	1 hour
1.5	How Python works. Detecting and correcting syntax errors. Using built in functions and modules in math module.	1 hour
1.6	Control statements : Selection structure, if-else, if ~~elif~~else	1 hour
1.7	Iteration structure - for, while	1 hour
1.8	Testing the control statements, Lazy evaluation.	1 hour
Module 2: Functions and Python Data Structures (8 hours)		
2.1	Functions: Hiding redundancy and complexity, Arguments and return values	1 hour
2.2	Variable scopes and parameter passing	1 hour
2.3	Named arguments, Main function,	1 hour
2.4	Working with recursion, Lambda functions	1 hour
2.5	Strings - String function	1 hour
2.6	Lists - Basic list Operations and functions, List of lists, Slicing, Searching and sorting list, List comprehension.	1 hour
2.7	Work with tuples. Sets.	1 hour
2.8	Dictionaries - Dictionary functions, dictionary literals, adding and removing keys, Accessing and replacing values, traversing dictionaries, reverse lookup	1 hour
Module 3: Object Oriented Programming (6 hours)		
3.1	Design with classes : Objects and Classes, Methods, Instance Variables	1 hour
3.2	Constructor, Accessors, and Mutators	1 hour
3.3	Structuring classes with Inheritance	1 hour
3.4	Polymorphism	1 hour
3.5	Abstract Classes	1 hour
3.6	Exceptions: Handle a single exception, Handle multiple exception	1 hour
Module 4: Visualization and File handling (6 hours)		

4.1	Plotting - An Interactive Session with PyPlot, Basic Plotting,	1 hour
4.2	Logarithmic Plots, More Advanced Graphical Output	1 hour
4.3	Plots with multiple axes, Mathematics and Greek symbols	1 hour
4.4	The Structure of matplotlib, Contour and Vector Field Plots	1 hour
4.5	File Processing -The *os* and *sys* modules, Introduction to file I/O, Reading and writing text files	1 hour
4.6	Working with CSV files	1 hour
Module 5: Scientific Computing (7 hours)		
5.1	Numerical Routines: SciPy and NumPy - Basics, Creating arrays, Arithmetic, Slicing	1 hour
5.2	Matrix Operations, Special Functions, Random Numbers	1 hour
5.3	Linear Algebra, Solving Nonlinear Equations	1 hour
5.4	Numerical Integration, Solving ODEs	1 hour
5.5	Data Manipulation and Analysis: Pandas - Reading Data from Files Using Pandas	1 hour
5.6	Data Structures - Series and DataFrame	1 hour
5.7	Extracting Information from a DataFrame, Grouping and Aggregation	1 hour

CST455	OBJECT ORIENTED CONCEPTS	CATEGORY	L	T	P	CREDIT	YEAR OF INTRODUCTION
		OEC	2	1	0	3	2019

Preamble: The purpose of this course is to enable learners to solve problems by breaking it down to object level while designing software and to implement it using Java. This course covers Object Oriented Principles, Object Oriented Programming in Java, Exception handling, Event handling, multithreaded programming and working with window-based graphics. This course provides learners the basics to develop Mobile applications, Enterprise Applications, Scientific Applications and Web based Applications.

Prerequisite: A sound background in any of the programming languages like C, C++, Python etc is mandatory. Students who completed the minor stream course CST 281 Object Oriented Programming are not allowed to choose this Open Elective Course.

Course Outcomes: After the completion of the course the student will be able to

CO1	Develop Java programs using the object-oriented concepts - classes, objects, constructors, data hiding, inheritance and polymorphism **(Cognitive Knowledge Level: Apply)**
CO2	Utilise data types, operators, control statements, built in packages & interfaces, Input/Output Streams and Files in Java to develop programs **(Cognitive Knowledge Level: Apply)**
CO3	Illustrate how robust programs can be written in Java using exception handling mechanism **(Cognitive Knowledge Level: Apply)**
CO4	Develop application programs in Java using multithreading **(Cognitive Knowledge Level: Apply)**
CO5	Develop Graphical User Interface based application programs by utilising event handling features and Swing in Java **(Cognitive Knowledge Level: Apply)**

Mapping of course outcomes with program outcomes

	PO 1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO1 0	PO11	PO1 2
CO1	✓	✓	✓									✓
CO2	✓	✓	✓									✓
CO3	✓	✓	✓									✓
CO4	✓	✓	✓									✓
CO5	✓	✓	✓									✓

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	**PO7**	Environment and Sustainability
PO2	Problem Analysis	**PO8**	Ethics
PO3	Design/Development of solutions	**PO9**	Individual and team work
PO4	Conduct investigations of complex problems	**PO10**	Communication
PO5	Modern tool usage	**PO11**	Project Management and Finance
PO6	The Engineer and Society	**PO12**	Life long learning

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks (%)
	Test 1 (%)	Test 2 (%)	
Remember	20	20	20
Understand	40	40	40
Apply	40	40	40
Analyze			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3

Continuous Internal Evaluation Pattern:

Attendance	**10 marks**
Continuous Assessment Tests(Average of Internal Tests1&2)	**25 marks**
Continuous Assessment Assignment	**15 marks**

Internal Examination Pattern

Each of the two internal examinations has to be conducted out of 50 marks. First series test shall be preferably conducted after completing the first half of the syllabus and the second series test shall be preferably conducted after completing remaining part of the syllabus. There will be two parts: Part A and Part B. Part A contains 5 questions (preferably, 2 questions each from the completed modules and 1 question from the partly completed module), having 3 marks for each question adding up to 15 marks for part A. Students should answer all questions from Part A. Part B contains 7 questions (preferably, 3 questions each from the completed modules and 1 question

from the partly completed module), each with 7 marks. Out of the 7 questions, a student should answer any5.

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 full questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carries 14 marks.

Syllabus

Module – 1 (Object Orientation and Java basics)

Object Orientation Principles – Object and Class, Data abstraction and Encapsulation, Inheritance, Polymorphism, Dynamic binding, Message communication, Benefits of using Object orientation.

Introduction to Java - Java programming Environment and Runtime Environment, Development Platforms -Standard, Enterprise. Java Virtual Machine (JVM), Java compiler, Bytecode, Java applet, Java Buzzwords, Java program structure, Comments, Garbage Collection, Lexical Issues.

Primitive Data types - Integers, Floating Point Types, Characters, Boolean. Literals, Type Conversion and Casting, Variables, Arrays, Strings, Vector class.

Module – 2 (Core Java Fundamentals)

Operators - Arithmetic Operators, Bitwise Operators, Relational Operators, Boolean Logical Operators, Assignment Operator, Conditional (Ternary) Operator, Operator Precedence.

Control Statements - Selection Statements, Iteration Statements and Jump Statements.

Object Oriented Programming in Java - Class Fundamentals, Declaring Objects, Object Reference, Introduction to Methods, Constructors, ***this*** Keyword, Method Overloading, Using Objects as Parameters, Returning Objects, Recursion, Access Control, Static Members, Command-Line Arguments, Variable Length Arguments.

Module - 3 (More features of Java)

Inheritance - Super Class, Sub Class, The Keyword ***super***, protected Members, Calling Order of Constructors, Method Overriding, the Object class, Abstract Classes and Methods, Using ***final*** with Inheritance.

Packages and Interfaces - Defining Package, CLASSPATH, Access Protection, Importing Packages, Interfaces.

Exception Handling - Checked Exceptions, Unchecked Exceptions, ***try*** Block and ***catch*** Clause, Multiple ***catch*** Clauses, Nested ***try*** Statements, ***throw***, ***throws*** and ***finally***.

Module - 4 (Advanced features of Java)

Input/Output - I/O Basics, Reading Console Input, Writing Console Output, PrintWriter Class, Reading and Writing Files.

Java Library - String Handling – String Constructors, String Length, Special String Operations - Character Extraction, String Comparison, Searching Strings, Modifying Strings, Using valueOf(), Comparison of String Buffer and String.

Module - 5 (GUI Programming, Event Handling and Multithreaded Programming)

Multithreaded Programming - The Java Thread Model, The Main Thread, Creating Thread, Creating Multiple Threads, Suspending, Resuming and Stopping Threads.

Event Handling - Event Handling Mechanisms, Delegation Event Model, Event Classes, Sources of Events, Event Listener Interfaces, Using the Delegation Model.

Swing Fundamentals - Swing Key Features, Model View Controller (MVC), Swing Controls, Components and Containers, Exploring Swing - JFrame, JLabel, JButton, JTextField.

Text Books

1. Herbert Schildt, Java: The Complete Reference, 8/e, Tata McGraw Hill, 2011.
2. Balagurusamy E., Programming JAVA a Primer, 5/e, McGraw Hill, 2014.

Reference Books

1. Paul Deitel, Harvey Deitel, Java How to Program, Early Objects 11/e, Pearson, 2018.
2. Y. Daniel Liang, Introduction to Java Programming, 7/e, Pearson, 2013.
3. Nageswararao R., Core Java: An Integrated Approach, Dreamtech Press, 2008.
4. Flanagan D., Java in A Nutshell, 5/e, O'Reilly, 2005.
5. Sierra K., Head First Java, 2/e, O'Reilly, 2005.

Course Level Assessment Questions

Course Outcome1 (CO1):

1. Three types of employees work in an organization: Regular, Contract and Hourly. Regular employees are permanent workers of the organization. Their salary is computed as the sum of basic pay, DA (50% of basic pay) and HRA. Contract employees work for the organization only for the contract period and earn a fixed salary. Hourly employees work for a fixed number of hours each day. Their salary is computed based on the total number of hours worked.

   Using object oriented principles, write a Java program to prepare pay roll of the organization.

2. Write a java program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Square, Triangle and Circle with proper class hierarchy. Each one of the classes contain only the method printArea( ) that prints the area of the given shape.

Course Outcome 2(CO2):

1. Write a Java program that reads a file and displays the file on the screen, with a line number before each line.
2. Write a Java program to prepare the rank list of computer science students based on their performance in the first Semester B.Tech. Degree examination at APJ Abdul Kalam Technological University. The output should be stored in a file.

Course Outcome 3(CO3):

1. Write a program to demonstrate the use of *throws* clause to handle an exception occurred within a method.
2. Write a program to demonstrate how exception handling is supported in Java.

Course Outcome 4(CO4):

1. Write a program to compute the sum of elements in an array using two threads in a parallel way. The first thread sums up the first half of the array and the second thread sums up the second half of the array. Finally, the main thread adds these partial sums and prints the result.
2. Write a java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and ifthe value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.

Course Outcome 5(CO5):

1. Write a GUI based program to convert temperature from degree Celsius to Fahrenheit.
2. Write a java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with buttons. On selecting a button, an appropriate message with "stop" or "ready" or "go" should appear above the buttons in a selected color. Initially there is no message shown.

Model Question Paper

QP CODE:

Reg No: _____

Name: _____ **PAGES :4**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST455

Course Name: Object Oriented Concepts

Max.Marks:100 **Duration: 3 Hours**

PART A

Answer All Questions. Each Question Carries 3 Marks

1. Java is considered to be secure and portable. Justify this statement.
2. Describe the concept of dynamic binding.
3. Explain the different arithmetic operators in Java.
4. What does the following Java function compute? Justify your answer.

```
intgreater(int a, int b)
{
while(a!=b)
{
```

```
    if(a>b)
a=a-b;
    else
b=b-a;
}
return a;
}
```

5. Explain the use of CLASSPATH with an example.

6. What are the different types of exceptions?

7. Explain file handling features available in Java.

8. Write a simple program to read an integer value from console and print it.

9. Explain the concept of *main thread* in multi-threading.

10. Explain any two Event classes in Java. **(10x3=30)**

Part B

(Answer any one question from each module. Each question carries 14 Marks)

11. (a) Describe in detail polymorphism, abstraction and inheritance with suitable examples. **(9)**

    (b) What is Java Virtual Machine? **(5)**

OR

12. (a) Explain the salient features of Java language. How does Java Enterprise Edition (J2EE) differ from Java Standard Edition (Java SE)? **(9)**

    (b) Explain the declaration and use of multi-dimensional array variables in Java, with example. **(5)**

13. (a) Explain iteration control statements in Java. Give examples. **(8)**

(b) Write a recursive program to compute the factorial of a number. **(6)**

OR

14. (a) Using a suitable Java program, explain the concept of methods and constructors. **(6)**

(b) Write a Java program that prompts the user for an integer and then prints out all the prime numbers up to that number. **(8)**

15. (a) In a table format, show the effect of access specifiers within and outside packages in Java. **(6)**

(b) Describe exception handling using **try** block and **catch** clause in Java with the help of a suitable Java program. **(8)**

OR

16. (a) What is an interface in Java? Explain with a suitable example. **(6)**

(b) Write a program that perform integer divisions. The user enters two input data (any data type) through console into variables Num1 and Num2. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the appropriate exception or result. **(8)**

17. (a) Write a Java program that displays the number of characters, lines and words in a text file. **(8)**

(b) Explain any three String constructors with the help of sample code for each. **(6)**

OR

18. (a) Write a program to demonstrate the usage of the *PrintWriter* class. **(7)**

(b) Write a Java program for sorting a given list of names in ascending order. **(7)**

19. (a) Explain Delegation Event model for event handling in Java. **(7)**

(b) Write a program to compute the sum of elements in an array using two **(7)**

threads in a parallel way. The first thread sums up the first half of the array and the second thread sums up the second half of the array. Finally, the main thread adds these partial sums and prints the result.Use Runnable interfacefor the creation of a thread.

OR

20. (a) What are the differences between a process and a thread? **(4)**

(b) Write a Graphical User Interface (GUI) based Java program to implement a simple calculator supporting the operations addition, subtraction, multiplication and division. Use Swing controls to implement GUI. There may be three text boxes, the first two for accepting the operands and the last for displaying the result. Add four buttons for the above operations. Write neat comments in your program to show how you handle events. **(10)**

Teaching Plan

No	Contents	No. of Lecture Hours (36hrs)
	Module – 1 (Object Orientation and Java basics) (7 hrs)	
1.1	Object Orientation Principles – Object and Class, Data abstraction and Encapsulation	1 hour
1.2	Inheritance, Polymorphism	1 hour
1.3	Dynamic binding, Message communication, Benefits of using Object orientation.	1 hour
1.4	Java programming Environment and Runtime Environment, Development Platforms -Standard, Enterprise. JVM, Java compiler, Bytecode	1 hour
1.5	Java applet, Java Buzzwords, Java program structure, Comments, Garbage Collection, Lexical Issues	1 hour
1.6	Primitive Data types - Integers, Floating Point Types, Characters, Boolean	1 hour
1.7	Literals, Type Conversion and Casting, Variables, Arrays, Strings, Vector	1 hour

	class.	
Module - 2 (Core Java Fundamentals) (7 hrs)		
2.1	Operators - Arithmetic Operators, Bitwise Operators, Relational Operators, Boolean Logical Operators, Assignment Operator, Conditional (Ternary) Operator, Operator Precedence.	1 hour
2.2	Control Statements - Selection Statements, Iteration Statements and Jump Statements.	1 hour
2.3	Object Oriented Programming in Java - Class Fundamentals, Declaring Objects	1 hour
2.4	Object Reference, Introduction to Methods, Constructors, ***this*** Keyword	1 hour
2.5	Method Overloading, Using Objects as Parameters, Returning Objects	1 hour
2.6	Recursion, Access Control, static Members	1 hour
2.7	Command-Line Arguments, Variable Length Arguments	1 hour
Module - 3 (More features of Java) (8 hrs)		
3.1	Inheritance - Super class, Sub class, the keyword super, protected Members	1 hour
3.2	Calling Order of Constructors, Method Overriding, the Object class	1 hour
3.3	Abstract Classes and Methods, Using final with Inheritance	1 hour
3.4	Packages and Interfaces - Defining Package, CLASSPATH, Access Protection	1 hour
3.5	Importing Packages, Interfaces	1 hour
3.6	Exception Handling - Checked Exceptions, Unchecked Exceptions, ***try*** Block and ***catch*** Clause	1 hour
3.7	Multiple ***catch*** Clauses, Nested ***try*** Statements	1 hour
3.8	***throw***, ***throws*** and ***finally***	1 hour
Module - 4 (Advanced features of Java) (6 hrs)		
4.1	Input/Output - I/O Basics, Reading Console Input	1 hour
4.2	Writing Console Output, PrintWriter Class	1 hour
4.3	Working with Files (Lecture-1)	1 hour

4.4	Working with Files (Lecture-2)	1 hour
4.5	Java Library - String Handling – String Constructors, String Length	1 hour
4.6	Special String Operations - Character Extraction, String Comparison, Searching Strings, Modifying Strings, Using valueOf( ), Comparison of StringBuffer and String.	1 hour
Module - 5 (GUI Programming, Event Handling and Multithreaded Programming) (8hrs)		
5.1	Multithreaded Programming - The Java Thread Model, The Main Thread, Creating Thread	1 hour
5.2	Creating Multiple Threads	1 hour
5.3	Suspending, Resuming and Stopping Threads.	1 hour
5.4	Event handling - Event Handling Mechanisms, Delegation Event Model	1 hour
5.5	Event Classes,Sources of Events, Event Listener Interfaces	1 hour
5.6	Using the Delegation Model, Swing fundamentals, Swing Key Features	1 hour
5.7	Model View Controller (MVC), Swing Controls, Components and Containers	1 hour
5.8	Exploring Swing –JFrame, JLabel, JButton, JTextField	1 hour

SEMESTER VII

MINOR

CSD481	MINI PROJECT	CATEGORY	L	T	P	CREDIT	YEAR OF INTRODUCTION
		PWS	0	0	3	4	2019

Preamble: The objective of this course is to apply the fundamental concepts of different courses learned in respective Minor Streams: Software Engineering, Machine Learning and Networking. This course helps the learners to get an exposure to the development of application software/hardware solutions/ software simulations in the field of Computer Science and Engineering. It enables the learners to understand the different steps to be followed such as literature review and problem identification, preparation of requirement specification &design document, testing, development and deployment. Mini project enables the students to boost their skills, widen the horizon of thinking and their ability to resolve real life problems.

Prerequisite:

A sound knowledge in courses studied in respective minor stream.

Course Outcomes: After the completion of the course the student will be able to

CO#	CO
CO1	Identify technically and economically feasible problems **(Cognitive Knowledge Level: Apply)**
CO2	Identify and survey the relevant literature for getting exposed to related solutions. **(Cognitive Knowledge Level: Apply)**
CO3	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions of minimal complexity by using modern tools & advanced programming techniques (**Cognitive Knowledge Level: Apply)**
CO4	Prepare technical report and deliver presentation **(Cognitive Knowledge Level: Apply)**
CO5	Apply engineering and management principles to achieve the goal of the project **(Cognitive Knowledge Level: Apply)**

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓
CO2	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓
CO5	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	**PO7**	Environment and Sustainability
PO2	Problem Analysis	**PO8**	Ethics
PO3	Design/Development of solutions	**PO9**	Individual and team work
PO4	Conduct investigations of complex problems	**PO10**	Communication
PO5	Modern tool usage	**PO11**	Project Management and Finance
PO6	The Engineer and Society	**PO12**	Lifelong learning

Assessment Pattern

Mark Distribution

Total Marks	CIE Marks	ESE Marks
150	**75**	**75**

Continuous Internal Evaluation Pattern:

Attendance **10 marks**

Project Guide **15 marks**

Project Report **10 marks**

Evaluation by the Committee (will be evaluating the level of completion and demonstration of functionality/specifications, presentation, oral examination, work knowledge and involvement) : **40 marks**

Student Groups with 4 or 5 members should identify a topic of interest in consultation with a Faculty Advisor/Project Coordinator/Guide. Review the literature and gather information pertaining to the chosen topic. State the objectives and develop a methodology to achieve the objectives. Carryout the design/fabrication or develop codes/programs to achieve the objectives by strictly following steps specified in the teaching plan. Innovative design concepts, performance, scalability, reliability considerations, aesthetics/ergonomic, user experience and security aspects taken care of in the project shall be given due weight.

The progress of the mini project is evaluated based on a minimum of two reviews. The review committee may be constituted by a senior faculty member, Mini Project coordinator and project guide. The internal evaluation shall be made based on the progress/outcome of the project, reports and a viva-voce examination, conducted internally by a 3-member committee. A project report is required at the end of the semester. The project has to be demonstrated for its full design specifications.

End Semester Examination Pattern:

The marks will be distributed as

Presentation : **30 marks**
Demo : **20 marks**
Viva : **25 marks.**
Total : **75 marks**.

TEACHING PLAN

Students are expected to follow the following steps.

1. Review of Literature and Identification of a problem
2. Create an abstract with a problem statement, solution approach, technology stack, team, etc.
3. Create Requirements Specification
4. Create Design Document . This may include designs like,
    a. System Architecture Design
    b. Application Architecture Design
    c. GUI Design
    d. API Design
    e. Database Design
    f. Technology Stack
5. Deployment, Test Run & Get Results
6. Prepare Project Report

Guidelines for the Report preparation

A bonafide report on the mini project shall be submitted within one week after the final presentation. Minimum number of pages should be 40.

- Use Times New Roman font for the entire report – Chapter/Section Title – Times New Roman18, Bold; Heading 2 – Times New Roman16, Bold; Heading 3 – Times New Roman14, Bold; Body- Times New Roman 12, Normal.
- Line Spacing – Between Heading 2 – 3 lines, between lines in paragraph 1.5 lines.
- Alignments – Chapter/Section Title – Center, Heading 2 & 3 should be Left Aligned. Ensure that all body text is paragraph justified.

- Figures & Tables – Ensure that all Figures and Tables are suitably numbered and given proper names/headings. Write figure title under the figure and table title above the table.

- **Suggestive order of documentation:**
  i. Top Cover
  ii. Title page
  iii. Certification page
  iv. Acknowledgement
  v. Abstract
  vi. Table of Contents
  vii. List of Figures and Tables
  viii. Chapters
  ix. Appendices, if any
  x. References/Bibliography

SEMESTER VII

HONOURS

CST495	CYBER FORENSICS	CATEGORY	L	T	P	CREDIT	YEAR OF INTRODUCTION
		VAC	3	1	0	4	2019

Preamble: The course on Cyber Forensics aims at exploring the basics of Cyber Forensics and Cyber security, the forensic investigation process and principles and the different types of cybercrimes and threats. This course also focuses on the forensic analysis of File systems, the Network, the Windows and Linux Operating systems. The course gives a basic understanding of the forensics analysis tools and a deep understanding of Anti forensics practices and methods. All the above aspects are dealt with case studies of the respective areas.

Prerequisite: Knowledge in File Systems, Operating systems, Networks and a general awareness on Cyber Technologies.

Course Outcomes: After the completion of the course the student will be able to

CO1	Explain thebasic concepts in Cyber Forensics, Forensics Investigation Process and Cyber security(**Cognitive Knowledge Level: Understand**)
CO2	Infer the basic concepts of File Systems and its associated attribute definitions (**Cognitive Knowledge Level: Understand)**
CO3	Utilize the methodologies used in data analysis and memory analysis for detection of artefacts**(Cognitive Knowledge Level: Apply)**
CO4	Identify web attacks and detect artefacts using OWASP and penetration testing. **(Cognitive Knowledge Level: Apply)**
CO5	Summarize anti-forensics practices and data hiding methods **(Cognitive Knowledge Level: Understand)**

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	✓	✓				✓						✓
CO2	✓	✓										✓
CO3	✓	✓	✓	✓	✓							✓
CO4	✓	✓	✓	✓	✓							✓
CO5	✓	✓			✓							✓

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	**PO7**	Environment and Sustainability
PO2	Problem Analysis	**PO8**	Ethics
PO3	Design/Development of solutions	**PO9**	Individual and team work
PO4	Conduct investigations of complex problems	**PO10**	Communication
PO5	Modern tool usage	**PO11**	Project Management and Finance
PO6	The Engineer and Society	**PO12**	Life long learning

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks
	Test1 (Percentage)	Test2 (Percentage)	
Remember	30	30	30
Understand	40	40	40
Apply	30	30	30
Analyze			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks

Continuous Assessment Tests : 25 marks

Continuous Assessment Assignment : 15 marks

Internal Examination Pattern:

Each of the two internal examinations has to be conducted out of 50 marks.

First Internal Examination shall be preferably conducted after completing the first half of the syllabus and the Second Internal Examination shall be preferably conducted after completing remaining part of the syllabus.

There will be two parts: Part A and Part B. Part A contains 5 questions (preferably, 2 questions each from the completed modules and 1 question from the partly covered module), having 3 marks for each question adding up to 15 marks for part A. Students should answer all questions from Part A. Part B contains 7 questions (preferably, 3 questions each from the completed modules and 1 question from the partly covered module), each with 7 marks. Out of the 7 questions in Part B, a student should answer any 5.

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which a student should answer any one. Each question can have maximum 2 sub-divisions and carries 14 marks.

Syllabus

Module-1(Cyber Forensics and Cyber Security)

Computer Forensics: History of computer forensics, preparing for computer investigations, understanding Public and private investigations- Forensics Investigation Principles - Forensic Protocol for Evidence Acquisition - Digital Forensics -Standards and Guidelines - Digital Evidence – Data Acquisition - storage formats for digital evidence, determining the best acquisition method, contingency planning for image acquisitions, Cyber Forensics tools- Challenges in Cyber Forensics, Skills Required to Become a Cyber Forensic Expert

Cyber Security: Cybercrimes, Types of Cybercrimes - Recent Data Breaches - Recent Cyber security Trends - Case Study: Sim Swapping Fraud, ATM Card Cloning, Hacking email for money, Google Nest Guard, Email Crimes, Phishing, Types of Phishing.

Module-2 (File System Forensics)

File system Analysis: FAT and NTFS concepts and analysis -File system category, Content category, Metadata category, File name category, Application category,Application-level search techniques, Specific file systems, File recovery, Consistency check. FAT data structure-Boot sector, FAT 32 FS info, directory entries, Long file name directory entries

Module-3 (Operating System Forensics)

Windows Forensics: Live Response- Data Collection- Locard's Exchange Principle, Order of Volatility Volatile and Non Volatile Data Live-Response Methodologies: Data Analysis- Agile Analysis, Windows Memory Analysis, Rootkits and Rootkit detection.

Linux Forensics: Live Response Data Collection- Prepare the Target Media, Format the Drive, Gather Volatile Information, Acquiring the Image, Initial Triage, Data Analysis- Log Analysis, Keyword Searches, User Activity, Network Connections, Running Processes, Open File Handlers, The Hacking Top Ten, Reconnaissance Tools

Module-4 (Network Forensics)

The OSI Model, Forensic Footprints, Seizure of Networking Devices, Network Forensic Artifacts, ICMP Attacks, Drive-By Downloads, Network Forensic Analysis Tools, Case Study: Wireshark. Web Attack Forensics: OWASP Top 10, Web Attack Tests, Penetration Testing.

Module-5 (Anti-Forensics)

Anti-forensic Practices - Data Wiping and Shredding- Data Remanence, Degaussing, Case Study: USB Oblivion, Eraser - Trail Obfuscation: Spoofing, Data Modification, Case Study: Timestamp – Encryption, Case Study: VeraCrypt, Data Hiding: Steganography and Cryptography, Case Study: SilentEye, Anti-forensics Detection Techniques, Case Study: Stegdetect

Text Books

1. Bill Nelson, Amelia Phillips and Christopher Steuart, Computer forensics - Guide to Computer Forensics and Investigations, 4/e, Course Technology Inc.
2. Brian Carrier, File System Forensic Analysis, Addison Wesley, 2005.
3. Harlan Carvey, Windows Forensic Analysis DVD Toolkit, 2/e, Syngress.
4. Cory Altheide, Todd Haverkos, Chris Pogue,Unix and Linux Forensic Analysis DVD Toolkit, 1/e, Syngress.
5. William Stallings,Network Security Essentials Applications and Standards, 4/e, Prentice Hall
6. Eric Maiwald, Fundamentals of Network Security, McGraw-Hill, 2004.

References

1. Michael. E. Whitman, Herbert. J. Mattord, Principles of Information Security, Course Technology, 2011.
2. William Stallings, Cryptography and Network Security Principles and Practice, 4/e, Prentice Hall.
3. Niranjan Reddy, Practical Cyber Forensics: An Incident-Based Approach to Forensic Investigations, Apress, 2019.

Sample Course Level Assessment Questions

CourseOutcome1(CO1): Explain the Forensics principles and protocols for evidence acquisition.

Discuss the different cyber forensics tools used for image acquisition.

CourseOutcome2(CO2): Explain the pros and cons of NTFS and FAT File systems. Also give the challenges the investigators would face in extracting evidences from these file systems.

CourseOutcome3 (CO3): Apply any memory forensics methodologies/tools to extract volatile and nonvolatile data from a Windows based system.

CourseOutcome4 (CO4): Use web attacks test tools like netcraft to identify web application vulnerabilities of a particular site say **www.xyz.com**

Course Outcome 5 (CO5): Explain the different anti-forensics practices used to destroy or conceal data in order to prevent others from accessing it.

Model Question Paper

QP CODE:

Reg No: _____

Name: _____ **PAGES : 3**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST495

Course Name: Cyber Forensics

Max. Marks : 100 **Duration: 3 Hours**

PART A

Answer All Questions. Each Question Carries 3 Marks

1. Distinguish between public and private investigations.

2. What are the three computer forensics data acquisitions formats?

3. List any three features of NTFS which are not in FAT.

4. Define the terms file slack, RAM slack and drive slack.

5. What is Locard's exchange principle? Why is it important in forensic investigations?

6. Why would you conduct a live response on a running system?

7. What are the different tools used in Network Forensics?

8. Explain how Risk Analysis and Penetration Testing are different.

9. Why we are using Steganography?

10. How is data wiping done in hard drive?

(10x3=30)

Part B

(Answer any one question from each module. Each question carries 14 Marks)

11. (a) Discuss the different types of Cybercrimes. List the tools used for identifying Cyber Crimes. **(8)**

(b) Differentiate between Static acquisition and Live acquisition with example. **(6)**

OR

12. (a) Explain the principles of Digital Forensic Investigation? Why is it important? Comment. **(8)**

(b) When you perform an acquisition at a remote location, what should you consider preparing this task? **(6)**

13. (a) Discuss the FAT File Structure. **(8)**

(b) Does Windows NT use FAT or NTFS? Explain. **(6)**

14. (a) What is Metadata? Discuss the first 16 metadata records you would find in the MFT? **(6)**

(b) Explain the different data categories in a File System. **(8)**

15. (a) What is Agile requirement analysis? **(6)**

(b) Explain the different types of volatile information in a live response system. List any two tools used for obtaining volatile information. **(8)**

16. (a) What are the main live response methodologies? **(6)**

(b) What is Physical Memory Dump? Explain how a physical memory dump is analysed. **(8)**

17. (a) What is OWASP? Also mention the Top 10 web application vulnerabilities in 2021. **(8)**

(b) How would you setup Wireshark to monitor packets passing through aninternet router? **(6)**

18. (a) What are the goals of conducting a pentesting exercise? **(3)**

(b) Discuss the types of penetration testing methodologies. **(5)**

(c) Define OSI Layers. **(6)**

19. (a) How is Steganography done? **(7)**

(b) Why does data need Cryptography? **(4)**

(c) What is the difference between a Cryptographer and a Crypter? **(3)**

OR

20. (a) Explain the different types of Anti-forensics Detection Techniques. **(8)**

(b) What is Spoofing? How to prevent spoofing attack? **(6)**

TEACHING PLAN

Sl.No.	Contents	No of Lecture Hrs (44hrs)
Module-1 (Cyber Forensics and Cyber Security) (11 Hrs)		
1.1	History of computer forensics, preparing for computer investigations	1 hour
1.2	Understanding Public and private investigations- Forensics Investigation Principles	1 hour
1.3	Forensic Protocol for Evidence Acquisition	1 hour
1.4	Digital Forensics -Standards and Guidelines - Digital Evidence	1 hour
1.5	Data Acquisition - storage formats for digital evidence, determining the best acquisition method	1 hour
1.6	Contingency planning for image acquisitions, Cyber Forensics tools	1 hour
1.7	Challenges in Cyber Forensics, Skills Required to Become a Cyber Forensic Expert	1 hour
1.8	Cybercrimes, Types of Cybercrimes - Recent Data Breaches - Recent Cyber security Trends	1 hour
1.9	Case Study: Sim Swapping Fraud, ATM Card Cloning	1 hour
1.10	Case Study:Hacking email for money, Google Nest Guard	1 hour
1.11	Email Crimes, Phishing, Types of Phishing	1 hour
Module-2 (File System Forensics) (9 Hrs)		

2.1	FAT and NTFS concepts and analysis	1 hour
2.2	File system category, Content category	1 hour
2.3	Metadata category	1 hour
2.4	File name category,Application category	1 hour
2.5	Application-level search techniques	1 hour
2.6	Specific file systems, File recovery, Consistency check	1 hour
2.7	FAT data structure-Boot sector	1 hour
2.8	FAT 32 FS info, directory entries	1 hour
2.9	Long file name directory entries	1 hour
Module-3 (Operating System Forensics) (11 Hrs)		
3.1	Live Response- Data Collection- Locard's Exchange Principle	1 hour
3.2	Order of Volatility, Volatile and Non Volatile Data	1 hour
3.3	Live-Response Methodologies: Data Analysis- Agile Analysis	1 hour
3.4	Windows Memory Analysis	1 hour
3.5	Rootkits and Rootkit detection	1 hour
3.6	Linux Forensics: Live Response Data Collection	1 hour
3.7	Prepare the Target Media, Format the Drive, Gather Volatile Information	1 hour
3.8	Acquiring the Image, Initial Triage	1 hour
3.9	Data Analysis- Log Analysis, Keyword Searches, User Activity	1 hour

3.10	Data Analysis- Network Connections, Running Processes, Open File Handlers	1 hour
3.11	The Hacking Top Ten, Reconnaissance Tools	1 hour
Module-4 (Network Forensics) (7 Hrs)		
4.1	OSI Model	1 hour
4.2	Forensic Footprints, Seizure of Networking Devices, Network Forensic Artifacts	1 hour
4.3	ICMP Attacks, Drive-By Downloads, Network Forensic Analysis Tools	1 hour
4.4	Web Attack Forensics	1 hour
4.5	OWASP Top 10, Web Attack Tests	1 hour
4.6	Penetration Testing-1	1 hour
4.7	Penetration Testing.-2	1 hour
Module – 5 (Anti-Forensics) (6 Hrs)		
5.1	Anti-forensic Practices - Data Wiping and Shredding	1 hour
5.2	Data Remanence, Degaussing	1 hour
5.3	Trail Obfuscation: Spoofing, Data Modification	1 hour
5.4	Role of Encryption in Forensics	1 hour
5.5	Data Hiding: Steganography and Cryptography	1 hour
5.6	Anti-forensics Detection Techniques	1 hour

CST497	REINFORCEMENT LEARNING	CATEGORY	L	T	P	CREDIT	YEAR OF INTRODUCTION
		VAC	3	1	0	4	2019

Preamble: This course covers fundamental principles and techniques in reinforcement learning. Reinforcement learning is concerned with building programs that learn how to predict and act in a stochastic environment, based on past experience. Applications of reinforcement learning range from classical control problems, such as power plant optimization or dynamical system control, to game playing, inventory control, and many other fields. Topics include Markov decision process, dynamic programming, Monte Carlo, temporal difference, function approximation reinforcement learning algorithms, and applications of reinforcement learning. This course enables the leaners to apply reinforcement learning on real world applications and research problems.

Prerequisite: A pass in CST 294(Computational Fundamentals for Machine Learning)

Course Outcomes: After the completion of the course the student will be able to

CO 1	Solve computational problems using probability and random variables. **(Cognitive Knowledge Level: Apply)**
CO 2	Apply policy iteration and value iteration reinforcement learning algorithms. **(Cognitive Knowledge Level: Apply)**
CO 3	Employ Monte Carlo reinforcement learning algorithms. **(Cognitive Knowledge Level: Apply)**
CO 4	Apply temporal-difference reinforcement learning algorithms.(**Cognitive Knowledge Level: Apply)**
CO 5	Apply on-policy and off-policy reinforcement learning algorithms with function approximation. (**Cognitive Knowledge Level: Apply)**

Mapping of course outcomes with program outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	✓	✓	✓									✓
CO 2	✓	✓	✓	✓								✓
CO 3	✓	✓	✓	✓								✓
CO 4	✓	✓	✓	✓								✓
CO 5	✓	✓	✓	✓	✓							✓

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	**PO7**	Environment and Sustainability
PO2	Problem Analysis	**PO8**	Ethics
PO3	Design/Development of solutions	**PO9**	Individual and team work
PO4	Conduct investigations of complex problems	**PO10**	Communication
PO5	Modern tool usage	**PO11**	Project Management and Finance
PO6	The Engineer and Society	**PO12**	Life long learning

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	**2**	
Remember	**30%**	**30%**	**30%**
Understand	**30%**	**30%**	**30%**
Apply	**40%**	**40%**	**40%**
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	**50**	**100**	**3 hours**

Continuous Internal Evaluation Pattern:

Attendance **: 10 marks**

Continuous Assessment Tests **: 25 marks**

Continuous Assessment Assignment **: 15 marks**

Internal Examination Pattern:

Each of the two internal examinations has to be conducted out of 50 marks

First Internal Examination shall be preferably conducted after completing the first half of the syllabus and the Second Internal Examination shall be preferably conducted after completing remaining part of the syllabus.

There will be two parts: Part A and Part B. Part A contains 5 questions (preferably, 2 questions each from the completed modules and 1 question from the partly covered module), having 3 marks for each question adding up to 15 marks for part A. Students should answer all questions from Part A. Part B contains 7 questions (preferably, 3 questions each from the completed modules and 1 question from the partly covered module), each with 7 marks. Out of the 7 questions in Part B, a student should answer any 5.

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer anyone. Each question can have maximum 2 sub-divisions and carry 14 marks.

Syllabus

Module 1 (Review Of Probability Concepts)

Probability concepts review - Axioms of probability, concepts of random variables, probability mass function, probability density function, cumulative density functions, Expectation. Concepts of joint and multiple random variables, joint, conditional and marginal distributions. Correlation and independence.

Module 2 (Markov Decision Process)

Introduction to Reinforcement Learning(RL) terminology - Examples of RL, Elements of RL, Limitations and Scope of RL.

Finite Markov Decision Processes - The Agent–Environment Interface, Goals and Rewards, Returns and Episodes, Policies and Value Functions, Optimal Policies and Optimal Value Functions.

Module 3 (Prediction And Control)

Dynamic Programming - Policy Evaluation (Prediction), Policy Improvement, Policy Iteration, Value Iteration.

Monte Carlo (MC) for model free prediction and control - Monte Carlo Prediction, Monte

Carlo Estimation of Action Values, Monte Carlo Control, Monte Carlo Control without Exploring Starts, Off-policy Prediction via Importance Sampling, Incremental Implementation, Off-policy Monte Carlo Control.

Module 4 (Temporal-Difference (TD) Methods For Model Free Prediction And Control)

TD Methods - TD Prediction, Advantages of TD Prediction Methods, Optimality of TD(0), Sarsa: On-policy TD Control, Q-learning: Off-policy TD Control, Expected Sarsa.

n-step Bootstrapping- n-step TD Prediction, n-step Sarsa, step Off-policy Learning, Off-policy Learning Without Importance Sampling: The n-step Tree Backup Algorithm.

Module 5 (Function Approximation Method)

On-policy Prediction with Approximation - Value-function Approximation, The Prediction Objective, Stochastic-gradient Methods, Linear Methods.

Eligibility Traces - The λ-return, TD(λ), n-step Truncated λ-return Methods, Sarsa(λ).

Policy Gradient Methods - Policy Approximation and its Advantages, The Policy Gradient Theorem, REINFORCE: Monte Carlo Policy Gradient, REINFORCE with Baseline, Actor–Critic Methods.

Text book:

1 Richard S. Sutton and Andrew G. Barto, Reinforcement Learning: An Introduction, , 2nd Edition

2 Alberto Leon-Garcia, Probability, Statistics, and Random Processes for Electrical Engineering, 3rd Edition,

Reference books:

1 Reinforcement Learning: State-of-the-Art, Marco Wiering and Martijn van Otterlo, Eds

2 Algorithms for Reinforcement Learning, Szepesvari (2010), Morgan & Claypool.

3 Artificial Intelligence: A Modern Approach, Stuart J. Russell and Peter Norvig

4 Mathematical Statistics and Data Analysis by John A. Rice,University of California, Berkeley, Third edition, published by Cengage.

5 Machine Learning: A Probabilistic Perspective, Kevin P. Murphy

Sample Course Level Assessment Questions.

Course Outcome 1 (CO1):

1 Let ***J***and ***T***be independent events, where ***P(J)=0.4***and ***P(T)=0.7***. Find ***P(J ∩ T), P(J∪T)*** and ***P(J∩T')***

2 Let ***A*** and ***B*** be events such that ***P(A)=0.45*** , ***P(B)=0.35*** and ***P(A ∪B)=0.5*** Find ***P(A / B)***

3 A random variable **R**has the probability distribution as shown in the following table:

r	1	2	3	4	5
P(R=r)	0.2	a	b	0.25	0.15

Given that ***E(R)=2.85***, find ***a*** and ***b***and ***P(R>2)***.

4 A biased coin (with probability of obtaining a head equal to ***p > 0***) is tossed repeatedly and independently until the first head is observed. Compute the probability that the first head appears at an even numbered toss.

5 Two players A and B are competing at a quiz game involving a series of questions. On any individual question, the probabilities that A and B give the correct answer are ***p*** and ***q*** respectively, for all questions, with outcomes for different questions being independent. The game finishes when a player wins by answering a question correctly. Compute the probability that A wins if

(i) A answers the first question,

(ii) B answers the first question.

6 A coin for which ***P(heads) = p***is tossed until two successive tails are obtained. Find the probability that the experiment is completed on the***n***thtoss.

7 An urn contains **p** black balls, **q** white balls, and **r** red balls; and **n** balls are chosen without replacement.

i. Find the joint distribution of the numbers of black, white, and red balls in the sample.

ii. Find the joint distribution of the numbers of black and white balls in the sample.

iii. Find the marginal distribution of the number of white balls in the sample.

8 Suppose that two components have independent exponentially distributed lifetimes, ***T1***and T2, with parameters ***α***and ***β***, respectively. Find (a) ***P(T1> T2)***and (b) ***P(T1> 2 T2)***.

9 Let ***Z1*** and ***Z2*** be independent random variables each having the standard normal distribution. Define the random variables ***X*** and ***Y*** by ***X = Z1 + 3Z2*** and Y = ***Z1+ Z2***. Argue that the joint distribution of ***(X, Y***) is a bivariate normal distribution. What are the parameters of this distribution?

10 Given a continuous random variable ***x***, with cumulative distribution function ***Fx(x)***, show that the random variable ***y = Fx(x)*** is uniformly distributed.

11 ou roll a fair dice twice. Let the random variable ***X*** be the product of the outcomes of the two rolls. What is the probability mass function of ***X***? What are the expected values and the standard deviation of ***X***?

12 Show that if two events ***A***and ***B***are independent, then ***A***and ***B'***are independent

13 Prove that ***X***and ***Y***are independent if and only if ***fX|Y (x|y) = fX(x)***for all ***x***and ***y***

14 A random square has a side length that is a uniform [0, 1] random variable. Find the expected area of the square.Let X be a continuous random variable with the density function f (x) = 2x, $0 \leq x \leq 1$

i. Find ***E(X)***.

ii. Find $E(X^2)$and ***Var(X)***.

Course Outcome 2 (CO2):

1 What are the main differences between supervised learning and reinforcement learning?

2 Give examples of Markovian and non-Markovian environments?

3 What are the advantages and disadvantages of value methods vs policy methods?

4 Define the optimal state-value function $\mathbf{V^*(s)}$ for an MDP.

5 Imagine that the rewards are at most 1 everywhere. What is the maximum value that the discounted return can attain ? Why ?

6 Write down the Bellman optimality equation for state-value functions

7 Suppose that you are in a casino. You have Rs 20 and will play until you lose it all or as soon as you double your money. You can choose to play two slot machines: 1) slot machine A costs Rs 10 to play and will return Rs 20 with probability 0.05 and Rs 0 otherwise; and 2) slot machine B costs Rs 20 to play and will return Rs30 with probability 0.01 and Rs 0 otherwise. Until you are done, you will choose to play machine A or machine B in each turn. Describe the state space, action space, rewards and transition probabilities. Assume the discount factor $\gamma = 1$. Rewards should yield a higher reward when terminating with Rs 40 than when terminating with Rs 0. Also, the reward for terminating with Rs 40 should be the same regardless of how we got there (and equivalently for Rs 0).

Course Outcome 3 (CO3):

1 Explain policy iteration and value iteration? What are their similarities and differences?

2 Why Monte Carlo methods for learning value functions require episodic tasks? How is it that n-step TD methods avoid this limitation and can work with continuing tasks?

3 List any three uses of the depth parameter in the Monte-Carlo tree search procedure.

4 Given that $\mathbf{q_\pi(s, a) > v_\pi(s)}$, can we conclude that $\boldsymbol{\pi}$ is not an optimal policy. Justify

Course Outcome 4 (CO4):

1 Draw the backup diagram for 2-step Sarsa. Write the corresponding learning rule for 2-step Sarsa.
2 Why is Sarsa an on-policy algorithm while Q-learning is an off-policy algorithm?
3 How would you differentiate between learning algorithms using on-policy from those that use off-policy?
4 When using Temporal Difference learning, why is it better to learn action values (Q-values) rather than state values (V-values)?
5 Supose that a Q-learning agent always chooses the action which maximizes the Q-value. What is one potential problem with that approach?
6 Describe any two ways that will force a Q-learning agent to explore.
7 Why and when do we need importance sampling?

Course Outcome 5 (CO5):

1 How do you deal with a large possible action space in reinforcement learning?

2 List any two benefits of policy gradient methods over value function based methods.

3 What is the relation between Q-learning and policy gradients methods?

4 Consider a five state random walk. There are five states, $\mathbf{s_1, s_2, ..., s_5,}$ in a row with two actions each, left and right. There are two terminal states at each end, with a reward of **+1** for terminating on the right, after $\mathbf{s_5}$ and a reward of **0** for all other transitions, including the one terminating on the left after $\mathbf{s_1}$. In designing a linear function approximator, what is the least number of state features required to represent the value of the equi-probable random policy?

Model Question paper

QP Code : **Total Pages: 4**

Reg No.:_____ Name:_____

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION (HONOURS), MONTH and YEAR

Course Code: CST497

Course Name: REINFORCEMENT LEARNING

Max. Marks: 100 Duration: 3 Hours

PART A

Answer all questions, each carries 3 marks.

1 The first three digits of a telephone number are 452. If all the sequences of the remaining four digits are equally likely, what is the probability that a randomly selected telephone number contains seven distinct digits?

2 If X is a discrete uniform random variable, i.e., $P(X = k) = 1/n$ for $k = 1, 2, \ldots, n$, find $E(X)$and $Var(X)$.

3 Define the discounted return $\mathbf{G_t}$. Give an expression for $\mathbf{G_t}$in terms of $\mathbf{G_{t+1}}$.

4 Write down the Bellman expectation equation for state-value functions.

5 Suppose that we are doing value iteration with $\gamma = 0$. How many iterations will it take for value iteration to converge to the optimal value function?

6 List any three advantages of Monte Carlo methods over dynamic programming techniques?

7 Draw the backup diagram for 2-step Q-learning. Write the corresponding learning rule for 2-step Q-learning.

8 Why Monte Carlo methods for learning value functions require episodic tasks. How does **n**-step TD methods avoid this limitation and can work with continuing tasks?

9 In using policy gradient methods, if we make use of the average reward formulation rather than the discounted reward formulation, then is it necessary to consider, for problems that do not have a unique start state, a designated start state, $\mathbf{s_0}$? Justify.

10 Value function based methods are oriented towards finding deterministic

policies whereas policy search methods are geared towards finding stochastic policies. True or false? Justify.

10 x 3 = 30

PART B

Answer any one Question from each module. Each question carries 14 Marks

11 a) Three players play 10 independent rounds of a game, and each player has probability 1/3 of winning each round. Find the joint distribution of the numbers of games won by each of the three players. **(7)**

b) Find the joint density of $X + Y$ and X/Y, where X and Y are independent exponential random variables with parameter λ. Show that $X + Y$ and X/Y are independent. **(7)**

OR

12 a) An experiment consists of throwing a fair coin four times. Find the probability mass function and the cumulative distribution function of the following random variables: (7)

i the number of heads before the first tail

ii the number of heads following the first tail

iii the number of heads minus the number of tails

iv the number of tails times the number of heads.

b) Let X be a continuous random variable with probability density function on $0 <= x <= 1$ defined by $f(x) = 3x^2$. Find the pdf of $Y = X^2$. **(7)**

13 a) What is the difference between a state value function **V(s)** and a state-action value function **Q(s,a)**? **(4)**

b) Consider designing a recycling robot whose job is to collect empty bottles around the building. The robot has a sensor to detect when a bottle is in front of it, and a gripper to pick up the bottle. It also senses the level of its battery. The robot can navigate, as well as pick up a bottle and throw a bottle it is holding in the trash. There is a battery charger in the building, and the robot should not run out of battery. **(10)**

i. Describe this problem as an MDP. What are the states and actions?

ii. Suppose that you want the robot to collect as many bottles as possible, while not running out of battery. Describe what rewards would enable it to achieve this task.

OR

14 a) Define the state-value function $\mathbf{V_\pi(s)}$ for a discounted MDP. **(5)**

b) Consider a 4x4 gridworld where the agent starts in the top left, the bottom rig state is terminal, rewards are always **-1**, γ **= 1**, and state transitions a deterministic. Consider the policy that always chooses the action to move dow except when it is on the bottom row, at which point it chooses the action to mo right. Starting with $\mathbf{v_0(s) = 0}$ for all **s**, compute $\mathbf{v_1, v_2, \ldots, v_7}$. **(10)**

15 a) During a single iteration of the Value Iteration algorithm, we typically iterate over the states in **S** in some order to update $\mathbf{V_t(s)}$ to $\mathbf{V_{t+1}(s)}$ for all states **s**. Is it possible to do this iterative process in parallel? Explain why or why not. **(5)**

b) Consider an undiscounted Markov Reward Process with two states A and B. The transition matrix and reward function are unknown, but you have observed two sample episodes: **(9)**

A +3 --> A +2 --> B -4 --> A +4 --> B -3

B -2 --> A +3 --> B -3

i. Using first-visit Monte-Carlo evaluation, estimate the state-value function **V(A),V(B)**.

ii. Using every-visit Monte-Carlo evaluation, estimate the state-value function **V(A),V(B).**

iii. Draw a diagram of the Markov Reward Process that best explains these two episodes. Show rewards and transition probabilities on your diagram.

OR

16 a) Suppose you are given a finite set of transition data. Assuming that the Markov model that can be formed with the given data is the actual MDP from which the data is generated, will the value functions calculated by the MC and TD methods necessarily agree? Justify. **(4)**

b) With respect to the expected Sarsa algorithm, is exploration required as it is in the normal Sarsa and Q-learning algorithms? Justify. **(5)**

c) For a specific MDP, suppose we have a policy that we want to evaluate through the use of actual experience in the environment alone and using Monte Carlo methods. We decide to use the first-visit approach along with the technique of always picking the start state at random from the available set of states. Will this approach ensure complete evaluation of the action value function corresponding to the policy? **(5)**

17 a) Consider the following **Q[S,A]** table **(9)**

	State 1	State 2
Action 1	1.5	2.5
Action 2	4	3

Assume the discount factor, γ= **0.5**, and the step size, α = **0.1**. After the experience **(s, a, r, s')=(1, 1, 5, 2),** which value of the table gets updated and what is its new value?

b) What is the difference between Q-learning and Sarsa? **(5)**

OR

18 a) Consider the following **Q[S,A]** table **(9)**

	State 1	State 2
Action 1	1.5	2.5
Action 2	4	3

Assume the discount factor, γ= **0.5**, and the step size, α = **0.1**. After the experience **(s, a, r, s', a')=(1, 1, 5, 2, 1),** which value of the table gets updated and what is its new value?

b) For Q-learning to converge we need to correctly manage the exploration vs. exploitation tradeoff. What property needs to be hold for the exploration strategy? **(5)**

19 a) Given the following sequence of states observed from the beginning of an episode, $s_2, s_1, s_3, s_2, s_1, s_2, s_1, s_6$, what is the eligibility value, $e_7(s_1)$, of state $s_1$at time step 7 given trace decay parameter λ, discount rate γ, and initial value, $e_0(s_1) = 0$, when accumulating traces are used? What is the eligibility value if replacing traces are used? **(8)**

b) Suppose that we are using a policy gradient method to solve a reinforcement learning problem and the policy returned by the method is not optimal. Give three plausible reasons for such an outcome? **(6)**

OR

20 a) Suppose that we have a Q-value function represented as a sigmoid function of a set of features: **(8)**

$$Q(\phi, a) = \frac{1}{1 + e^{\theta^T \phi}}$$

Write down the update rule that Sarsa would give for this function.

b) Suppose that in a particular problem, the agent keeps going back to the same state in a loop. What is the maximum value that can be taken by the eligibility trace of such a state if we consider accumulating traces with **λ = 0.25** and **γ = 0.8**? **(6)**

Teaching Plan

No	Topic	No. of Lectures (42)
Module-1 (Review Of Probability Concepts) TB-2(Ch 2,3,4,5) (8 hours)		
1.1	Axioms of probability, concepts of random variables	1 hour
1.2	Probability mass function	1 hour
1.3	Probability density function	1 hour
1.4	Cumulative density functions	1 hour
1.5.	Expectation of random variables	1 hour
1.6.	Joint and multiple random variables	1 hour
1.7	Conditional and marginal distributions	1 hour
1.8	Correlation and independence	1 hour
Module-2 (Markov Decision Process) TB-1(Ch 1,3)(8 hours)		
2.1.	Introduction to Reinforcement Learning(RL) terminology - Examples of RL, Elements of RL, Limitations and Scope of RL	1 hour
2.2	Finite Markov Decision Processes	1 hour
2.3	The Agent–Environment Interface	1 hour
2.4.	Goals and Rewards	1 hour
2.5.	Returns and Episodes	1 hour
2.6.	Policies and Value Functions	1 hour
2.7	Optimal Policies and Optimal Value Functions	1 hour
2.8	Optimal Policies and Optimal Value Functions	1 hour
Module-3 (Prediction And Control) TB-1(Ch 4,5) (9 hours)		

3.1	Policy Evaluation (Prediction)	1 hour
3.2	Policy Improvement	1 hour
3.3	Policy Iteration, Value Iteration	1 hour
3.4	Monte Carlo Prediction	1 hour
3.5	Monte Carlo Estimation of Action Values	1 hour
3.6	Monte Carlo Control, Monte Carlo Control without Exploring Starts	1 hour
3.7	Off-policy Prediction via Importance Sampling	1 hour
3.8	Incremental Implementation	1 hour
3.9	Off-policy Monte Carlo Control	1 hour
Module-4 (Temporal-Difference(Td) Methods) TB-1 (Ch 6,7) (8 hours)		
4.1	TD Prediction, Advantages of TD Prediction Methods	1 hour
4.2	Optimality of TD(0)	1 hour
4.3	Sarsa: On-policy TD Control	1 hour
4.4	Q-learning: Off-policy TD Control	1 hour
4.5	Expected Sarsa	1 hour
4.6	n-step TD Prediction, n-step Sarsa	1 hour
4.7	n-step Off-policy Learning	1 hour
4.8	Off-policy Learning Without Importance Sampling: The n-step Tree Backup Algorithm	1 hour
Module-5 (Function Approximation Method) TB-1 (Ch 9,12,13) (9 hours)		
5.1	Value-function Approximation	1 hour
5.2	The Prediction Objective	1 hour
5.3	Stochastic-gradient Methods	1 hour
5.4	Linear Methods	1 hour
5.5	The Lambda-return , TD(Lambda)	1 hour
5.6	n-step Truncated Lambda-return Methods, Sarsa(Lambda)	1 hour
5.7	Policy Approximation and its Advantages	1 hour
5.8	The Policy Gradient Theorem, REINFORCE: Monte Carlo Policy Gradient	1 hour
5.9	REINFORCE with Baseline, Actor–Critic Methods	1 hour

CST499	LOGIC FOR COMPUTER SCIENCE	CATEGORY	L	T	P	CREDIT	YEAR OF INTRODUCTION
		VAC	3	1	0	4	2019

Preamble: This course enables the learners to understand the concepts of various logics used in computer science. The course covers the standard and most popular logics such as propositional logic, predicate logic, linear temporal logic, computation tree logic, Hoare logic and modal logic. This course helps the students to develop solutions for specification and verification of real world systems.

Prerequisite: Nil

Mapping of course outcomes with program outcomes

CO1	Explain the concepts of Predicate Logic, Propositional Logic, Linear Temporal Logic, Computation Tree Logic, Hoare Logic and Modal Logic as a formal language. **(Cognitive Knowledge Level: Understand)**
CO2	Develop proofs to show the satisfiability, validity and equivalence of logic formulas.**(Cognitive Knowledge Level: Apply)**
CO3	Illustrate model checking and program verification to prove correctness of systems.**(Cognitive Knowledge Level: Apply)**
CO4	Demonstrate *Alloy Analyzer* to model and analyze software systems. **(Cognitive Knowledge Level: Apply)**
CO5	Demonstrate *New Symbolic Model Verifier (NuSMV)* as a model checking tool to check the validity of temporal logic formulas.**(Cognitive Knowledge Level: Apply)**

Mapping of course outcomes with program outcomes

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

CO5	✓	✓	✓	✓	✓							✓

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	**PO7**	Environment and Sustainability
PO2	Problem Analysis	**PO8**	Ethics
PO3	Design/Development of solutions	**PO9**	Individual and team work
PO4	Conduct investigations of complex problems	**PO10**	Communication
PO5	Modern tool usage	**PO11**	Project Management and Finance
PO6	The Engineer and Society	**PO12**	Life long learning

Assessment Pattern

Bloom's Category	**Continuous Assessment Tests**		**End Semester Examination Marks (%)**
	Test 1 (%)	**Test 2 (%)**	
Remember	**30**	**30**	**30**
Understand	**30**	**30**	**30**
Apply	**40**	**40**	**40**
Analyze			
Evaluate			
Create			

Mark Distribution

Total Marks	**CIE Marks**	**ESE Marks**	**ESE Duration**
150	**50**	**100**	**3**

Continuous Internal Evaluation Pattern:

Attendance	**10 marks**
Continuous Assessment Tests(Average of Internal Tests1&2)	**25 marks**
Continuous Assessment Assignment	**15 marks**

Internal Examination Pattern

Each of the two internal examinations has to be conducted out of 50 marks. First series test shall be preferably conducted after completing the first half of the syllabus and the second series test shall be preferably conducted after completing remaining part of the syllabus. There will be two parts: Part A and Part B. Part A contains 5 questions (preferably, 2 questions each from the completed modules and 1 question from the partly completed module), having 3 marks for each question adding up to 15 marks for part A. Students should answer all questions from Part A. Part B contains 7 questions (preferably, 3 questions each from the completed modules and 1 question from the partly completed module), each with 7 marks. Out of the 7 questions, a student should answer any 5.

End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 full questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carries 14 marks.

Syllabus

Module – 1 (Propositional Logic)

Declarative Sentences, Natural Deduction, Propositional Logic as a Formal Language, Semantics of Propositional Logic, Normal Forms, SAT Solvers.

Module– 2(Predicate Logic)

The Need of a Richer Language, Predicate Logic as a Formal Language, Proof Theory of Predicate Logic, Semantics of Predicate Logic, Undecidability of Predicate Logic, Expressiveness of Predicate Logic.

Module - 3 (Verification by Model Checking)

Motivation for Verification, Linear Time Temporal Logic (LTL), Model Checking Systems, Tools, Properties, Branching Time Logic, Computation Tree Logic (CTL) and the Expressive Powers of LTL and CTL, Model Checking Algorithms, The Fixed Point Characterization of CTL.

Module–4 (Program Verification)

Why Should We Specify and Verify Code, A Framework for Software Verification, Proof Calculus for Partial Correctness, Proof Calculus for Total Correctness, Programming by Contract.

Module–5 (Modal Logics and Agents)

Modes of Truth, Basic Modal Logic, Logic Engineering, Natural Deduction, Reasoning about Knowledge in a Multi-Agent System.

Text Books

1. Michael Huth and Mark Ryan, Logic in Computer Science, 2/e, Cambridge University Press, 2004.

Reference Books

1. Daniel Jackson, Software Abstractions, MIT Press, 2011.
2. Roberto Cavada, Alessandro Cimatti, Gavin Keighren, Emanuele Olivetti, Marco Pistore and Marco Roveri, NuSMV 2.6 Tutorial (available at *https://nusmv.fbk.eu*).
3. Tutorial for Alloy Analyzer 4.0 (available at *https://alloytools.org/tutorials/online/*).

Course Level Assessment Questions

Course Outcome1 (CO1):

1. Express the following statements as appropriate logic formulas.
   a. If the barometer falls, either it will rain or it will snow.
   b. No student attended every lecture.
   c. Once you are on the field, you keep on playing until the game is over.
   d. There are eight planets in the solar system.
2. Explain Horn Clause and Horn Formula.
3. Explain modal logic.

Course Outcome 2(CO2):

1. Prove the validity of the following sequents.

   $(p \wedge q) \wedge r, s \wedge t \vdash q \wedge s$

2. Prove the validity of
   (a) $\forall x \forall y\, P(x, y) \vdash \forall u \forall v\, P(u, v)$
   (b) $\exists x \exists y\, F(x, y) \vdash \exists u \exists v\, F(u, v)$
   (c) $\exists x \forall y\, P(x, y) \vdash \forall y \exists x\, P(x, y)$.
3. Prove that for all paths π of all models, $\pi \models \phi\, W\, \psi \wedge F\, \psi$ implies $\pi \models \phi\, U\, \psi$.

Course Outcome 3(CO3):

1. Consider an LTL formula $\phi \equiv (a\ U\ b) \longrightarrow F\ b$. Is ϕ valid? If yes, give an automata-theoretic proof of validity (i.e., construct a suitable NBA and use nested DFS to check an appropriate persistence condition). Otherwise, give a transition system that violates the formula. Illustrate the constructions clearly.

2. A familiar command missing from the core language (described in the text book) is the *for-statement*. It may be used to sum the elements in an array, for example, by programming as follows:

```
s = 0;
for (i = 0; i <= max; i = i+1) {
        s = s + a[i];
}
```

After performing the initial assignment s = 0, this executes i = 0 first, then executes the body $s = s + a[i]$ and the incrementation $i = i + 1$ continually until$i <= max$ becomes false. Explain how ***for(C1;B;C2) {C3}*** can be defined as a derived program in our core language.

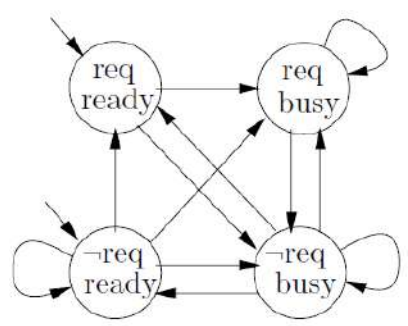
Course Outcome 4(CO4): .

1. Use *Alloy Analyzer* to model and solve the following problem.

   A farmer is on one shore of a river and has with him a fox, a chicken, and a sack of grain. He has a boat that fits one object besides himself. In the presence of the farmer nothing gets eaten, but if left without the farmer, the fox will eat the chicken, and the chicken will eat the grain. How can the farmer get all three possessions across the river safely?

Course Outcome 5(CO5):

1. Simulate the following system using NuSMV..



Verify that $G\ (req\ \longrightarrow F\ busy)$ holds in all initial states.

Model Question Paper

QP CODE:

Reg No: _____

Name: _____ **PAGES : 4**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SEVENTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST499

Course Name: Logic for Computer Science

Max.Marks:100 **Duration: 3 Hours**

PART A

Answer All Questions. Each Question Carries 3 Marks

1. Check the validity of the following sequents.
   a. $\sim p \longrightarrow \sim q \vdash q \longrightarrow p$
   b. $\sim(\sim p \vee q) \vdash p)$

2. For the formula $\phi = p \wedge \sim(q \vee \sim p)$, we compute the inductively defined translation as $T(\phi) = p \wedge \sim\sim(\sim q \wedge \sim\sim p)$. Draw the parse tree of $T(\phi)$.

3. Translate the following into predicate logic.
   a. All red things are in the box.
   b. No animal is both a cat and a dog.

4. Let ϕ be $\exists x\ (P(y,z) \wedge\ (\forall y\ (\sim Q(y,x) \vee\ P(y,z))))$, where P and Q are predicate symbols with two arguments. Identify all bound and free variables in ϕ.

5. Show the syntax of Computation Tree Logic (CTL).

6. Prove that the LTL equivalence between $\phi\ U\ \psi$and~ $\big(\sim\psi\ U\ (\sim\phi\ \wedge \sim\psi)\big) \wedge F\psi$.

7. Explain the need of specification and verification of code.

8. In what circumstances would $if\ (B)\{C1\}\ else\ \{C2\}$ fail to terminate?

9. Illustrate Kripke model.

10. With an example, explain the equivalences between modal formulas.

(10x3=30)

Part B

(Answer any one question from each module. Each question carries 14 Marks)

11. (a) Give the rules for Natural Deduction in propositional logic. **(6)**

(b) Use Natural Deduction to show the equivalence of the following formulas. **(8)**

a. $(p \wedge q) \wedge r, s \wedge t \vdash q \wedge s$

b. $(q \longrightarrow r) \longrightarrow ((\sim q \longrightarrow \sim p) \longrightarrow (p \longrightarrow r))$

OR

12. (a) What is a Horn Formula? How do you decide the satisfiability of a Horn formula. **(6)**

(b) Check the satisfiability of the following Horn Formulas. **(8)**

a. $(p \wedge q \wedge s \longrightarrow p) \wedge (q \wedge r \longrightarrow p) \wedge (p \wedge s \longrightarrow s)$

b. $(T \longrightarrow q) \wedge (T \longrightarrow s) \wedge (w \longrightarrow \perp) \wedge (p \wedge q \wedge s \longrightarrow v) \wedge (v \longrightarrow s) \wedge (T \longrightarrow r) \wedge (r \longrightarrow p)$

13. (a) Use Natural Deduction to prove the following equivalences. **(8)**

a. $\forall x\big(Q(x) \rightarrow R(x)\big), \exists x(P(x) \wedge Q(x)) \vdash \exists x(P(x) \wedge R(x))$

b. $\exists x P(x), \forall x\big(P(x) \longrightarrow Q(x)\big) \vdash \forall y Q(y)$

(b) Illustrate how Quantifier Equivalences can be used to check the equivalence of predicate logic formulas. **(6)**

OR

14. (a) Model the following system in *Alloy Analyzer*. **(7)**

There is an entity named **Person**, **Man** and **Woman** are two specializations of the entity **Person**. Every **Person** has a **Father** (a **Man**) and a **Mother** as **Parent**. The **Parent**s of a **Person** should be married. A **Man**'s **spouse** should be a **Woman** and a **Woman**'s **spouse** should be a **Man**. The **spouse** relation is symmetric.

Add a predicate to check whether marriage between siblings is possible in the above system.

(b) Explain Existential Second Order Logic and Universal Second Order Logic. **(7)**

15. (a) Model the Ferryman problem using New Symbolic Model Verifier (NuSMV). **(7)**

(b) Construct a Generalized Buchi Automaton for the LTL formula $\mathcal{O}a$. **(7)**

OR

16. (a) Show the closure of the LTL formula $\sim p\ U\ (F\ r\ \bigvee G\ \sim q\ \longrightarrow q\ W\ \sim r)$. **(7)**

(b) Explain the Fixed Point Characterization of CTL. **(7)**

17. (a) Illustrate partial correctness and total correctness in program verification. **(7)**

(b)
```
boolean withdraw(amount: int) {
  if (amount < 0 && isGood(amount))
     { balance = balance - amount;
       return true;
     } else { return false; }
}
```
(7)

Consider the method named *withdraw* which attempts to withdraw amount from an integer field *balance* of the class within which the method *withdraw* lives. This method makes use of another method *isGood* which returns true iff the value of *balance* is greater than or equal to the value of *amount*.

Write a contract for the method *isGood*. Use that contract to show the validity of the contract for *withdraw*:

Method name: *withdraw*

Input: *amount* of type int

Assumes: *0 <= balance*

Guarantees: *0 <= balanace*

Output: of type boolean

Modifies only: *balance*

Upon validation, this contract establishes that all calls to *withdraw* leave *0<=balance* invariant.

OR

18. (a) Consider the program for computing the factorial of a number as given below. **(7)**

```
y = 1;
z = 0;
while (z != x) {
    z = z + 1;
    y = y * z;
}
```

Find a partial correctness proof for the above program.

(b) Explain the proof calculus for total correctness. **(7)**

19. (a) Let $\mathcal{F} = (W, R)$ be a frame. Prove the two claims given below. **(7)**

1. The following statements are equivalent:
   - R is reflexive;
   - $\mathcal{F}$ satisfies $\Box\phi \rightarrow \phi$;
   - $\mathcal{F}$ satisfies $\Box p \rightarrow p$;
2. The following statements are equivalent:
   - R is transitive;
   - $\mathcal{F}$ satisfies $\Box\phi \rightarrow \Box\Box\phi$;
   - $\mathcal{F}$ satisfies $\Box p \rightarrow \Box\Box p$.

(b) Explain the modal logics *K*, *KT45* and *KT4*. **(7)**

OR

20. (a) Prove the following using Natural Deduction. **(8)**

$\vdash_{\mathrm{KT45}} p \rightarrow \Box\Diamond p$, $\vdash_{\mathrm{KT45}} \Box\Diamond\Box p \rightarrow \Box p$

(b) Find a modal logic to formalize and solve *The Wise-Men Puzzle*. **(6)**

Teaching Plan

No	Contents	No. of Lecture Hours (45 hrs)
Module-1(Propositional Logic) (8 hours)		
1.1	Declarative Sentences, Natural Deduction	1 hour
1.2	Rule for Natural Deduction	1 hour
1.3	Derived Rules, Natural Deduction in Summary	1 hour
1.4	Provable Equivalence, Proof by Contradiction. Propositional Logic as a Formal language	1 hour
1.5	Semantics of Propositional Logic – The Meaning of Logical Connectives, Soundness of Propositional Logic, Completeness of Propositional Logic (Proof not required)	1 hour
1.6	Semantic Equivalence, Satisfiability and Validity	1 hour
1.7	Normal Forms – Conjunctive Normal Forms and Validity, Horn Clauses and Satisfiability	1 hour
1.8	SAT Solvers – A Linear Solver, A Cubic Solver	1 hour
Module-2(Predicate Logic) (7 hours)		
2.1	The Need of a Richer language, Predicate Logic as a Formal Language – Terms, Formulas, Free and Bound Variables, Substitution	1 hour
2.2	Proof Theory of Predicate Logic – Natural Deduction Rules	1 hour
2.3	Proof Theory of Predicate Logic – Quantifier Equivalences	1 hour
2.4	Semantics of Predicate Logic – Models, Semantic Entailment, The Semantics of Equality	1 hour
2.5	Undecidabilty of Predicate Logic (*no proof required*), Expressiveness of Predicate Logic – Existential Second Order Logic, Universal Second Order Logic	1 hour
2.6	Micromodels of Software – State Machines, A Software Micromodel (*Alloy*) (Lecture 1)	1 hour
2.7	A Software Micromodel (*Alloy*) (Lecture 2)	1 hour
Module-3(Verification by Model Checking) (13 hours)		

3.1	Motivation for Verification, Linear Time Temporal Logic (LTL) - Syntax	1 hour
3.2	Semantics of LTL – Practical Patterns of Specifications, Important Equivalences between LTL Formulas, Adequate Sets of Connectives for LTL	1 hour
3.3	Introduction to model checking	1 hour
3.4	Model Checking Systems, Tools, Properties	1 hour
3.5	Model checking example: Mutual Exclusion	1 hour
3.6	The New Symbolic Model Verifier (NuSMV) Model Checker – Introduction, Mutual Exclusion Revisited	1 hour
3.7	The NuSMV Model Checker – The Ferryman, The Alternating Bit Protocol	1 hour
3.8	Branching Time Logic – Syntax of Computation Tree Logic (CTL), Semantics of CTL	1 hour
3.9	Practical Patterns of Specification, Important Equivalences between CTL Formulas, Adequate Sets of CTL Connectives	1 hour
3.10	CTL and the Expressive Powers of LTL and CTL – Boolean Combinations of Temporal Formulas in CTL	1 hour
3.11	Model-Checking Algorithms – The CTL Model Checking Algorithm	1 hour
3.12	CTL Model Checking with Fairness	1 hour
3.13	The LTL Model Checking Algorithm (Algorithm only)	1 hour
Module-4 (Program Verification) (8 hours)		
4.1	Introduction to Program Verification, Need of Specification and Verification of Code	1 hour
4.2	A Framework for Software Verification – A Core Programming Language, Hoare Triples	1 hour
4.3	A Framework for Software Verification – Partial and Total Correctness, Program Variables and Logical Variables	1 hour
4.4	Proof Calculus for partial Correctness – Proof Rules	1 hour
4.5	Proof Calculus for partial Correctness – Proof Tableaux	1 hour
4.6	Proof Calculus for partial Correctness – A Case Study: Minimal-Sum Section	1 hour
4.7	Proof Calculus for Total Correctness	1 hour
4.8	Programming by Contract	1 hour
Module-5 (Modal Logics and Agents) (9 hours)		

5.1	Modes of Truth, basic Modal Logic - Syntax	1 hour
5.2	Basic Modal Logic - Semantics	1 hour
5.3	Logic Engineering – The Stock of Valid Formulas, Important Properties of the Accessibility Relation	1 hour
5.4	Logic Engineering – Correspondence Theory, Some Modal Logics	1 hour
5.5	Natural Deduction	1 hour
5.6	Reasoning about Knowledge in a Multi-Agent System –Examples (The Wise - Man Puzzle, The Muddy – Children Puzzle)	1 hour
5.7	The Modal Logic KT45n	1 hour
5.8	Natural Deduction for KT45n	1 hour
5.9	Formalizing the Examples (The Wise - Man Puzzle, The Muddy – Children Puzzle)	1 hour