SEMESTER VI

SEMESTER VI

CST 302	COMPILER DESIGN	Category	L	T	P	Credit	Year of Introduction
		PCC	3	1	0	4	2019

Preamble:

The purpose of this course is to create awareness among students about the phases of a compiler and the techniques for designing a compiler. This course covers the fundamental concepts of different phases of compilation such as lexical analysis, syntax analysis, semantic analysis, intermediate code generation, code optimization and code generation. Students can apply this knowledge in design and development of compilers.

Prerequisite: Sound knowledge in Data Structures, Formal Languages & Automata Theory.

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

CO1	Explain the phases in compilation process(lexical analysis, syntax analysis, semantic analysis, intermediate code generation, code optimization and code generation) and model a lexical analyzer **(Cognitive Knowledge Level: Apply)**
CO2	Model language syntax using Context Free Grammar and develop parse tree representation using leftmost and rightmost derivations (**Cognitive Knowledge Level: Apply)**
CO3	Compare different types of parsers(Bottom-up and Top-down) and construct parser for a given grammar **(Cognitive Knowledge Level: Apply)**
CO4	Build Syntax Directed Translation for a context free grammar, compare various storage allocation strategies and classify intermediate representations **(Cognitive Knowledge Level: Apply)**
CO5	Illustrate code optimization and code generation techniques in compilation **(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 (Marks)	Test 2 (Marks)	
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 hours**

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 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 maximum 2 sub-divisions and carries 14 marks.

Syllabus

Module - 1 (Introduction to compilers and lexical analysis)

Analysis of the source program - Analysis and synthesis phases, Phases of a compiler. Compiler writing tools. Bootstrapping. Lexical Analysis - Role of Lexical Analyser, Input Buffering, Specification of Tokens, Recognition of Tokens.

Module - 2 (Introduction to Syntax Analysis)

Role of the Syntax Analyser – Syntax error handling. Review of Context Free Grammars - Derivation and Parse Trees, Eliminating Ambiguity. Basic parsing approaches - Eliminating left recursion, left factoring. Top-Down Parsing - Recursive Descent parsing, Predictive Parsing, LL(1) Grammars.

Module - 3 (Bottom-Up Parsing)

Handle Pruning. Shift Reduce parsing. Operator precedence parsing (Concept only). LR parsing - Constructing SLR, LALR and canonical LR parsing tables.

Module - 4 (Syntax directed translation and Intermediate code generation)

Syntax directed translation - Syntax directed definitions, S-attributed definitions, L-attributed definitions, Bottom-up evaluation of S-attributed definitions. Run-Time Environments - Source Language issues, Storage organization, Storage-allocation strategies. Intermediate Code Generation - Intermediate languages, Graphical representations, Three-Address code, Quadruples, Triples.

Module 5 – (Code Optimization and Generation)

Code Optimization - Principal sources of optimization, Machine dependent and machine independent optimizations, Local and global optimizations. Code generation - Issues in the design of a code generator, Target Language, A simple code generator.

Text Books

1. Aho A.V., Ravi Sethi and D. Ullman. Compilers – Principles Techniques and Tools, Addison Wesley, 2006.

Reference Books

1. D.M.Dhamdhere, System Programming and Operating Systems, Tata McGraw Hill & Company, 1996.
2. Kenneth C. Louden, Compiler Construction – Principles and Practice, Cengage Learning Indian Edition, 2006.

3. Tremblay and Sorenson, The Theory and Practice of Compiler Writing, Tata McGraw Hill & Company,1984.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1) Explain the phases of a compiler with a neat diagram.

2) Define a token. Identify the tokens in the expression a := b + 10.

Course Outcome 2 (CO2):

1) Illustrate the process of eliminating ambiguity, left recursion and left factoring the grammar.

2) Is the following grammar ambiguous? If so eliminate ambiguity.

E→E + E | E*E |(E) | id

Course Outcome 3 (CO3):

1. What are the different parsing conflicts in the SLR parsing table?
2. Design a recursive descent parser for the grammar

   E→E + T | T

   T→T*F | F

   F→(E) | id
3. Construct canonical LR(0) collection of items for the grammar below.

   S → L = R

   S → R

   L → * R

   L → id

   R → L

Also identify a shift reduce conflict in the LR(0) collection constructed above.

Course Outcome 4 (CO4):

1. Write the quadruple and triple representation of the following intermediate code

   R1 = C * D

   R2 = B + R1

   A = R2

   B[0] = A

2. Differentiate S-attributed Syntax Directed Translation(SDT) and L-attributed SDT. Write S - attributed SDT for a simple desktop calculator

Course Outcome 5 (CO5):

1. List out the examples of function preserving transformations.
2. What are the actions performed by a simple code generator for a typical three-address statement of the form x: = y op z.

Model Question Paper

QP CODE:

Reg No: _____

Name: _____ **PAGES : 4**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SIXTH SEMESTER B.TECH DEGREE EXAMINATION , MONTH & YEAR

Course Code: CST 302

Course Name: Compiler Design

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

PART A

Answer All Questions. Each Question Carries 3 Marks

1. Specify the analysis and synthesis parts of compilation.
2. Define the terms token, lexemes and patterns with examples.
3. Is the grammar S --> S | (S) S / Ɛ ambiguous? Justify your answer.
4. What is left recursive grammar? Give an example. What are the steps in removing left recursion?
5. Compare different bottom-up parsing techniques.
6. What are the possible actions of a shift reduce parser.

7. Differentiate synthesized and inherited attributes with examples.

8. Translate a[i] = b * c – b * d, to quadruple.

9. What is the role of peephole optimization in the compilation process

10. What are the issues in the design of a code generator

(10x3=30)

Part B

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

11. (a) Explain the different phases of a compiler with a running example. **(9)**

(b) List and explain any three compiler construction tools. **(5)**

OR

12. (a) What is a regular definition? Give the regular definition of an unsigned integer **(7)**

(b) Express the role of transition diagrams in recognition of tokens. **(7)**

13. (a) What is Recursive Descent parsing? List the challenges in designing such a parser? **(4)**

(b) Consider the following grammar **(10)**

E-→E or T | T

T→ T and F |F

F→not F | (E) | true | false

(i) Remove left recursion from the grammar.

(ii) Construct a predictive parsing table.

(iii) Justify the statement "The grammar is LL (1)".

OR

14. (a) What is Recursive Descent parsing? List the problems in designing such a parser **(4)**

(b) Design a recursive descent parser for the grammar S→cAd, A→ab/ b **(5)**

Find the FIRST and FOLLOW of the non-terminals S, A and B in the grammar **(5)**

S→aABe
A→Abc | b
B→d

15. (a) Construct the LR(0) set of items and their GOTO function for the grammar **(10)**
S→S S + | S S * | a

(b) Is the grammar SLR? Justify your answer **(4)**

OR

16. (a) Identify LR(1) items for the grammar
S→ CC **(7)**
C→ cC | d

(b) Construct LALR table for the above grammar **(7)**

17. (a) Design a Syntax Directed Translator(SDT) for the arithmetic expression (4 * 7 + 19) * 2 and draw an annotated parse tree for the same. **(8)**

(b) Consider the grammar with following translation rules and E as the start symbol **(6)**

E → E1 # T {E.value=E1.value x T.value ;}
| T{E.value=T.value ;}
T → T1 & F{ T.value=T1.value + F.value ;}
| F{T.value= F.value ; }
F → num { F.value=num. lvalue ;}

Compute E.value for the root of the parse tree for the expression
2#3 & 5# 6 &7

OR

18. (a) Write Syntax Directed Translator (SDT) and parse tree for infix to postfix translation of an expression. **(8)**

    (b) Explain the storage allocation strategies. **(6)**

19. (a) Describe the principal sources of optimization **(7)**

    (b) Illustrate the optimization of basic blocks with examples. **(7)**

OR

20. (a) Write the Code Generation Algorithm and explain the *getreg* function **(6)**

    (b) Generate target code sequence for the following statement **(8)**

d := (a-b)+(a-c)+(a-c).

Teaching Plan

No	Contents	No. of Lecture Hours
Module - 1(Introduction to Compilers and lexical analyzer) (8 hours)		
1.1	Introduction to compilers, Analysis of the source program	1 hour
1.2	Phases of the compiler – Analysis Phases	1 hour
1.3	Phases of the Compiler - Synthesis Phases	1 hour
1.4	Symbol Table Manager and Error Handler	1 hour
1.5	Compiler writing tools, bootstrapping	1 hour
1.6	The role of Lexical Analyzer , Input Buffering	1 hour
1.7	Specification of Tokens	1 hour
1.8	Recognition of Tokens	1 hour

Module – 2 (Introduction to Syntax Analysis) (10 hours)		
2.1	Role of the Syntax Analyser, Syntax error handling	1 hour
2.2	Review of Context Free Grammars	1 hour
2.3	Parse Trees and Derivations	1 hour
2.4	Grammar transformations, Eliminating ambiguity	1 hour
2.5	Eliminating left recursion	1 hour
2.6	Left factoring the grammar	1 hour
2.7	Recursive Descent parsing	1 hour
2.8	First and Follow	1 hour
2.9	Predictive Parsing table constructor	1 hour
2.10	LL(1) Grammars	1 hour
Module - 3 (Bottom up parsing) (9 hours)		
3.1	Bottom-up parsing - Handle Pruning	1 hour
3.2	Shift Reduce parsing	1 hour
3.3	Operator precedence parsing (Concept only)	1 hour
3.4	LR parsing , SLR Grammar, items	1 hour
3.5	Augmented Grammar, Canonical collection of LR(0) items	1 hour
3.6	SLR Parser Table Construction	1 hour
3.7	Constructing Canonical LR Parsing Tables	1 hour
3.8	Constructing LALR Parsing Tables	1 hour
3.9	LALR parser	1 hour
Module - 4 (Syntax Directed Translation and Intermediate code Generation) (9 hours)		
4.1	Syntax directed definitions	1 hour
4.2	S- attributed definitions, L- attributed definitions	1 hour
4.3	Bottom- up evaluation of S- attributed definitions.	1 hour
4.4	Source Language issues	1 hour
4.5	Storage organization	1 hour

4.6	Storage- allocation strategies	1 hour
4.7	Intermediate languages , Graphical representations	1 hour
4.8	Three-Address code	1 hour
4.9	Quadruples, Triples	1 hour
Module - 5 (Code Optimization and Generation) (9 hours)		
5.1	Principal sources of optimization	1 hour
5.2	Machine dependent optimizations	1 hour
5.3	Machine independent optimizations	1 hour
5.4	Local optimizations	1 hour
5.5	Global optimizations	1 hour
5.6	Issues in the design of a code generator – Lecture 1	1 hour
5.7	Issues in the design of a code generator – Lecture 2	1 hour
5.8	Target Language	1 hour
5.9	Design of a simple code generator.	1 hour

CST 304	COMPUTER GRAPHICS AND IMAGE PROCESSING	Category	L	T	P	Credit	Year of Introduction
		PCC	3	1	0	4	2019

Preamble:
The purpose of this course is to make awareness about strong theoretical relationships between computer graphics and image processing. This course helps the learner to understand three-dimensional environment representation in a computer, transformation of 2D/3D objects, basic mathematical techniques and algorithms used to build useful applications, imaging, and image processing techniques. The study of computer graphics and image processing develops the ability to create image processing frameworks for different domains and develops algorithms for emerging display technologies.

Prerequisite: A sound knowledge of Mathematics and a 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, transformations on 2D & 3D objects, clipping algorithms and projection algorithms(Cognitive Knowledge level: Apply)
CO4	Summarize visible surface detection methods(Cognitive Knowledge level: Understand)
CO5	Summarize the concepts of digital image representation, processing and demonstrate pixel relationships(Cognitive Knowledge level: Apply)
CO6	Solve image enhancement and segmentation problems using spatial domain techniques(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	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 and Algorithms)

Basics of Computer Graphics and its applications. Video Display devices- Refresh Cathode Ray Tubes, Random Scan Displays and systems, Raster scan displays and systems. Line drawing algorithms- DDA, Bresenham's algorithm. Circle drawing algorithms- Midpoint Circle generation algorithm, Bresenham's algorithm.

Module - 2(Filled Area Primitives and transformations)

Filled Area Primitives- Scan line polygon filling, Boundary filling and flood filling. Two dimensional transformations-Translation, Rotation, Scaling, Reflection and Shearing, Composite transformations, Matrix representations and homogeneous coordinates. Basic 3D transformations.

Module - 3 (Clipping and Projections)

Window to viewport transformation. Cohen Sutherland Line clipping algorithm. Sutherland Hodgeman Polygon clipping algorithm. Three dimensional viewing pipeline. Projections- Parallel and Perspective projections. Visible surface detection algorithms- Depth buffer algorithm, Scan line algorithm.

Module - 4 (Fundamentals of Digital Image Processing)

Introduction to Image processing and applications. Image as 2D data. Image representation in Gray scale, Binary and Colour images. Fundamental steps in image processing. Components of image processing system. Coordinate conventions. Sampling and quantization. Spatial and Gray Level Resolution. Basic relationship between pixels– neighbourhood, adjacency, connectivity. Fundamentals of spatial domain-convolution operation.

Module - 5 (Image Enhancement in Spatial Domain and Image Segmentation)

Basic gray level transformation functions - Log transformations, Power-Law transformations, Contrast stretching. Histogram equalization. Basics of spatial filtering - Smoothing spatial filter-Linear and nonlinear filters, and Sharpening spatial filters-Gradient and Laplacian.

Fundamentals of Image Segmentation. Thresholding - Basics of Intensity thresholding and Global Thresholding. Region based Approach - Region Growing, Region Splitting and Merging. Edge Detection - Edge Operators- Sobel and Prewitt.

Text Book

1. Donald Hearn and M. Pauline Baker, Computer Graphics, PHI, 2e, 1996
2. Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing. Pearson, 4e, 2017

References

1) William M. Newman and Robert F. Sproull, Principles of Interactive Computer Graphics. McGraw Hill, 2001

2) Zhigang Xiang and Roy Plastock, Computer Graphics (Schaum's outline Series), McGraw Hill, 2019.
3) David F. Rogers , Procedural Elements for Computer Graphics, Tata McGraw Hill,2001.
4) M. Sonka, V. Hlavac, and R. Boyle, Image Processing, Analysis, and Machine Vision, Thomson India Edition, 4e, 2017.

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 with end points(2,3) and (5,8) using Bresenham's line drawing algorithm.
2. Explain how the 4-connected area filling approach differs from 8- connected area filling in boundary filling algorithm

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 coordinate ABC is given as A(4,1), B(5,2) and C(4,3).
2. 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)

Course Outcome 4 (CO4):

1. Explain scan line algorithm for detecting visible surfaces in an object.

Course Outcome 5 (CO5):

1. Give an image representation model and describe how the representation changes in grayscale, binary and colour images.
2. Consider an image segment shown below.

3 1 2 1 (q)

2 2 0 2

1 2 1 1

(p) 1 0 1 2

(a) Let V={0,1} and compute the length of the shortest 4-,8- and m- path between p and q. If a particular path does not exist between these two points , explain why?

(b) Repeat for V={1,2}.

3. The spatial resolution of an image is given by 128 X 128.What is its storage requirements if it is represented by 64 gray levels?

Course Outcome 6 (CO6):

1. A skilled medical technician is charged with the job of inspecting a certain class of monochrome images generated by electronic microscope. To facilitate the inspection, the technician uses image processing aids. However when he examines the images he finds the following problems.
   (a) Presence of bright isolated dots that are not of interest.
   (b) Lack of sharpness
   (c) Poor contrast

   Identify the sequence of preprocessing steps that the technician may use to overcome the above mentioned problems and explain it.

2. A 4x4, 4 bits/pixel original image is given by

10	12	8	9
10	12	12	14
12	13	10	9
14	12	10	12

   (a) Apply histogram equalisation to the image by rounding the resulting image pixels to integers
   (b) Sketch the histogram of the original image and the histogram-equalised image.
3. You have Sobel operator and Laplacian operator for edge detection. Which operator will you select for edge detection in the case of noisy image? Explain.**(Assignment)**

Model Question Paper

QP CODE:

Reg No: _____

Name: _____ **PAGES : 4**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST 304

Course Name: Computer Graphics and Image Processing

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

PART A

Answer All Questions. Each Question Carries 3 Marks

1. Justify the approach of using integer arithmetic in Bresenham's line drawing algorithm.

2. Consider a raster system with a resolution of 1024*1024. What is 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. Show that two successive reflections about either of the coordinate axes is equivalent to a single rotation about the coordinate origin.

4. Determine a sequence of basic transformations that are equivalent to the x-direction shearing matrix.

5. Find the window to viewport normalization transformation with window lower left corner at (1,1) and upper right corner at (2,6).

6. Find the orthographic projection of a unit cube onto the x=0, y=0 and z=0 plane.

7. Define Sampling and Quantization of an image.

8. Give any three applications of digital image processing.

9. A captured image appears very dark because of wrong lens aperture setting. Describe an enhancement technique which is appropriate to enhance such an image.

10. Suggest an approach of thresholding that should be used in case of uniform illumination. **(10x3=30)**

Part B

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

11. (a) Write Midpoint circle drawing algorithm and use it to plot a circle with radius=20 and center is (50,30). **(10)**

(b) Draw the architecture of raster scan display systems and explain its working principle. **(4)**

OR

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

(b) Explain the working principle of color CRT monitors with suitable illustrations. **(4)**

13. (a) Compare boundary fill algorithm and flood fill algorithm. **(5)**

(b) Reflect a triangle ABC about the line 3x-4y+8=0. The position vector of the coordinate ABC is given as A(4,1), B(5,2) and C(4,3). **(9)**

OR

14. (a) Explain the need of using vanishing points in projections. **(4)**

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

15. (a) Describe Sutherland Hodegman polygon clipping algorithm and what are its **(7)**

limitations.

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

OR

16. (a) Describe Sutherland Hodegman polygon clipping algorithm and what are its limitations. **(7)**

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

17. (a) Explain the components of an image processing system with suitable diagram **(9)**

(b) Define Resolution of an image. Explain the spatial and gray level resolution of an image with an example. **(5)**

OR

18. (a) Define 4-adjacency, 8 adjacency and m-adjacency. Consider the image segment shown. **(7)**

	4	2	3	2 (q)
	3	3	1	3
	2	3	2	2
(p)	2	1	2	3

Let V={1,2} and compute the length of the shortest 4- ,8- and m- path between p and q. If a particular path does not exist between these two points, explain why?

(b) Using any one application, explain the steps involved in image processing. **(7)**

19. (a) A 5x5 image patch is shown below. Compute the value of the marked pixel if it is smoothened by a 3x3 average filter and median filter. **(4)**

$$f(m,n)=\begin{pmatrix} 0 & 1 & 2 & 3 & 2 \\ 5 & 6 & 7 & 8 & 4 \\ 4 & 3 & \textcircled{2} & 1 & 2 \\ 8 & 7 & 6 & 5 & 3 \\ 1 & 5 & 3 & 7 & 6 \end{pmatrix}$$

(b) Define Image segmentation and describe in detail method of edge and region based segmentation technique. **(10)**

OR

20. (a) Distinguish between smoothing and sharpening filters in terms of **(10)**

(i) Functionality
(ii) Types
(iii) Applications
(iv) Mask Coefficients

(b) Describe how an image is segmented using split and merge technique in association with the region adjacency graph. **(8)**

Teaching Plan

No	Contents	No of Lecture Hrs (45 hrs)
Module – 1 (Basics of Computer Graphics and Algorithms) (9 hrs)		
1.1	Basics of Computer Graphics and applications	1 hour
1.2	Refresh Cathode Ray Tubes	1 hour
1.3	Random Scan Displays and systems	1 hour
1.4	Raster scan displays and systems	1 hour
1.5	DDA Line drawing Algorithm	1 hour
1.6	Bresenham's line drawing algorithm	1 hour
1.7	Midpoint Circle generation algorithm	1 hour
1.8	Bresenham's Circle generation algorithm	1 hour
1.9	Illustration of line drawing and circle drawing algorithms	1 hour
Module - 2 (Filled Area Primitives and transformations) (9 hrs)		
2.1	Scan line polygon filling	1 hour
2.2	Boundary filling and flood filling	1 hour
2.3	Basic 2D transformations-Translation	1 hour

2.4	Basic 2D transformations- Rotation and Scaling	1 hour
2.5	Reflection and Shearing	1 hour
2.6	Composite transformations	1 hour
2.7	Matrix representations and homogeneous coordinates	1 hour
2.8	Basic 3D transformation-Translation and scaling	1 hour
2.9	Basic 3D transformation-Rotation	1 hour
Module - 3 (Clipping and Projections) (8 hrs)		
3.1	Window to viewport transformation	1 hour
3.2	Cohen Sutherland Line clipping algorithm	1 hour
3.3	Sutherland Hodgeman Polygon clipping algorithm	1 hour
3.4	Practice problems on Clipping algorithms	1 hour
3.5	Three dimensional viewing pipeline, Projections-Parallel projections	1 hour
3.6	Projections- Perspective projections	1 hour
3.7	Visible surface detection algorithms- Depth buffer algorithm	1 hour
3.8	Scan line visible surface detection algorithm	1 hour
Module - 4 (Fundamentals of Digital Image Processing) (8 hrs)		
4.1	Introduction to Image processing-Image as a 2D data, Image representation-Gray scale, Binary and Colour images.	1 hour
4.2	Fundamental steps in image processing and applications	1 hour
4.3	Components of image processing system	1 hour
4.4	Coordinate conventions, Sampling and quantization, Spatial and Gray Level Resolution	1 hour
4.5	Basic relationship between pixels – neighbourhood, adjacency, connectivity	1 hour
4.6	Illustration of basic relationship between pixels– neighbourhood,	1 hour

	adjacency, connectivity	
4.7	Fundamentals of spatial domain - Convolution operation	1 hour
4.8	Illustration of Convolution operation	1 hour
Module - 5 (Image Enhancement in spatial domain and Image Segmentation) (11 hrs)		
5.1	Basic gray level transformation functions- Log transformations.	1 hour
5.2	Power-Law transformations, Contrast stretching	1 hour
5.3	Histogram equalization	1 hour
5.4	Illustration of Histogram equalization	1 hour
5.5	Basics of spatial filtering, Smoothing spatial filter- Linear and nonlinear filters	1 hour
5.6	Sharpening spatial filtering-Gradient filter mask	1 hour
5.7	Sharpening spatial filtering-Laplacian filter mask	1 hour
5.8	Fundamentals of Image Segmentation, Basics of Intensity thresholding, Basic Global Thresholding	1 hour
5.9	Region Based Approach- Region Growing, Region Splitting and Merging	1 hour1
5.10	Basics of Edge Detection	1 hour
5.11	Sobel and Prewitt edge detection masks	1 hour

CST 306	ALGORITHM ANALYSIS AND DESIGN	Category	L	T	P	Credit	Year of Introduction
		PCC	3	1	0	4	2019

Preamble:

The course introduces students to the design of computer algorithms, as well as analysis of algorithms. Algorithm design and analysis provide the theoretical backbone of computer science and are a must in the daily work of the successful programmer. The goal of this course is to provide a solid background in the design and analysis of the major classes of algorithms. At the end of the course students will be able to develop their own versions for a given computational task and to compare and contrast their performance.

Prerequisite:

Strong Foundation in Mathematics, Programming in C, Data Structures and Graph Theory.

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

CO#	CO
CO1	Analyze any given algorithm and express its time and space complexities in asymptotic notations. **(Cognitive Level: Apply)**
CO2	Derive recurrence equations and solve it using Iteration, Recurrence Tree, Substitution and Master's Method to compute time complexity of algorithms. **(Cognitive Level: Apply)**
CO3	Illustrate Graph traversal algorithms & applications and Advanced Data structures like AVL trees and Disjoint set operations. **(Cognitive Level: Apply)**
CO4	Demonstrate Divide-and-conquer, Greedy Strategy, Dynamic programming, Branch-and Bound and Backtracking algorithm design techniques (**Cognitive Level: Apply)**
CO5	Classify a problem as computationally tractable or intractable, and discuss strategies to address intractability **(Cognitive Level: Understand)**
CO6	Identify the suitable design strategy to solve a given problem. **(Cognitive Level: Analyze)**

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 (%)
	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. 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 Algorithm Analysis)

Characteristics of Algorithms, Criteria for Analysing Algorithms, Time and Space Complexity - Best, Worst and Average Case Complexities, Asymptotic Notations - Big-Oh (O), Big- Omega (Ω), Big-Theta (Θ), Little-oh (o) and Little- Omega (ω) and their properties. Classifying functions by their asymptotic growth rate, Time and Space Complexity Calculation of simple algorithms.

Analysis of Recursive Algorithms: Recurrence Equations, Solving Recurrence Equations – Iteration Method, Recursion Tree Method, Substitution method and Master's Theorem (Proof not required).

Module–2 (Advanced Data Structures and Graph Algorithms)

Self Balancing Tree - AVL Trees (Insertion and deletion operations with all rotations in detail, algorithms not expected); Disjoint Sets- Disjoint set operations, Union and find algorithms.

DFS and BFS traversals - Analysis, Strongly Connected Components of a Directed graph, Topological Sorting.

Module–3 (Divide & Conquer and Greedy Strategy)

The Control Abstraction of Divide and Conquer- 2-way Merge sort, Strassen's Algorithm for Matrix Multiplication-Analysis. The Control Abstraction of Greedy Strategy- Fractional Knapsack Problem, Minimum Cost Spanning Tree Computation- Kruskal's Algorithms - Analysis, Single Source Shortest Path Algorithm - Dijkstra's Algorithm-Analysis.

Module-4 (Dynamic Programming, Back Tracking and Branch & Bound))

The Control Abstraction- The Optimality Principle- Matrix Chain Multiplication-Analysis, All Pairs Shortest Path Algorithm - Floyd-Warshall Algorithm-Analysis. The Control Abstraction of Back Tracking – The N Queen's Problem. Branch and Bound Algorithm for Travelling Salesman Problem.

Module-5 (Introduction to Complexity Theory)

Tractable and Intractable Problems, Complexity Classes – P, NP, NP- Hard and NP-Complete Classes- NP Completeness proof of Clique Problem and Vertex Cover Problem- Approximation algorithms- Bin Packing, Graph Coloring. Randomized Algorithms (Definitions of Monte Carlo and Las Vegas algorithms), Randomized version of Quick Sort algorithm with analysis.

Text Books

1. T.H.Cormen, C.E.Leiserson, R.L.Rivest, C. Stein, Introduction to Algorithms, 2nd Edition, Prentice-Hall India (2001)
2. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", 2nd Edition, Orient Longman Universities Press (2008)

3. Sara Baase and Allen Van Gelder —Computer Algorithms, Introduction to Design and Analysis, 3rd Edition, Pearson Education (2009)

Reference Books

1. Jon Kleinberg, Eva Tardos, "Algorithm Design", First Edition, Pearson (2005)
2. Robert Sedgewick, Kevin Wayne, "Algorithms",4th Edition Pearson (2011)
3. GIlles Brassard, Paul Brately, "Fundamentals of Algorithmics", Pearson (1996)
4. Steven S. Skiena, "The Algorithm Design Manual", 2nd Edition, Springer(2008)

Course Level Assessment Questions

Course Outcome 1 (CO1):

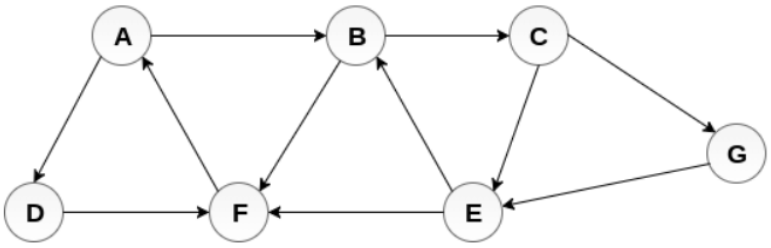
1. Is $2^{n+1} = O(2^n)$? Is $2^{2n} = O(2^n)$? Justify your answer.
2. What is the need of asymptotic analysis in calculating time complexity? What are the notations used for asymptotic analysis?
3. Calculate the time complexity for addition of two matrices.
4. Define time complexity and space complexity. Write an algorithm for adding n natural numbers and analyse the time and space requirements of the algorithm.

Course Outcome 2 (CO2):

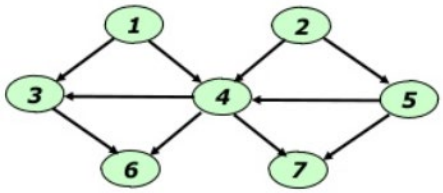
1. State Master's theorem for solving recurrences.
2. Solve the recurrence T(n) = 3T(n-2), using iteration method
3. State the conditions in recurrences where Master Theorem is not applicable.
4. Solve the following recurrence equations using Master's theorem.
   a) $T(n) = 8T(n/2) + 100\ n^2$
   b) $T(n) = 2T(n/2) + 10\ n$
5. Using Recursion Tree method, Solve T(n)= 2T(n/10)+ T(9n/10)+n. Assume constant time for small values of n.

Course Outcome 3 (CO3):

1. Explain the rotations performed for insertion in AVL tree with example.
2. Write down BFS algorithm and analyse the time complexity. Perform BFS traversal on the given graph starting from node A. If multiple node choices are available for next travel, choose the next node in alphabetical order.

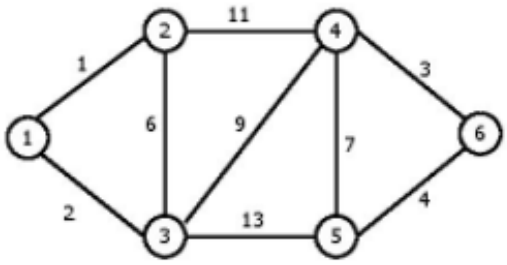


3. Find the minimum and maximum height of any AVL-tree with 7 nodes? Assume that the height of a tree with a single node is 0. (3)
4. Find any three topological orderings of the given graph.

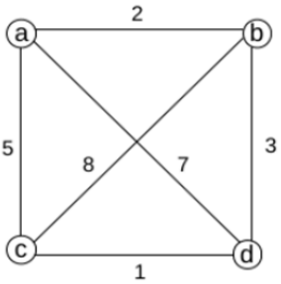


Course Outcome 4 (CO4):

1. Give the control abstraction for Divide and Conquer method.
2. Construct the minimum spanning tree for the given graph using Kruskal's algorithm. Analyse the complexity of the algorithm.



3. Compare Divide and Conquer and Dynamic programming methodologies
4. What is Principle of Optimality?
5. Define Travelling Salesman Problem (TSP). Apply branch and bound algorithm to solve TSP for the following graph, assuming the start city as 'a'. Draw the state space tree.



Course Outcome 5 (CO5):

1. Compare Tractable and Intractable Problems
2. With the help of suitable code sequence convince Vertex Cover Problem is an example of NP-Complete Problem

3. Explain Vertex Cover problem using an example. Suggest an algorithm for finding Vertex Cover of a graph.
4. Write short notes on approximation algorithms.
5. Compare Conventional quick sort algorithm and Randomized quicksort with the help of a suitable example?

Course Outcome 6 (CO6): (CO attainment through assignment only, not meant for examinations)

Choosing the best algorithm design strategy for a given problem after applying applicable design strategies – Sample Problems Given.

1. Finding the Smallest and Largest elements in an array of 'n' numbers
2. Fibonacci Sequence Generation.
3. Merge Sort
4. Travelling Sales Man Problem
5. 0/1 Knapsack Problem

Model Question Paper

QP CODE:

Reg No: _____

Name: _____ **PAGES : 4**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST 306

Course Name: Algorithm Analysis and Design

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

PART A

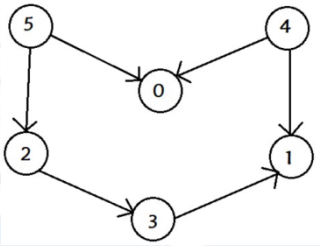
Answer All Questions. Each Question Carries 3 Marks

1. Define asymptotic notation? Arrange the following functions in increasing order of asymptotic growth rate.

   n^3, 2^n, $\log n^3$, 2^{100}, $n^2 \log n$, n^n, $\log n$, $n^{0.3}$, $2^{\log n}$

2. State Master's Theorem. Find the solution to the following recurrence equations using Master's theorem.
   a) $T(n) = 8T(n/2) + 100\ n^2$
   b) $T(n) = 2T(n/2) + 10\ n$

3. Find any two topological ordering of the DAG given below.



4. Show the UNION operation using linked list representation of disjoint sets.

5. Write the control abstraction of greedy strategy to solve a problem.

6. Write an algorithm based on divide-and-conquer strategy to search an element in a given list. Assume that the elements of list are in sorted order.

7. List the sequence of steps to be followed in Dynamic Programming approach.

8. Illustrate how optimal substructure property could be maintained in Floyd-Warshall algorithm.

9. Differentiate between P and NP problems.

10. Specify the relevance of approximation algorithms.

(10x3=30)

Part B

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

11. (a) Define Big O, Big Ω and Big Θ Notation and illustrate them graphically. **(7)**

    (b) Solve the following recurrence equation using recursion tree method **(7)**

    $T(n) = T(n/3) + T(2n/3) + n$, where $n>1$

    $T(n) = 1$, Otherwise

OR

12. (a) Explain the iteration method for solving recurrences and solve the following recurrence equation using iteration method. **(7)**

$$T(n) = 3T(n/3) + n;\ T(1) = 1$$

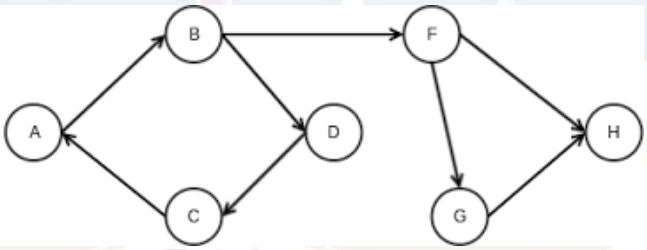
(b) Determine the time complexities of the following two functions fun1( ) and fun2( ). **(7)**

i)
```
int fun1(int n)
{
    if (n <= 1) return n;
    return 2*fun1(n-1);
}
```

ii)
```
int fun2 (int n)
{
    if (n <= 1) return n;
    return fun2 (n-1) + fun2 (n-1)
}
```

13. (a) Write DFS algorithm and analyse its time complexity. Illustrate the classification of edges in DFS traversal. **(7)**

(b) Find the strongly connected components of the digraph given below: **(7)**



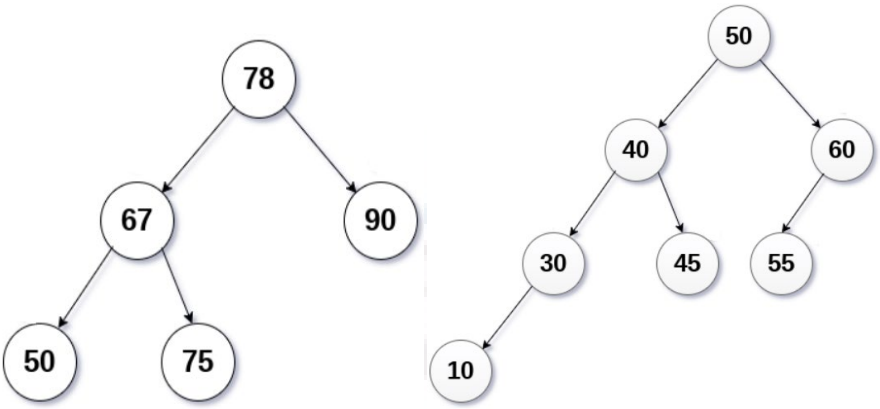
OR

14. (a) Illustrate the advantage of height balanced binary search trees over binary search trees? Explain various rotations in AVL trees with example. **(7)**

(b) Perform the following operations in the given AVL trees. **(7)**

i) Insert 70

ii) Delete 55



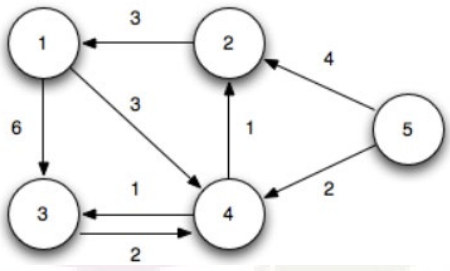
15. (a) State Fractional Knapsack Problem and write Greedy Algorithm for Fractional Knapsack Problem. **(7)**

(b) Find the optimal solution for the following Fractional Knapsack problem. Given the number of items(n) = 7, capacity of sack(m) = 15, W={2,3,5,7,1,4,1} and P = {10,5,15,7,6,18,3} **(7)**

OR

16. (a) Write and explain merge sort algorithm using divide and conquer strategy using the data {30, 19, 35, 3, 9, 46, 10}. Also analyse the time complexity. **(7)**

(b) Write the pseudo code for Dijkstra's algorithm. Compute the shortest distance from vertex 1 to all other vertices using Dijkstra's algorithm. **(7)**



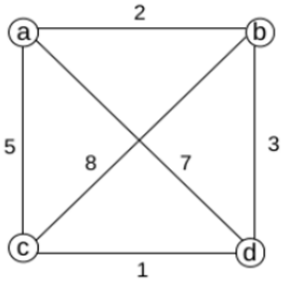
17. (a) Write Floyd-Warshall algorithm and analyse its complexity. **(5)**

(b) Write and explain the algorithm to find the optimal parenthesization of matrix chain product whose sequence of dimension is 4x10,10x3, 3x12,12x20. **(9)**

OR

18. (a) Explain the concept of Backtracking method using 4 Queens problem. **(7)**

(b) Define Travelling Salesman Problem (TSP). Apply branch and bound algorithm to solve TSP for the following graph, assuming the start city as 'a'. Draw the state space tree. **(7)**



19. (a) State bin packing problem? Explain the first fit decreasing strategy **(7)**

(b) Prove that the Clique problem is NP-Complete. **(7)**

OR

20. (a) Explain the need for randomized algorithms. Differentiate Las Vegas and Monte Carlo algorithms. **(6)**

(b) Explain randomized quicksort and analyse the expected running time of randomized quicksort with the help of a suitable example? **(9)**

Teaching Plan

No	Topic	No. of Hours (45 hrs)
Module -1 (Introduction to Algorithm Analysis) 9 hrs.		
1.1	Introduction to Algorithm Analysis: Characteristics of Algorithms.	1 hour
1.2	Criteria for Analysing Algorithms, Time and Space Complexity - Best, Worst and Average Case Complexities.	1 hour
1.3	Asymptotic Notations - Properties of Big-Oh (O), Big- Omega (Ω), Big-Theta (Θ), Little-Oh (o) and Little- Omega (ω).	1 hour
1.4	Illustration of Asymptotic Notations	1 hour

1.5	Classifying functions by their asymptotic growth rate	1 hour
1.6	Time and Space Complexity Calculation of algorithms/code segments.	1 hour
1.7	Analysis of Recursive Algorithms: Recurrence Equations, Solving Recurrence Equations – Iteration Method.	1 hour
1.8	Recursion Tree Method	1 hour
1.9	Substitution method and Master's Theorem and its Illustration.	1 hour
Module-2 (Advanced Data Structures and Graph Algorithms) 10 Hrs.		
2.1	Self Balancing Trees - Properties of AVL Trees, Rotations of AVL Trees	1 hour
2.2	AVL Trees Insertion and Illustration	1 hour
2.3	AVL Trees Deletion and Illustration	1 hour
2.4	Disjoint set operations.	1 hour
2.5	Union and find algorithms.	1 hour
2.6	Illustration of Union and find algorithms	1 hour
2.7	Graph Algorithms: BFS traversal, Analysis.	1 hour
2.8	DFS traversal, Analysis.	1 hour
2.9	Strongly connected components of a Directed graph.	1 hour
2.10	Topological Sorting.	1 hour
Module-3 (Divide & Conquer and Greedy Method) 8 Hrs		
3.1	Divide and Conquer: The Control Abstraction.	1 hour
3.2	2-way Merge Sort, Analysis.	1 hour
3.3	Strassen's Algorithm for Matrix Multiplication, Analysis	1 hour

3.4	Greedy Strategy: The Control Abstraction.	1 hour
3.5	Fractional Knapsack Problem.	1 hour
3.6	Minimum Cost Spanning Tree Computation- Kruskal's Algorithm, Analysis.	1 hour
3.7	Single Source Shortest Path Algorithm - Dijkstra's Algorithm	1 hour
3.8	Illustration of Dijkstra's Algorithm-Analysis.	1 hour
Module-4 (Dynamic Programming, Back Tracking and Branch and Bound) 8 Hrs.		
4.1	Dynamic Programming: The Control Abstraction, The Optimality Principle.	1 hour
4.2	Matrix Chain Multiplication-Analysis.	1 hour
4.3	Illustration of Matrix Chain Multiplication-Analysis.	1 hour
4.4	All Pairs Shortest Path Algorithm- Analysis and Illustration of Floyd-Warshall Algorithm.	1 hour
4.5	Back Tracking: The Control Abstraction .	1 hour
4.6	Back Tracking: The Control Abstraction – The N Queen's Problem.	1 hour
4.7	Branch and Bound:- Travelling salesman problem.	1 hour
4.8	Branch and Bound:- Travelling salesman problem.	1 hour
Module-5 (Introduction to Complexity Theory) 10 Hrs		
5.1	Introduction to Complexity Theory: Tractable and Intractable Problems.	1 hour
5.2	Complexity Classes – P, NP.	1 hour
5.3	NP- Hard and NP-Complete Problems.	1 hour
5.4	NP Completeness Proof of Clique Problem.	1 hour

5.5	NP Completeness Proof of Vertex Cover Problem.	1 hour
5.6	Approximation algorithms- Bin Packing Algorithm and Illustration.	1 hour
5.7	Graph Colouring Algorithm and Illustration.	1 hour
5.8	Randomized Algorithms (definitions of Monte Carlo and Las Vegas algorithms).	1 hour
5.9	Randomized Version of Quick Sort Algorithm with Analysis.	1 hour
5.10	Illustration of Randomized Version of Quick Sort Algorithm with Analysis.	1 hour

CST 308	COMPREHENSIVE COURSE WORK	Category	L	T	P	Credit	Year of Introduction
		PCC	1	0	0	1	2019

Preamble:

The objective of this Course work is to ensure the comprehensive knowledge of each student in the most fundamental core courses in the curriculum. Six core courses credited from Semesters 3, 4 and 5 are chosen for the detailed study in this course work. This course helps the learner to become competent in cracking GATE, placement tests and other competitive examinations

Prerequisite:

1. **Discrete Mathematical Structures**
2. **Data Structures**
3. **Operating Systems**
4. **Computer Organization And Architecture**
5. **Database Management Systems**
6. **Formal Languages And Automata Theory**

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

CO1	Comprehend the concepts of discrete mathematical structures (Cognitive Knowledge Level: **Understand)**
CO2 :	Comprehend the concepts and applications of data structures (Cognitive Knowledge Level: **Understand)**
CO3 :	Comprehend the concepts, functions and algorithms in Operating System (Cognitive Knowledge Level: **Understand)**)
CO4 :	Comprehend the organization and architecture of computer systems (Cognitive Knowledge Level: **Understand)**
CO5 :	Comprehend the fundamental principles of database design and manipulation (Cognitive Knowledge Level: **Understand)**
CO6 :	Comprehend the concepts in formal languages and automata theory 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	✓	✓										✓

Assessment Pattern

Bloom's Category	End Semester Examination
Remember	**10**
Understand	**20**
Apply	**20**
Analyse	
Evaluate	
Create	

Mark distribution

Total Marks	CIE	ESE	ESE Duration
50	**0**	**50**	**1 hour**

End Semester Examination Pattern: Objective Questions with multiple choice (Four). Question paper include fifty questions of one mark each covering the five identified courses.

Syllabus

Full Syllabus of all six selected Courses.

1. **Discrete Mathematical Structures**
2. **Data Structures**
3. **Operating Systems**
4. **Computer Organization And Architecture**
5. **Database Management Systems**
6. **Formal Languages And Automata Theory**

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**DISCRETE MATHEMATICAL STRUCTURES (14 hours)**	
1.1	Mock Test on Module 1 and Module 2	1 hour
1.2	Mock Test on Module 3, Module 4 and Module 5	1 hour
2	**DATA STRUCTURES**	
2.1	Mock Test on Module 1, Module 2 and Module 3	1 hour
2.2	Mock Test on Module 4 and Module 5	1 hour
3	**OPERATING SYSTEMS**	
3.1	Mock Test on Module 1 and Module 2	1 hour
3.2	Mock Test on Module 3, Module 4 and Module 5	1 hour
3.3	Feedback and Remedial	1 hour
4	**COMPUTER ORGANIZATION AND ARCHITECTURE**	
4.1	Mock Test on Module 1, Module 2 and Module 3	1 hour
4.2	Mock Test on Module 4 and Module 5	1 hour
5	**DATABASE MANAGEMENT SYSTEMS**	

5.1	Mock Test on Module 1, Module 2 and Module 3	1 hour
5.2	Mock Test on Module 4 and Module 5	1 hour
6	**FORMAL LANGUAGES AND AUTOMATA THEORY**	
6.1	Mock Test on Module 1, Module 2 and Module 3	1 hour
6.2	Mock Test on Module 4 and Module 5	1 hour
6.3	Feedback and Remedial	1 hour

Model Question Paper

QP CODE:

Reg No: _____

Name: _____ **PAGES : 10**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST 308

Course Name: Comprehensive Course Work

Max. Marks: 50 **Duration: 1 Hour**

Objective type questions with multiple choices. Mark one correct answer for each question. Each Question Carries 1 Mark

1. What is the maximum possible number of relations from a set with 5 elements to another set with 4 elements?

   (A) 2^10 (B)2^16 (C)2^20 (D)2^25

2. The set {1,2,4,7,8,11,13,14} is a group under multiplication modulo 15. Find the inverse of element 13

   (A) 7 (B) 13 (C) 1 (D) 8

3. Consider the recurrence relation $a_1 = 2$, $a_n = 3n+a_{n-1}$ Then a_{72} is

(A) 7882 (B) 7883 (C) 7884 (D) 7885

4. Which among the following is a contradiction?
   (A) $(p \wedge q) \vee \neg(p \vee q)$ (B) $(p \vee q) \wedge \neg(p \wedge q)$
   (C) $(p \wedge q) \wedge \neg(p \vee q)$ (D) $(p \wedge q) \vee (p \wedge \neg q)$

5. The number of non-negative solutions to $x + y + z = 18$, with conditions $x \geq 3, y \geq 2, z \geq 1 is$
   (A) 84 (B) 91 (C) 105 (D) 121

6. The solution of the recurrence relation $a_n = a_{n-1} + 2a_{n-2}$ with initial conditions $a_0 = 2, a_1 = 7$, is
   (A) $3(2)^n - (-1)^n$ (B) $3(2)^n + (-1)^n$
   (C) $-3(2)^n - (-1)^n$ (D) $-3(2)^n + (-1)^n$

7. Which among the following is not a subgroup of the set of Complex numbers under addition?
   (A) R, the set of all Real numbers.
   (B) Q^+, the set of positive rational numbers.
   (C) Z, the set of all integers.
   (D) The set iR of purely imaginary numbers including 0

8. Minimum number n of integers to be selected from $S = \{1,2,\ldots,9\}$ to guarantee that the difference of two of the n integers is 5 is
   (A) 3 (B) 4 (C) 6 (D) 9

9. Find the contrapositive the of statement "If it is a sunday, then I will wake up late"
   (A) If I am not waking up late, then it is a suniday
   (B) If I am not waking up late, then it is not a suniday
   (C) If it is not a sunday, then I will not wake up late.
   (D) It is not a sunday or I will wake up late

10. In the poset $(Z^+, |)$ (where Z^+ is the set of all positive integers and | is the divides relation), which of the following are false?
   I. 3 and 9 is comparable
   II. 7 and 10 is comparable
   III. The poset (Z+, |) is a total order
   (A) I and III (B) II only (C) II and III (D) III only

11. Consider the following sequence of operations on an empty stack.
   push(22); push(43); pop(); push(55); push(12); s=pop();

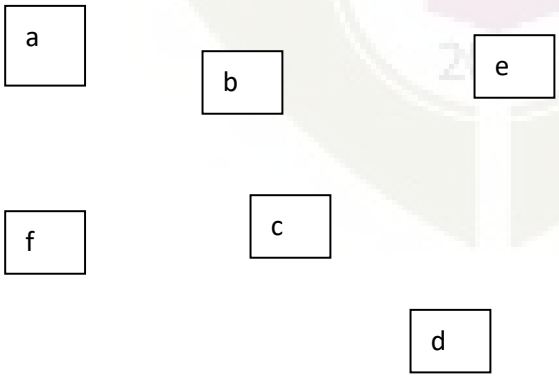
Consider the following sequence of operations on an empty queue.
enqueue(32);enqueue(27); dequeue(); enqueue(38); enqueue(12); q=dequeue();
The value of s+q is _____
(A) 44 (B) 54 (C) 39 (D) 70

12. The following postfix expression with single digit operands is evaluated using a stack:
8 2 2 ^ / 4 3 * + 5 1 * -
Note that ^ is the exponentiation operator. The top two elements of the stack after the first * is evaluated are:
(A) 12,2 (B) 12,5 (C) 2,12 (D) 2,5

13. Construct a binary search tree by inserting 8, 6, 12, 3, 10, 9 one after another. To make the resulting tree as AVL tree which of the following is required?
(A) One right rotation only
(B) One left rotation followed by two right rotations
(C) One left rotation and one right rotation
(D) The resulting tree itself is AVL

14. In a complete 4-ary tree, every internal node has exactly 4 children or no child. The number of leaves in such a tree with 6 internal nodes is:
(A) 20 (B) 18 (C) 19 (D) 17

15. Consider the following graph with the following sequences
I. a b c f d e
II. a b e d f c
III. a b f c d e
IV. a f c b e d



Which are Depth First Traversals of the above graph?

(A) I, II and IV only (B) I and IV only
(C) II, III and IV only (D) I, III and IV only

16. Consider a hash table of size seven, with starting index zero, and a hash function (2x + 5) mod7. Assuming the hash table is initially empty, which of the following is the contents of the table when the sequence 1, 4, 9, 6 is inserted into the table using closed hashing? Note that '_' denotes an empty location in the table.
(A) 9, _, 1, 6, _, _, 4 (B) 1, _, 6, 9, _, _, 4
(C) 4, _, 9, 6, _, _, 1 (D) 1, _, 9, 6, _, _, 4

17. Consider the following C program where TreeNode represents a node in a binary tree

```
struct TreeNode{
struct TreeNode *leftChild;
struct TreeNode *rightChild;
int element;
};
int CountNodes(struct TreeNode *t)
{
if((t==NULL)||((t->leftChild==NULL) && (t->rightChild==NULL)))
        return 0;
else
{
        return 1+CountNodes(t->leftChild)+CountNodes(t->rightChild)
}
}
```

The value returned by CountNodes when a pointer to the root of a binary tree is passed as its argument is
(A) number of nodes
(B) number of leaf nodes
(C) number of non leaf nodes
(D) number of leaf nodes-number of non leaf nodes

18. How many distinct binary search trees can be created out of 6 distinct keys?
(A) 7 (B) 36 (C) 140 (D) 132

19. Suppose a disk has 400 cylinders, numbered from 0 to 399. At some time the disk arm is at cylinder 58, and there is a queue of disk access requests for cylinder 66, 349, 201, 110, 38, 84, 226, 70, 86. If Shortest-Seek Time First (SSTF) is being used for scheduling the disk access, the request for cylinder 86 is serviced after servicing _____ number of

requests.
(A) 1 (B) 2 (C)3 (D)4

20. If frame size is 4KB then a paging system with page table entry of 2 bytes can address _____ bytes of physical memory.
(A) 2^12 (B) 2^16 (C) 2^18 (D) 2^28

21. Calculate the internal fragmentation if page size is 4KB and process size is 103KB.
(A) 3KB (B) 4KB (C) 1KB (D) 2KB

22. Which of the following scheduling policy is likely to improve interactiveness?
(A) FCFS (B) Round Robin
(C) Shortest Process Next (D) Priority Based Scgeduling

23. Consider the following program

Semaphore X=1, Y=0

```
Void A ( )
{
    While (1)
    {
    P(X);
    Print'1';
    V(Y);
    }
}
```

```
Void B ( )
{
    While (1)
    {
    P(Y);
    P(X);
    Print'0';
    V(X);
    }
}
```

The possible output of the program:
(A) Any number of 0's followed by any number of 1's.
(B) Any number of 1's followed by any number of 0's.
(C) 0 followed by deadlock
(D) 1 followed by deadlock

24. In a system using single processor, a new process arrives at the rate of 12 processes per minute and each such process requires 5 seconds of service time. What is the percentage of CPU utilization?
(A) 41.66 (B) 100.00 (C) 240.00 (D) 60.00

25. A system has two processes and three identical resources. Each process needs a maximum of two resources. This could cause
(A) Deadlock is possible (B) Deadlock is not possible

(C) Starvation may be present (D) Thrashing

26. Which of the following is true with regard to Round Robin scheduling technique?
    (A) Responds poorly to short process with small time quantum.
    (B) Works like SJF for larger time quantum
    (C) Does not use a prior knowledge of burst times of processes.
    (D) Ensure that the ready queue is always of the same size.

27. The size of the physical address space of a 32-bit processor is 2^W words. The capacity of cache memory is 2^N words. The size of each cache block is 2^K words. For a M-way set-associative cache memory, the length (in number of bits) of the tag field is
    (A) $W - N + \log_2 M$ (B) $W - N - \log_2 M$
    (C) $W - N - K - \log_2 M$ (D) $W - N - K + \log_2 M$

28. A 64-bit processor can support a maximum memory of 8 GB, where the memory is word-addressable (one word is of 64 bits). The size of the address bus of the processor is atleast ____ bits.
    (A) 30 (B) 31 (C) 32 (D) None

29. The stage delays in a 4-stage pipeline are 900, 450, 400 and 350 picoseconds. The first stage (with delay 900 picoseconds) is replaced with a functionally equivalent design involving two stages with respective delays 600 and 550 picoseconds. The throughput increase of the pipeline is _____ percent.
    (A) 38 (B) 30 (C) 58 (D) 50

30. Consider a direct mapped cache of size 256 Kilo words with block size 512 words. There are 6 bits in the tag. The number of bits in block (index) and word (offset) fields of physical address are is:
    (A) block (index) field = 6 bits, word (offset) field = 9 bits
    (B) block (index) field = 7 bits, word (offset) field = 8 bits
    (C) block (index) field = 9 bits, word (offset) field = 9 bits
    (D) block (index) field = 8 bits, word (offset) field = 8 bits

31. The memory unit of a computer has 1 Giga words of 64 bits each. The computer has instruction format, with 4 fields: an opcode field; a mode field to specify one of 12 addressing modes; a register address field to specify one of 48 registers; and a memory address field. If an instruction is 64 bits long, how large is the opcode field?
    (A) 34 bits (B) 24 bits (C) 20 bits (D) 14 bits

32. A computer has 64-bit instructions and 28-bit address. Suppose there are 252 two-address instructions. How many 1-address instructions can be formulated?

(A) 2^24 (B) 2^26 (C) 2^28 (D) 2^30

33. Determine the number of clock cycles required to process 200 tasks in a six-segment pipeline.(Assume there were no stalls),each segment takes 1 cycle.
(A) 1200 cycles (B) 206 cycles (C) 207 cycles (D) 205 cycles

34. Match the following Lists:

P.DMA	1.Priority Interrupt
Q. Processor status Word	2.I/O Transfer
R. Daisy chaining	3.CPU
S. Handshaking	4.Asynchronous Data Transfer

(A) P-1, Q-3, R-4, S-2 (B) P-2, Q-3, R-1, S-4
(C) P-2, Q-1, R-3, S-4 (D) P-4, Q-3, R-1, S-2

35. Let E1, E2 and E3 be three entities in an E/R diagram with simple single-valued attributes. R1 and R2 are two relationships between E1 and E2, where R1 is one-to-many, R2 is many-to-many. R3 is another relationship between E2 and E3 which is many-to-many. R1, R2 and R3 do not have any attributes of their own. What is the minimum number of tables required to represent this situation in the relational model?
(A) 3 (B) 4 (C) 5 (D) 6

36. Identify the minimal key for relational scheme R(U, V, W, X, Y, Z) with functional dependencies F = {U → V, V → W, W → X, VX → Z}
(A) UV (B) UW (C) UX (D) UY

37. It is given that: “Every student need to register one course and each course registered by many students”, what is the cardinality of the relation say “Register” from the “Student” entity to the “Course” entity in the ER diagram to implement the given requirement.
(A) M:1 relationship (B) M:N relationship
(C) 1:1 relationship (D) option (B) or(C)

38. Consider the relation branch(branch_name, assets, branch_city)
SELECT DISTINCT T.branch_name FROM branch T, branch S WHERE T.assets > L.assets AND S.branch_city = "TVM" .
Finds the names of
(A) All branches that have greater assets than all branches located in TVM.
(B) All branches that have greater assets than some branch located in TVM.
(C) The branch that has the greatest asset in TVM.
(D) Any branch that has greater asset than any branch located in TVM.

39. Consider the following relation instance, where "A" is primary Key.

A1	A2	A3	A4
1	1	1	Null
5	2	5	1
9	5	13	5
13	13	9	15

Which one of the following can be a foreign key that refers to the same relation?
(A) A2 (B) A3 (C) A4 (D) ALL

40. A relation R(ABC) is having the tuples(1,2,1),(1,2,2),(1,3,1) and (2,3,2). Which of the following functional dependencies holds well?
(A) A → BC (B) AC → B (C) AB → C (D) BC → A

41. Consider a relation R with attributes A, B, C, D and E and functional dependencies A → BC, BC → E, E →DA. What is the highest normal form that the relation satisfies?
(A) BCNF (B) 3 NF (C) 2 NF (D) 1 NF

42. For the given schedule S, find out the conflict equivalent schedule.
S : r1(x); r2(Z) ; r3(X); r1(Z); r2(Y); r3(Y);W1(X); W2(Z); W3(Y); W2(Y)
(A) T1→T2→T3 (B) T2->T1->T3
(C) T3→T1→T2 (D) Not conflict serializable

43. Which of the following strings is in the language defined by the grammar:
S → aX
X → aX | bX | b
(A) aaaba (B) babab (C) aaaaa (D) ababb

44. Consider the regular expression (x+y)*xyx(x+y)* where Σ = (x,y). If L is the language represented by this regular expression, then what will be the minimum number of states in a DFA recognizing L ?
(A) 2 (B) 3 (C) 4 (D) 5

45. Which of the following cannot handle the same set of languages?
(A) Deterministic Finite Automata and Non-Deterministic Finite Automata
(B) Deterministic Push Down Automata and Non-Deterministic Push Down Automata
(C) All of these
(D) None of these

46. Consider L be a context-free language and M be a non-context-free language. Which among the following is TRUE?

(I) L will definitely pass the pumping lemma test for CFLs.
(II) M will definitely pass the pumping lemma test for CFLs.
(III) L will not definitely pass the pumping lemma test for CFLs.
(IV) M will not definitely pass the pumping lemma test for CFLs.
(V) L may or maynot pass the pumping lemma test for CFLs.
(VI) M may or maynot pass the pumping lemma test for CFLs.
(A) I, II (B) II, V (C) I, VI (D) IV, V

47. Which of the following problem(s) is/are decidable?
(I) Whether a CFG is empty or not.
(II) Whether a CFG generates all possible strings.
(III) Whether the language generated by a Turing Machine is regular.
(IV) Whether the language generated by DFA and NFA are same.
(A) I and II (B) II and III (C) II and IV (D) I and IV

48. Which of the following is/are TRUE?
(I) Regular languages are closed under complementation.
(II) Recursive languages are closed under complementation.
(III) Context free languages are closed under complementation.
(IV) Context free languages are not closed under complementation.
(A) I, II and III (B) I, II and IV (C) II and III (D) III only

49. Which of the following regular expressions defined over the alphabet $\Sigma = \{0,1\}$ defines the language of all strings of length l where l is a multiple of 3?
(A) (0 + 1 + 00 + 11 + 000 +111)* (B) (000 + 111)*
(C) ((0 + 1)(0 + 1)(0 + 1))* (D) ((000 + 01 + 1)(111 + 10 + 0))*

50. Determine the minimum number of states of a DFA that recognizes the language over the alphabet {a,b} consisting of all the strings that contain at least three a's and at least four b's.
(A) 6 (B) 12 (C) 15 (D) 20

ANSWER KEY:-

QNo	Ans. Key	QNo	Ans. Key	QNo	Ans. Key	QNo	Ans. Key	QNo	Ans. Key
1	(C)	11	(C)	21	(C)	31	(B)	41	(A)

2	(A)	12	(A)	22	(B)	32	(D)	42	(D)
3	(B)	13	(A)	23	(D)	33	(D)	43	(D)
4	(C)	14	(C)	24	(B)	34	(B)	44	(C)
5	(B)	15	(A)	25	(B)	35	(C)	45	(B)
6	(A)	16	(D)	26	(C)	36	(D)	46	(C)
7	(B)	17	(C)	27	(A)	37	(A)	47	(D)
8	(C)	18	(D)	28	(A)	38	(B)	48	(B)
9	(B)	19	(C)	29	(D)	39	(B)	49	(C)
10	(C)	20	(D)	30	(C)	40	(D)	50	(D)

CSL332	NETWORKING LAB	CATEGORY	L	T	P	Credit	Year of Introduction
		PCC	0	0	3	2	2

Preamble:

The course enables the learners to get hands-on experience in network programming using Linux System calls and network monitoring tools. It covers implementation of network protocols and algorithms, configuration of network services and familiarization of network simulators. This helps the learners to develop, implement protocols and evaluate its performance for real world networks.

Prerequisite: Sound knowledge in Programming in C, Data Structures and Computer Networks

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

CO#	Course Outcomes
CO1	Use network related commands and configuration files in Linux Operating System. **(Cognitive Knowledge Level: Understand).**
CO2	Develop network application programs and protocols. **(Cognitive Knowledge Level: Apply)**
CO3	Analyze network traffic using network monitoring tools. **(Cognitive Knowledge Level: Apply)**
CO4	Design and setup a network and configure different network protocols. **(Cognitive Knowledge Level: Apply)**
CO5	Develop simulation of fundamental network concepts using a network simulator. **(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 teamwork
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 Test (Internal Exam) Marks in percentage**	**End Semester Examination Marks in percentage**
Remember	**20**	**20**
Understand	**20**	**20**
Apply	**60**	**60**
Analyze		
Evaluate		
Create		

Mark Distribution

Total Marks	**CIE Marks**	**ESE Marks**	**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 Internal examination shall be conducted for 100 marks, which will be converted to out of 15, while calculating Internal Evaluation marks. The marks will be distributed as, Algorithm - 30 marks, Program - 20 marks, Output - 20 marks and Viva - 30 marks.

End Semester Examination Pattern:

The End Semester Examination will be conducted for a total of 75 marks and shall be distributed as, Algorithm - 30 marks, Program - 20 marks, Output - 20 marks and Viva- 30 marks.

Operating System to Use in Lab	: Linux
Compiler/Software to Use in Lab	: gcc, NS2
Programming Language to Use in Lab	: Ansi C

Fair Lab Record:

All the students attending the Networking Lab should have a Fair Record. 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 the experiment, procedure/algorithm followed, other such details of the experiment and final result. The left hand page should contain a print out of the respective code with sample input and corresponding output obtained. All the experiments noted in the fair record should be verified by the faculty regularly. The fair record, properly certified by the faculty, should be produced during the time of End Semester Examination for the verification by the examiners.

Syllabus

***Mandatory**

(Note: At least one program from each topic in the syllabus should be completed in the Lab)

1. Getting started with the basics of network configuration files and networking commands in Linux.*
2. To familiarize and understand the use and functioning of system calls used for network programming in Linux.*
3. Implement client-server communication using socket programming and TCP as transport layer protocol*
4. Implement client-server communication using socket programming and UDP as transport layer protocol*
5. Simulate sliding window flow control protocols.* (Stop and Wait, Go back N, Selective Repeat ARQ protocols)
6. Implement and simulate algorithm for Distance Vector Routing protocol or Link State Routing protocol.*
7. Implement Simple Mail Transfer Protocol.
8. Implement File Transfer Protocol.*
9. Implement congestion control using a leaky bucket algorithm.*
10. Understanding the Wireshark tool.*
11. Design and configure a network with multiple subnets with wired and wireless LANs using required network devices. Configure commonly used services in the network.*
12. Study of NS2 simulator*

Networking Lab-Practice Questions

1. a) View the configuration, including addresses of your computers network interfaces.
   b) Test the network connectivity between your computer and several other computers.
   c) View the active TCP connections in the computer after visiting a website.
   d) Find the hardware/MAC address of another computer in the network using ARP.
2. Write the system calls used for creating sockets and transferring data between two nodes.
3. a) Implement a multi-user chat server using TCP as transport layer protocol.
   b) Implement a simple web proxy server that accepts HTTP requests and forwarding to remote servers and returning data to the client using TCP

4. Implement a Concurrent Time Server application using UDP to execute the program at a remote server. Client sends a time request to the server, server sends its system time back to the client. Client displays the result.
5. a) Implement Stop-and-Wait ARQ flow control protocol.

   b) Implement Go-Back--N ARQ flow control protocol.

   c) Implement Selective Repeat ARQ flow control protocol.
6. Implement Distance Vector Routing algorithm or Link State Routing algorithm..
7. Implement Simple Mail Transfer Protocol.
8. Develop a concurrent file server which will provide the file requested by a client if it exists. If not, the server sends appropriate message to the client. Server should also send its process ID (PID) to clients for display along with the file or the message.
9. Implement leaky bucket algorithm for congestion control.
10. a) Using Wireshark, Capture packets transferred while browsing a selected website. Investigate the protocols used in each packet, the values of the header fields and the size
    of the packet.

    b) Using Wireshark, observe three way handshaking connection establishment, three way handshaking connection termination and Data transfer in client server communication using TCP.

    c) Explore at least the following features of Wireshark: filters, Flow graphs (TCP), statistics, and protocol hierarchies.
11. Design and configure a network (wired and wireless LANs) with multiple subnets using required network devices. Configure at least three of the following services in the network- TELNET, SSH, FTP server, Web server, File server, DHCP server and DNS server.
12. a) The network consists of TCP source node (n0) and destination node (n1) over an area size of 500m x 500m. Node (n0) uses Agent/TCP/Reno as the sending TCP agent and FTP traffic source. Node (n1) is the receiver of FTP transfers, and it uses Agent/TCP sink as its TCP-agent for the connection establishment. Run the simulation for 150 seconds and show the TCP window size in two static nodes scenario with any dynamic routing protocol. Run the script and analyze the output graph for the given scenario.

    b) Simulate the transmission of ping messages over a star network topology consisting of ‘n’ nodes and find the number of packets dropped due to congestion using NS2simulator.

    c) Simulate Link State Protocol or Distance Vector Routing protocol in NS2.

Reference Books:

1. W. Richard Stevens, Bill Fenner, Andy Rudoff, UNIX Network Programming: Volume 1, The Sockets Networking API, 3rd Edition, Pearson, 2015
2. Lisa Bock, Learn Wireshark: Confidently navigate the Wireshark interface and solve real-world networking problems, Packt Publishing, 2019
3. Teerawat Issariyakul, Ekram Hossain, Introduction to Network Simulator NS2,2nd Edition, Springer,2019

CSD 334	MINI PROJECT	Category	L	T	P	Credit	Year of Introduction
		PCC	0	0	3	2	2019

Preamble:

The objective of this course is to apply the fundamental concepts of Software Engineering principles for the effective development of an application/research project. This course helps the learners to practice the different steps to be followed in the software development process such as literature review and problem identification, preparation of Software Requirement Specification &Software Design Document (SDD), 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 any programming language and fundamental concepts of Software Engineering.

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 and get familiarized with software development processes **(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	ESE Duration
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 3 or 4 members should identify a topic of interest in consultation with a Faculty/Advisor. 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 the Head of the Department comprising HoD or 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. and get department approval. Register Online course/ Collect study materials.
3. Create Software Requirements Specification (SRS Document)
4. Create Software Design Document (SDD). This may include designs like,
    a. System Architecture Design
    b. Application Architecture Design
    c. GUI Design (Mockups)
    d. API Design
    e. Database Design
    f. Technology Stack
5. Create Test Plan, Test Scenarios and Test Cases (Test Case Document) & Traceability Matrix
6. Create a Project Plan (with Modules, Tasks, Resources, Time schedule) [May use any project management tool or excel for this] – Choose and follow agile or waterfall models.
7. Development
    a. Set coding standards
    b. Environment Setup
    c. Source Code Control Setup (Like Subversion(SVN), Git)
    d. Development
    e. Unit Testing
    f. Integration Testing
    g. Testing /Quality Assurance(QA)
        i. Functional Testing
        ii. Load Testing
        iii. Report Bugs
    h. Resolve Bugs & Retest

8. Deployment (of software from local development environment to a production environment)
9. Test Run & Get Results
10. 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 figuretitle 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 VI

PROGRAM ELECTIVE I

CST 312	FOUNDATIONS OF 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 mathematical foundations of Machine Learning concepts. This course covers Linear Algebra, Probability and Distributions. Concepts in this course help the learners to identify the inherent assumptions & limitations of the current methodologies and develop new Machine Learning solutions.

Prerequisite: A sound background in higher secondary school Mathematics.

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

CO 1	Illustrate operations and applications of linear equations, matrix algebra, vector spaces, eigen values & eigenvectors **(Cognitive Knowledge Level: Apply)**
CO 2	Illustrate the concepts of orthogonality & diagonalization. **(Cognitive Knowledge Level: Apply)**
CO 3	Solve computational problems using probability and random variables. **(Cognitive Knowledge Level: Apply)**
CO 4	Identify an appropriate probability distribution for a given discrete or continuous random variable and use its properties. (**Cognitive Knowledge Level: Apply)**
CO 5	Illustrate moment generating function, law of large numbers and central limit theorems (**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 (LINEAR ALGEBRA)

Systems of Linear Equations – Matrices, Solving Systems of Linear Equations. Vector Spaces - Linear Independence, Basis and Rank, Linear Mappings.

Module 2 (LINEAR ALGEBRA)

Norms - Inner Products, Lengths and Distances, Angles and Orthogonality. Orthonormal Basis, Orthogonal Complement, Orthogonal Projections. Matrix Decompositions - Eigenvalues and Eigenvectors, Eigen decomposition and Diagonalization.

Module 3 (PROBABILITY AND DISTRIBUTIONS)

Probability Space - Sample Spaces, Probability Measures, Computing Probabilities, Conditional Probability, Baye's Rule, Independence. Random Variables - Discrete Random Variables (Bernoulli Random Variables, Binomial Distribution, Geometric and Poisson Distribution, Continuous Random Variables (Exponential Density, Gamma Density, Normal Distribution, Beta Density)

Module 4 (RANDOM VARIABLES)

Functions of a Random Variable. Joint Distributions - Independent Random Variables, Conditional Distributions, Functions of Jointly Distributed Random Variables.

Expected Values - Expected Value of a Random Variable, Expectations of Functions of Random Variables, Expectations of Linear Combinations of Random Variables, Variance and Standard Deviation, Covariance and Correlation, Conditional Expectation

Module 5 (LIMIT THEOREMS)

Moment-Generating Function. Limit Theorems(Proof not expected) - Law of Large Numbers, Convergence in Distribution and the Central Limit Theorem. Distributions derived from the Normal Distribution - Chi-square, t, and F Distributions, Sample Mean and the Sample Variance.

Text book:

1. Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong, Mathematics for Machine Learning, Cambridge University Press (freely available at https://mml – book.github.io)
2. John A. Rice, Mathematical Statistics and Data Analysis, University of California, Berkeley, Third edition, published by Cengage.

Reference books:

1. Gilbert Strang, Linear Algebra and Its Applications, 4th Edition,
2. Axler, Sheldon, Linear Algebra Done Right, 2015 Springer
3. Stephen Boyd and Lieven Vandenberghe, Introduction to Applied Linear Algebra, 2018 published by Cambridge University Press

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Find the set ***S*** of all solutions in ***x*** of the following inhomogeneous linear systems ***Ax = b***, where ***A*** and ***b*** are defined as follows:

$$A = \begin{bmatrix} 1 & -1 & 0 & 0 & 1 \\ 1 & 1 & 0 & -3 & 0 \\ 2 & -1 & 0 & 1 & -1 \\ -1 & 2 & 0 & -2 & -1 \end{bmatrix}, \quad b = \begin{bmatrix} 3 \\ 6 \\ 5 \\ -1 \end{bmatrix}$$

2. Determine the inverses of the following matrix if possible

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

3. Are the following sets of vectors linearl independent?

$$x_1 = \begin{bmatrix} 2 \\ -1 \\ 3 \end{bmatrix}, \quad x_2 = \begin{bmatrix} 1 \\ 1 \\ -2 \end{bmatrix}, \quad x_3 = \begin{bmatrix} 3 \\ -3 \\ 8 \end{bmatrix}$$

4. A set of ***n*** linearly independent vectors in R^n forms a basis. Does the set of vectors **(2, 4,−3) (0, 1, 1)** , **(0, 1,−1**) form a basis for R^3? Explain your reasons.

Course Outcome 2 (CO2):

1. Determine which of the following sets are orthogonal sets.

$$\left\{\begin{bmatrix}3\\1\\1\end{bmatrix},\begin{bmatrix}-1\\2\\1\end{bmatrix},\begin{bmatrix}-1/2\\-2\\7/2\end{bmatrix}\right\} \qquad \left\{\begin{bmatrix}1\\-1\\1\end{bmatrix},\begin{bmatrix}2\\1\\-1\end{bmatrix},\begin{bmatrix}3\\0\\-3\end{bmatrix}\right\} \qquad \left\{\begin{bmatrix}3\\-2\\1\\3\end{bmatrix},\begin{bmatrix}-1\\3\\-3\\4\end{bmatrix},\begin{bmatrix}3\\8\\7\\0\end{bmatrix}\right\}$$

2. Find the characteristic equation, eigenvalues, and eigenspaces corresponding to each eigenvalue of the following matrix.

$$\begin{bmatrix}2&0&4\\0&3&0\\0&1&2\end{bmatrix}$$

3. Diagonalize the following matrix, if possible

$$\begin{bmatrix}3&0&0&0\\0&2&0&0\\0&0&2&0\\1&0&0&3\end{bmatrix}$$

Course Outcome 2 (CO3):

1. Let ***J*** and ***T*** be independent events, where ***P(J)=0.4*** and ***P(T)=0.7***.
   i. Find ***P(J∩T)***
   ii. Find ***P(J∪T)***
   iii. Find ***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

i. Given that ***E(R)=2.85***, find ***a*** and ***b***.
ii. Find ***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^{th} toss.

Course Outcome- 3 (CO4):

1. An urn contains **p** black balls, **q** white balls, and **r** red balls; and **n** balls are chosen without replacement.
   a. Find the joint distribution of the numbers of black, white, and red balls in the sample.
   b. Find the joint distribution of the numbers of black and white balls in the sample.
   c. Find the marginal distribution of the number of white balls in the sample.
2. Suppose that two components have independent exponentially distributed lifetimes, T_1 and T_2, with parameters α and β, respectively. Find (a) $P(T_1 > T_2)$ and (b) $P(T_1 > 2T_2)$.
3. Let Z_1 and Z_2 be independent random variables each having the standard normal distribution. Define the random variables X and Y by $X = Z_1 + 3Z_2$ and $Y = Z_1 + Z_2$. Argue that the joint distribution of (X, Y) is a bivariate normal distribution. What are the parameters of this distribution?

4. Given a continuous random variable x, with cumulative distribution function $F_x(x)$, show that the random variable $y = F_x(x)$ is uniformly distributed.
5. You 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?
6. Let X be a continuous random variable with the density function $f(x) = 2x,\ 0 \leq x \leq 1$
   a. Find $E(X)$.
   b. Find $E(X^2)$ and $Var(X)$.

Course Outcome 5 (CO5):

1. Find the moment-generating function of a Bernoulli random variable, and use it to find the mean, variance, and third moment.
2. Use moment-generating functions to show that if **X** and **Y** are independent, then $Var(aX + bY) = a^2Var(X) + b^2Var(Y)$.
3. Suppose that you bet Rs 5 on each of a sequence of 50 independent fair games. Use the central limit theorem to approximate the probability that you will lose more than Rs 75.
4. Suppose that the number of insurance claims, N, filed in a year is Poisson distributed with $E(N) = 10,000$. Use the normal approximation to the Poisson to approximate $P(N > 10,200)$.

Model Question paper

QP Code : **Total Pages: 4**

Reg No.:_____ Name:_____

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SIXTH SEMESTER B.TECH DEGREE EXAMINATION (ELECTIVE), MONTH and YEAR

Course Code: CST 312

Course Name: FOUNDATIONS OF MACHINE LEARNING

Max. Marks: 100 Duration: 3 Hours

PART A

Answer all questions, each carries 3 marks.

1 Show that with the usual operation of scalar multiplication but with addition on reals given by $x \# y = 2(x + y)$ is not a vector space.

2 Are the following vectors linearly independent? Justify your answer.

$$x_1 = \begin{bmatrix} 2 \\ -1 \\ 3 \end{bmatrix}, \quad x_2 = \begin{bmatrix} 1 \\ 1 \\ -2 \end{bmatrix}, \quad x_3 = \begin{bmatrix} 3 \\ -3 \\ 8 \end{bmatrix}$$

3 Find the eigenvalues of the following matrix in terms of k. Can you find an eigenvector corresponding to each of the eigenvalues?

$$\begin{bmatrix} 1 & k \\ 2 & 1 \end{bmatrix}$$

4 Find a unit vector in $\mathbf{R^2}$ that is orthogonal to **(-1 , 2)**.

5 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?

6 Show that if two events ***A*** and ***B*** are independent, then ***A*** and ***B'*** are independent.

7 Prove that ***X*** and ***Y*** are independent if and only if $f_{X|Y}(x|y) = f_X(x)$ for all ***x*** and ***y***.

8 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)***.

9 Compare the Poisson cdf and the normal approximation for (a) $\lambda = 10$, (b) $\lambda = 20$, and (c) $\lambda = 40$.

10 State law of large numbers. **10 x 3 = 30**

PART B

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

11 a) Find all solutions to the system of linear equations **(8)**

$$\begin{aligned} -4x + 5z &= -2 \\ -3x - 3y + 5z &= 3 \\ -x + 2y + 2z &= -1 \end{aligned}$$

b) Consider the transformation $T(x, y) = (x + y, x + 2y, 2x + 3y)$. Obtain **ker T** and use this to calculate the nullity. Also find the transformation matrix for **T**. **(6)**

OR

12 a) Consider the following linear mapping **(8)**

$$\Phi : \mathbb{R}^3 \to \mathbb{R}^4$$

$$\Phi\left(\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}\right) = \begin{bmatrix} 3x_1 + 2x_2 + x_3 \\ x_1 + x_2 + x_3 \\ x_1 - 3x_2 \\ 2x_1 + 3x_2 + x_3 \end{bmatrix}$$

i. Find the transformation matrix T.

ii. Determine rank(T).

iii. Compute the kernel and image of the mapping and find their dimension

b) Prove that all vectors orthogonal to $[2, -3, 1]^T$ forms a subspace W of R^3. What is $dim\ (W)$ and why? **(6)**

13 a) Find an orthonormal basis of R^3 consisting of eigenvectors for the following matrix **(8)**

$$\begin{bmatrix} 1 & 0 & -2 \\ 0 & 5 & 0 \\ -2 & 0 & 4 \end{bmatrix}$$

b) Find a 3 × 3 orthogonal matrix S and a 3 × 3 diagonal matrix D such that $A = SDS^T$ **(6)**

OR

14 a) Find an orthogonal basis for the subspace of R^4 spanned by $\{ w_1 = (1, 1, 3, 2), w_2 = (1, -2, 0, -1), w_3 = (0, 2, 1, 2) \}$. **(8)**

b) Find the characteristic equation, eigenvalues, and eigenspaces corresponding tc each eigenvalue of the following matrix **(6)**

$$\begin{bmatrix} 2 & 0 & 4 \\ 0 & 3 & 0 \\ 0 & 1 & 2 \end{bmatrix}$$

15 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) 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.

OR

16 a) A factory runs three shifts. On a given day, 1% of the items produced by the first shift are defective, 2% of the second shift's items are defective, and 5% of the third shift's items are defective. If the shifts all have the same productivity, what percentage of the items produced in a day are defective? If an item is defective, what is the probability that it was produced by the third shift? **(8)**

b) Show that if A and B are two independent events, then $P(A \cup B) = P(A) + P(B) - P(A)P(B)$ **(6)**

17 a) 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. **(8)**

b) Let X be a discrete random variable that takes on values 0, 1, 2 with probabilities 1/2 , 3/8 , 1/8 , respectively. **(6)**

i. Find **E(X)** and **Var(X)**.

ii. Let $Y = X^2$. Find the probability mass function of Y and use it to find **E(Y)**.

18 a) A random square has a side length that is a uniform [0, 1] random variable. Find the expected area of the square. **(7)**

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)**

19 a) Using the fact that the mean of the chi-squared distribution is $(n-1)$, prove that $E(S^2) = \sigma^2$. **(7)**

b) i. Random samples of size 36 are taken from an infinite population whose mean is 80 and standard deviation is 18. Find the mean and standard error of the **(7)**

sampling distribution.

ii. Why is the Central Limit Theorem so important to statistical analysis?

OR

20 a) A six-sided die is rolled 100 times. Using the normal approximation, find the probability that the face showing a six turns up between 15 and 20 times. Find the probability that the sum of the face values of the 100 trials is less than 300. **(8)**

b) Determine an interval **(a, b)** such that $P[a \leq t \leq b\ ] = 0.80$, and that 10% of the area is on each side of a and b, assuming that the sample is of size 21. **(6)**

Teaching Plan		
No	**Topic**	**No. of Lectures (35)**
1	**Module-1 (LINEAR ALGEBRA) TB-1(Ch 2,3,4) (6 hours)**	
1.1	Systems of Linear Equations – Matrices, Solving Systems of Linear Equations.	1 hour
1.2	Vector Spaces, sub space	1 hour
1.3	Linear Independence,	1 hour
1.4	Basis and Rank	1 hour
1.5.	Linear Mappings- Kernel, Range	1 hour
1.6.	Linear Mappings- Rank, Nullity	
2	**Module-2 (LINEAR ALGEBRA) (6 hours)**	
2.1.	Norms, Inner Products, Lengths and Distances, Angles and Orthogonality,	1 hour
2.2	Orthonormal Basis, Orthogonal Complement,	1 hour
2.3	Orthogonal Projections	1 hour
2.4.	Eigenvalues and Eigenvectors	1 hour
2.5.	Eigen decomposition	1 hour
2.6.	Eigen Diagonalization	1 hour
3.	**Module-3 (PROBABILITY AND DISTRIBUTIONS) TB-2(Ch 1,2) (9 hours)**	

3.1	Sample Spaces, Probability Measures, Computing Probabilities	1 hour
3.2	Conditional Probability,	1 hour
3.3	Baye's Rule	1 hour
3.4	Independence of events	1 hour
3.5	Discrete Random Variables -Bernoulli Random Variables, Binomial Distribution	1 hour
3.6	Discrete Random Variables -Geometric Distribution	1 hour
3.7	Discrete Random Variables -Poisson Distribution	1 hour
3.8	Continuous Random Variables - Exponential Density, Gamma Density,	1 hour
3.9	Continuous Random Variables - Normal Distribution, Beta Density	1 hour
4.	**Module-4 (RANDOM VARIABLES) TB-2 (Ch 3, 4, 5, 6) (9 hours)**	
4.1	Functions of a Random Variable	1 hour
4.2	Joint Distributions - Independent Random Variables	1 hour
4.3	Conditional Distributions	1 hour
4.4	Functions of Jointly Distributed Random Variables	1 hour
4.5	Expected Value of a Random Variable,	1 hour
4.6	Expectations of Functions of Random Variables,	1 hour
4.7	Expectations of Linear Combinations of Random Variables	1 hour
4.6	Variance and Standard Deviation	1 hour
4.9	Covariance and Correlation	1 hour

5	Module-5 (LIMIT THEOREMS) (6 hours)	
5.1	Conditional Expectation,	1 hour
5.2	Moment-Generating Function	1 hour
5.3	Limit Theorems(Proof not expected) - Law of Large Numbers,	1 hour
5.4	Convergence in Distribution and the Central Limit Theorem.	1 hour
5.5	Distributions derived from the Normal Distribution - Chi-square and, and F Distributions,	1 hour
5.6	Distributions derived from the Normal Distribution - Sample Mean and the Sample Variance.	1 hour

CST 322	DATA ANALYTICS	Category	L	T	P	Credits	Year of Introduction
		PEC	2	1	0	3	2019

Preamble:

This course helps the learner to understand the basic concepts of data analytics. This course covers mathematics for data analytics, predictive and descriptive analytics of data, Big data and its applications, techniques for managing big data and data analysis & visualization using R programming tool. It enables the learners to perform data analysis on a real world scenario using appropriate tools.

Prerequisite: NIL

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

CO#	Course Outcomes
CO1	Illustrate the mathematical concepts for data analytics **(Cognitive Knowledge Level: Apply)**
CO2	Explain the basic concepts of data analytics **(Cognitive Knowledge Level: Understand)**
CO3	Illustrate various predictive and descriptive analytics algorithms (**Cognitive Knowledge Level: Apply)**
CO4	Describe the key concepts and applications of Big Data Analytics **(Cognitive Knowledge Level: Understand)**
CO5	Demonstrate the usage of Map Reduce paradigm for Big Data Analytics **(Cognitive Knowledge Level: Apply)**
CO6	Use R programming tool to perform data analysis and visualization **(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**	Lifelong 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

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 Series 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. 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 students should answer any one. Each question can have a maximum 2 sub-divisions and carries 14 marks.

Syllabus

Module – 1 (Mathematics for Data Analytics)

Descriptive statistics - Measures of central tendency and dispersion, Association of two variables - Discrete variables, Ordinal and Continuous variable, Probability calculus - probability distributions, Inductive statistics - Point estimation, Interval estimation, Hypothesis Testing - Basic definitions, t-test

Module - 2 (Introduction to Data Analytics)

Introduction to Data Analysis - Analytics, Analytics Process Model, Analytical Model Requirements. Data Analytics Life Cycle overview. Basics of data collection, sampling, preprocessing and dimensionality reduction

Module - 3 (Predictive and Descriptive Analytics)

Supervised Learning - Classification, Naive Bayes, KNN, Linear Regression. Unsupervised Learning - Clustering, Hierarchical algorithms – Agglomerative algorithm, Partitional algorithms - K- Means. Association Rule Mining - Apriori algorithm

Module - 4 (Big Data Analytics)

Big Data Overview – State of the practice in analytics, Example Applications - Credit Risk Modeling, Business Process Analytics.Big Data Analytics using Map Reduce and Apache Hadoop, Developing and Executing a HadoopMapReduce Program.

Module - 5 (R programming for Data Analysis)

Overview of modern data analytic tools.Data Analysis Using R - Introduction to R - R Graphical User Interfaces, Data Import and Export, Attribute and Data Types, Descriptive Statistics, Exploratory Data Analysis - Visualization Before Analysis, Dirty Data, Visualizing a Single Variable, Examining Multiple Variables, Data Exploration Versus Presentation, Statistical Methods for Evaluation

Text Book

1. Bart Baesens," Analytics in a Big Data World: The Essential Guide to Data Science and its Business Intelligence and Analytic Trends", John Wiley & Sons, 2013.
2. David Dietrich, "EMC Education Services, Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data", John Wiley & Sons, 2015.
3. Jaiwei Han, MichelineKamber, "Data Mining Concepts and Techniques", Elsevier, 2006.
4. Christian Heumann and Michael Schomaker, "Introduction to Statistics and DataAnalysis", Springer, 2016

References

1. Margaret H. Dunham, Data Mining: Introductory and Advanced Topics. Pearson, 2012.
2. Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer, 2007.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Explain the measures of central tendency.
2. Drive the mean and variance of normal distribution.
3. Collect sample data associated with a real world scenario, and identify central tendency and dispersion measures. Explain your inferences.

Course Outcome 2 (CO2):

1. Explain the life cycle of Data Analytics.
2. Discuss in detail the relevance of data sampling.

Course Outcome 3 (CO3):

1. The following table shows the midterm and final exam marks obtained for students in a database course.

X (Midterm exam)	Y (Final exam)
72	84
50	63

81	77
74	78
94	90
86	75
59	49
83	79
65	77
33	52
88	74
81	90

a) Use the method of least squares to find an equation for the prediction of a student's final exam marks based on the student's midterm grade in the course.
b) Predict the final exam marks of a student who received an 86 on the midterm exam.

2. Perform knn classification on the following dataset and predict the class for the data point X (P1 = 3, P2 =7), assuming the value of k as 3.

P1	P2	Class
7	7	False
7	4	False
3	4	True
1	4	True

Course Outcome 4 (CO4):

1. List down the characteristics of Big Data.
2. Illustrate process discovery task in business analytics using the scenario of insurance claim handling process. Draw the annotated process map.

Course Outcome 5 (CO5):

1. Explain how fault tolerance is achieved in HDFS.
2. Write down the pseudocode for Map and Reduce functions to solve any one data analytic problem.

Course Outcome 6 (CO6):

1. Illustrate any three R functions used in data analytics.
2. Explain the different categories of attributes and data types in R.

Model Question Paper

QP CODE:

Reg No:_____

Name :_____ **PAGES : 4**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST 322

Course Name: Data Analytics

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

PART A

(Answer all Questions. Each question carries 3 Marks)

1. Outline the errors that arise in hypothesis testing.

2. The number of members of a millionaires' club were as follows:

Year	2011	2012	2013	2014	2015	2016
Members	23	24	27	25	30	28

(a)What is the average growth rate of the membership?
(b)Based on the results of (a), how many members would one expect in 2018?

3. List and explain any two methods for dealing with missing values in a dataset.

4. Consider the following data (in increasing order) for the attribute age: 13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36, 40, 45, 46, 52, 70. Sketch an example for stratified sampling using samples of size 5 and the strata "youth," "middle-aged," and "senior."

5. Why is k nearest neighbor classifier called a lazy learner?

6. Find the absolute support, relative support and confidence of the rule (bread => jam) in the following set of transactions

   T1 {bread, butter}, T2{bread, jam, milk}

   T3{Milk, curd}, T4{bread, jam}

7. Explain the 3 Vs of Big Data.

8. Discuss the application of big data analytics in credit risk modeling.

9. Why is Exploratory Data Analysis important in business application ?

10. Explain how box plots be used for data summarization.

(10x3=30)

Part B

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

11. (a) Illustrate the Maximum Likelihood Estimation of Bernoulli distribution. **(8)**

(b) A hiking enthusiast has a new app for his smartphone which summarizes his hikes by using a GPS device. Let us look at the distance hiked (in km) and maximum altitude (in m) for the last 10 hikes: **(6)**

Distance	12.5	29.9	14.8	18.7	7.6	16.2	16.5	27.4	12.1	17.5
Altitude	342	1245	502	555	398	670	796	912	238	466

Calculate the arithmetic mean and median for both distance and altitude.

OR

12. (a) Explain the steps in conducting a hypothesis test. **(8)**

(b) A total of 150 customers of a petrol station are asked about their satisfaction with their car and motorbike insurance. The results are summarized below: Determine and interpret Pearson's $\chi 2$ statistic and Cramer's V. **(6)**

	Satisfied	Unsatisfied	Total
Car	33	25	58
Car (Diesel engine)	29	31	60
Motor bike	12	20	32
Total	74	76	150

13. (a) Explain the data analytical process model. **(8)**

(b) Discuss the methods for handling noisy data. Consider the following sorted data for price (in dollars) 4, 8, 15, 21, 21, 24, 25, 28, 34. **(6)**
Illustrate smoothing by bin means and bin boundaries

OR

14. (a) a) What is the need for sampling in data analytics? Discuss the different sampling techniques. **(8)**

(b) Use these methods to *normalize* the following group of data: **(6)**

200, 300, 400, 600, 1000

(i) min-max normalization by setting *min* = 0 and *max* = 1
(ii) z-score normalization
(iii) normalization by decimal scaling .

15. (a) A database has five transactions. Let min_sup be 60% and min_conf be 80%.
.

TID	*items_bought*
T100	{M, O, N, K, E, Y}
T200	{D, O, N, K, E, Y}
T300	{M, A, K, E}
T400	{M, U, C, K, Y}
T500	{C, O, O, K, I, E}

(a) Find all frequent itemsets using Apriori algorithm **(10)**

(b) Generate strong association rules from any one 3 itemset. **(4)**

OR

16. (a) Explain agglomerative hierarchical clustering with an example. **(8)**

(b) Suppose that the data mining task is to cluster points (with (x, y) representing location) into three clusters, where the points areA1(2,10), A2 (2,5), A3 (8,4), B1 (5,8), B2 (7,5), B3 (6,4), C1(1,2), C2 (4,9). The distance function is Euclidean distance. Suppose initially we assign A1, B1, and C1as the center of each cluster, respectively. Use the k-means algorithm to show only **(6)**
(a) The three cluster centers after the first round of execution.
(b) The final three clusters.

17. (a) Illustrate the working of a Map Reduce program with example. **(8)**

(b) Explain the data analytic architecture with a diagram. **(6)**

OR

18. (a) Discuss the architecture of HDFS and its features. **(8)**

(b) Illustrate the use of big data analytics in credit risk modeling. **(6)**

19. (a) List and explain the R functions used in descriptive statistics. **(8)**

(b) Explain hypothesis testing using ANOVA. **(6)**

OR

20. (a) Discuss the data visualization for multiple variables in R **(8)**

(b) Describe the R functions used for cleaning dirty data. **(6)**

(5 x 14 = 70)

Teaching Plan

No	Contents	No of Lecture Hrs
Module – 1(Mathematics for Data Analytics) (7 hrs)		
1.1	Descriptive statistics - Measures of central tendency	1
1.2	Measures of dispersion	1
1.3	Association of two variables - Discrete Variables	1
1.4	Association of two variables - Ordinal and Continuous variable	1
1.5	Probability calculus - Probability distributions	1
1.6	Inductive statistics - Point estimation, Interval estimation	1
1.7	Hypothesis Testing - Basic definitions, t-test	1
Module – 2 (Introduction to Data Analytics) (6 hrs)		
2.1	Introduction to Data Analysis –Analytics, Analytics process model	1
2.3	Analytical model requirements	1
2.4	Data Analytics Life Cycle overview	1
2.5	Basics of data collection	1
2.6	Basics of sampling and preprocessing	1
2.7	Dimensionality reduction	1
Module - 3 (Predictive and Descriptive Analytics) (8 hrs)		
3.1	Supervised Learning, Naive Bayes classification	1
3.2	KNN algorithm	1

3.3	Linear Regression	1
3.4	Unsupervised Learning- Clustering	1
3.5	Hierarchical algorithms Agglomerative algorithm	1
3.6	Partitional algorithms -K- Means	1
3.7	Association Rule Mining	1
3.8	Apriori algorithm	1
Module - 4 (Big Data Analytics) (7 hrs)		
4.1	Big Data Overview – State of the practice in analytics.	1
4.2	Example Applications - Credit Risk Modeling	1
4.3	Business Process Analytics.	1
4.4	Big Data Analytics using Map Reduce and Apache Hadoop	1
4.5	Big Data Analytics using Map Reduce and Apache Hadoop	1
4.6	Developing and Executing a Hadoop MapReduce Program	1
4.7	Developing and Executing a Hadoop MapReduce Program	1
Module - 5 (R programming for Data Analysis) (8 hrs)		
5.1	Overview of modern data analytic tools, Introduction to R, R Graphical User Interfaces	1
5.2	Data Import and Export, Attribute and Data Types	1

5.3	Descriptive Statistics	1
5.4	Exploratory Data Analysis, Visualization Before Analysis	1
5.5	Dirty Data, Visualizing a Single Variable	1
5.6	Examining Multiple Variable	1
5.7	Data Exploration Versus Presentation	1
5.8	Statistical Methods for Evaluation	1

CST 332	FOUNDATIONS OF SECURITY IN COMPUTING	Category	L	T	P	Credit	Year Of Introduction
		PEC	2	1	0	3	2019

Preamble: The purpose of this course is to create awareness among learners about the fundamentals of security and number theory. This course covers Integer & Modular Arithmetic, Primes & Congruences, Discrete Logarithms & Elliptic Curve Arithmetic and an overview of computer security. The concepts covered in this course enable the learners in effective use of cryptographic algorithms and to identify the security threats in computing.

Prerequisite: A sound knowledge in Mathematics, Discrete Computational Structures, Operating Systems and Database Systems.

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

CO1	Illustrate the operations and properties of algebraic structures, integer arithmetic and modular arithmetic. **(Cognitive Knowledge Level: Understand)**
CO2	Use the concepts of prime numbers and factorization for ensuring security in computing systems **(Cognitive Knowledge Level: Apply)**
CO3	Illustrate the concepts of Linear Congruence, Primitive Roots, Discrete Logarithms and Elliptic Curve Arithmetic **(Cognitive Knowledge Level: Apply)**
CO4	Summarize the threats and attacks related to computer and program security **(Cognitive Knowledge Level: Understand)**
CO5	Outline the key aspects of operating system and database 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	✔	✔	✔			✔		✔				✔

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	**30**	**30**	**30**
Apply	**40**	**40**	**40**
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 (Modular Arithmetic)

Integer arithmetic - Integer division, Divisibility, Greatest Common Divisor (GCD), Euclid's algorithm for GCD, Extended Euclid’s algorithm, Linear Diophantine Equations. Modular

arithmetic - Operations, Properties. Algebraic structures - Groups, Rings, Fields, Finite fields, GF(p), GF (2^n).

Module-2 (Prime Numbers and Factorization)

Prime numbers - Prime numbers and prime-power factorization, Fermat and Mersenne primes, Fermat's theorem, Applications, Euler's theorem, Euler's totient function, Applications. Primality testing – Deterministic algorithms and Probabilistic algorithms. Factorization - Fermat's factorization, Pollard p-1 method.

Module-3 (Linear Congruence, Primitive Roots and Elliptic Curve Arithmetic)

Linear congruence - Simultaneous linear congruence, Chinese Remainder Theorem (CRT). Congruence with a prime - Power modulus, Arithmetic modulo p, Pseudoprimes and Carmichael numbers, Solving congruence modulo prime powers. Primitive roots - Existence of primitive roots for primes, Discrete logarithms. Elliptic curve arithmetic – Prime curves, Binary curves, Addition of two points, Multiplication of a point by a constant.

Module-4 (Computer and Program Security)

Introduction to computer security – Threats, Vulnerabilities, Controls. Browser attack types, Web attacks targeting users, Email attack types. Introduction to program security - Non-malicious programming oversights, Malware.

Module-5 (Operating System and Database Security)

Operating system security – Security in operating system, Security in design of operating system. Database security – Security requirements of databases, Reliability and integrity, Database disclosure.

Text Books

1. Behrouz A Forouzan, Cryptography and Network Security, 3/e, Tata McGraw-Hill.
2. Charles P Pfleeger, Shari Lawrence Pfleeger, Jonathan Margulies, Security in Computing, 5/e, Prentice Hall.
3. G.A. Jones & J.M. Jones, Elementary Number Theory, Springer UTM, 2007

References

1. William Stallings, Cryptography and Network Security Principles and Practices, 4/e, Pearson Ed.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Find the n- bit word that is represented by the polynomial $x^2 + 1$ in GF(2^5).
2. Solve the linear Diophantine equation 21x + 14y=35.

Course Outcome 2 (CO2):

1. Prove that a Carmichael number cannot be the product of two distinct primes.
2. Use the Pollard p-1 method to find a factor of 57247159 with the bound B=8.

Course Outcome 3 (CO3):

1. Find an integer that has a remainder of 3 when divided by 7 and 13, but is divisible by 12.
2. In the elliptic curve E(1,2) over the field GF(11), find the equation of the curve and all the points on the curve.

Course Outcome 4 (CO4):

1. List three controls that could be applied to detect or prevent off-by-one errors.
2. How does fake email messages act as spam?

Course Outcome 5 (CO5):

1. Discuss the importance of auditability and access control in database security.
2. Explain the various factors which can make data sensitive.

Model Question Paper

QP CODE: **PAGES: ___**

Reg No:_____
Name:_____

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST 332
Course Name : FOUNDATIONS OF SECURITY IN COMPUTING

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

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

1. List the four properties of divisibility with examples.
2. Find gcd (401,700) using Euclid's algorithm.
3. Use Fermat's Little theorem to show that 91 is not a prime.
4. If m is relatively prime to n , show that $\Phi(mn) = \Phi(m)\,\Phi(n)$.
5. Solve the congruence relation $103x \equiv 57 \pmod{211}$.
6. Find a solution for the congruence $3x \equiv 5 \bmod 7^3$
7. What are the problems created by an off-by-one error?
8. How does a clickjacking attack succeed?
9. Explain the significance of correctness and completeness in the design of operating systems.
10. How does the two-phase update technique help the database manager in handling failures?

(10x3=30)

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

11. (a) For the group G = < Z_6*, x>, prove that it is an Abelian group. Also show the result of 5 x 1 and 1 ÷ 5. **(6)**

(b) Find a particular and the general solution to the following linear Diophantine equations. **(8)**
i) 19 x+13y = 20 ii) 40 x + 16 y =88

OR

12. (a) Describe the properties of modular arithmetic and modulo operator. **(6)**

(b) Using Extended Euclidean algorithm, find the multiplicative inverse of (i) 131 in Z_{180} and (ii) 23 in Z_{100}. **(8)**

13. (a) State and prove Fermat's theorem. **(6)**

(b) Explain Fermat's factorization method and use it to factor 809009. **(8)**

OR

14. (a) Define Euler's totient function. Prove that, Ø(pq)=(p-1)(q-1) where p and q are prime numbers. **(7)**

(b) Define Fermat primes. Show that any two distinct Fermat numbers are relatively prime. **(7)**

15. (a) Using Chinese Remainder Theorem, solve the system of congruence, x ≡2(mod 3), x ≡3(mod 5), x ≡2(mod 7). **(7)**

(b) Define Carmichael number and show that a Carmichael number must be the product of at least three distinct primes. **(7)**

OR

16. (a) For the group G = <Z_{19*}, x>, find the primitive roots in the group. **(6)**

(b) Consider the elliptic curve $y^2 = x^3 + x + 1$ defined over Z_{23}. If P = (3, 10) and Q = (9,7) are two points on the elliptic curve, find 2P and P + Q. **(8)**

17. (a) Distinguish the terms vulnerability, threat and control. **(4)**

(b) With the help of suitable examples, explain the security problems created by incomplete mediation and time-of-check to time-of use. **(10)**

OR

18. (a) Differentiate between man-in-the-browser attack and page-in-the-middle attack. **(4)**

(b) Explain the four aspects of malicious code infection. **(10)**

19. (a) List any six computer security related functions addressed by operating systems. **(6)**

(b) How does a kernelized design support in enforcing security mechanisms? **(8)**

OR

20. (a) Explain any four security requirements of databases. **(4)**

(b) How can database disclosure be prevented? With the help of suitable examples, explain any six types of disclosure. **(10)**

Teaching Plan

No	Contents	No.of Lecture Hrs
	Module-1 (Modular Arithmetic) (6 hrs)	
1.1	Integer arithmetic, Integer division, Divisibility, Greatest Common Divisor (GCD)	1
1.2	Euclid's algorithm for GCD, Extended Euclid's algorithm	1
1.3	Linear Diophantine Equations	1
1.4	Modular arithmetic operations, Properties of modular arithmetic	1
1.5	Groups, Rings and Fields	1
1.6	Finite fields – GF(p), GF(2^n)	1
	Module-2 (Prime Numbers and Factorization) (7 hrs)	
2.1	Prime numbers and prime-power factorization	1
2.2	Fermat and Mersenne primes	1
2.3	Fermat's theorem, Applications – Exponentiation, Multiplicative inverse	1
2.4	Euler's theorem, Euler's totient function, Applications	1
2.5	Primality testing – Deterministic algorithms – Divisibility algorithm	1

2.6	Primality testing – Probabilistic algorithms-Fermat test, Square root test, Miller - Rabin test	1
2.7	Factorization - Fermat's factorization, Pollard p-1 method	1
Module-3 (Linear Congruence, Primitive Roots and Elliptic Curve Arithmetic) (7 hrs)		
3.1	Linear congruence, Simultaneous linear congruence	1
3.2	Chinese Remainder Theorem (CRT)	1
3.3	Congruence with a Prime-Power Modulus, Arithmetic modulo p	1
3.4	Pseudo-primes and Carmichael numbers	1
3.5	Solving congruence modulo prime powers	1
3.6	Primitive roots, Existence of primitive roots for primes, Discrete logarithms	1
3.7	Elliptic curve arithmetic – Prime curves, Binary curves, Addition of two points, Multiplication of a point by a constant	1
Module-4 (Computer and Program Security) (7 hrs) (Text book2: Chapters 1, 3, 4)		
4.1	Threats, Vulnerabilities, Controls	1
4.2	Browser attack types	1
4.3	Web attacks targeting users	1
4.4	Email attack types	1
4.5	Non-malicious programming oversights (Lecture 1)	1
4.6	Non-malicious programming oversights (Lecture 2)	1
4.7	Malware – Four aspects of infection	1
Module-5 (Operating System and Database Security) (8 hrs)(Text book2: Chapters 5, 7)		
5.1	Security in operating system (Lecture 1)	1
5.2	Security in operating system (Lecture 2)	1
5.3	Security in design of operating system (Lecture 1)	1

5.4	Security in design of operating system (Lecture 2)	1
5.5	Security requirements of databases	1
5.6	Reliability & integrity	1
5.7	Database disclosure (Lecture 1)	1
5.8	Database disclosure (Lecture 2)	1

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
KTU
Estd. 2014

CST 342	AUTOMATED VERIFICATION	Category	L	T	P	CREDIT	YEAR OF INTRODUCTION
		PEC	2	1	0	3	2019

Preamble: This course is intended to impart the basic theory and algorithm for an automatic verification process namely model checking. This course covers finite-state modelling of hardware/software, linear-time properties, classification of linear-time properties, Linear Temporal Logic (LTL) - a formal language for property specification, LTL model checking algorithm and model checking case studies. This course enables the learners to prove correctness of a hardware/software used in safety critical systems in domains such as avionics, health care and automotive.

Prerequisite: NIL

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

CO1	Illustrate an application for model checking. **(Cognitive Knowledge Level: Understand)**
CO2	Describe finite-state modelling for hardware and software. **(Cognitive Knowledge Level: Understand)**
CO3	Identify linear-time properties required to represent the requirements of a system. **(Cognitive Knowledge Level: Apply)**
CO4	Specify a given linear-time property in Linear Temporal Logic (LTL). **(Cognitive Knowledge Level: Apply)**
CO5	Perform LTL model checking using the tool Symbolic Analysis Laboratory (SAL). **(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 (Marks)	Test 2 (Marks)	
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 Test : 25 marks

Continuous Assessment Assignment : 15 marks (Out 15, 10 marks shall be given for a model checking project to be implemented in SAL.)

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 (Introduction to Model Checking)

System Verification – Hardware and Software Verification, Model Checking, Characteristics of Model Checking.

Transition Systems – Transition System, Direct Predecessors and Successors, Terminal State, Deterministic Transition System.

Executions - Execution Fragment, Maximal and Initial Execution Fragment, Execution, Reachable States.

Module - 2 (Linear Time Properties)

Linear-Time (LT) Properties - Deadlock. Linear-Time Behavior - Paths and State Graph, Path Fragment, Maximal and Initial Path Fragment, Path. Traces - Trace and Trace Fragment, LT Properties - LT Property, Satisfaction Relation for LT Properties, Trace Equivalence and LT Properties. Safety Properties and Invariants - Invariants, Safety Properties, Trace Equivalence and Safety properties. Liveness Properties - Liveness Property, Safety vs. Liveness Properties. Fairness - Fairness, Unconditional, Weak and Strong Fairness, Fairness Strategies, Fairness and Safety. (Definition and examples only for all topics - no proof required).

Module - 3 (Regular Properties)

Regular Properties - Model Checking Regular Safety properties - Regular Safety property, Verifying Regular Safety Properties. Automata on Infinite Words - ω-Regular Languages and Properties, Nondeterministic Buchi Automata (NBA), Deterministic Buchi Automata (DBA),

Generalised Buchi Automata (Definitions only). Model Checking ω-Regular Properties - Persistence Properties and Product, Nested Depth-First Search (Only algorithms required).

Module - 4 (Linear Time Logic)

Linear Temporal Logic (LTL) - Syntax, Semantics, Equivalence of LTL Formulae, Weak Until, Release and Positive Normal Form, Fairness, Safety and Liveness in LTL (Definitions only). Automata Based LTL Model Checking (Algorithms and examples only).

Module - 5 (Model Checking in SAL)

Introduction - Introduction to the tool Symbolic Analysis Laboratory (SAL).

The Language of SAL - The expression language, The transition Language, The module language, SAL Contexts.

SAL Examples - Mutual Exclusion, Peterson's Protocol, Synchronous Bus Arbiter, Bounded Bakery protocol, Bakery Protocol, Traffic Signalling System.

Text Books

1. Christel Baier and Joost-Pieter Katoen, Principles of Model Checking, The MIT Press. (Modules 1 - 4)
2. Leonardo de Moura, Sam Owre and N. Shankar, The SAL Language Manual, SRI International (http://sal.csl.sri.com/doc/language-report.pdf, Chapters 1, 3, 4, 5, 6, 7) (Module 5)

Reference Materials

1. SAL Examples (http://sal.csl.sri.com/examples.shtml) (Module 5)

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Illustrate how model checking can make a system design reliable, based on a required set of properties/constraints.

Course Outcome 2 (CO2):

1. Consider a message delivery system. The sender *s* is trying to send a series of messages to the receiver *r* in such a way that the $(i+1)^{st}$ message is sent only after the i^{th} message is delivered. There is a possibility of error in sending a message and in that case, *s* keeps on

trying until it is able to send the message. Show a finite state transition system modeling this system.

Course Outcome 3 (CO3):

1. Consider a shared memory segment *s* protected using a mutex lock variable *m*. Two processes p_1 and p_2 are trying to access *s*. List the Linear Time properties of the system which will ensure safety, liveness and fairness.

Course Outcome 4 (CO4):

1. Show the LTL specifications of the safety, liveness and fairness properties listed for the assessment question given in CO3.

Course Outcome 5 (CO5):

1. Model the system mentioned in the question given in CO3 in SAL and verify that the system is correct with respect to the LTL properties shown as the answer for CO4.

Model Question paper

QP CODE: **PAGES: 3**

Reg No:_____ **Name :_____**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST342

Course Name: Automated Verification

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

PART A

Answer all questions. Each question carries 3 marks.

1. Define model checking. Show the schematic diagram of the model checking approach.

2. Show a transition system modeling a coffee/Tea vending machine.

3. Define invariant as a Linear Time (LT) property. Give an example

4. List any three Linear Time properties in the Mutual Exclusion problem of processes.

5. Illustrate the construction of a product automaton from two automata.

6. Differentiate between Deterministic Buchi Automaton and Non-deterministic Buchi Automaton.

7. Specify the following statements about traffic lights in Linear Temporal Logic (LTL).
   a. Once red, the light can not become green immediately.
   b. Once red, the light always becomes green eventually after being yellow for some time.

8. What is Positive Normal Form (PNF) in LTL? Give an example.

9. List any three applications of the tool Symbolic Analysis Laboratory (SAL).

10. What is a SAL context? Give an example.

(10x3=30)

Part B

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

11. (a) Explain in detail the various phases of the model checking process. **(8)**

    (b) Explain the strengths and weaknesses of model checking. **(6)**

OR

12. (a) Define and illustrate the following terms of a transition system. **(14)**
    a. Execution Fragment
    b. Maximal and Initial Execution Fragment
    c. Execution
    d. Reachable States

13. (a) With an example, explain the satisfaction relation for LT properties. **(7)**

(b) What is trace equivalence in Transition Systems? Give an example to show that if two transition systems satisfy the trace equivalence property, then they satisfy the same set of LT properties. **(7)**

OR

14. (a) Give the transition system for the fault tolerant variant of the dining philosophers problem. **(4)**

(b) With a suitable example, explain the algorithms to check whether a Transition System satisfies an invariant or not. **(10)**

15. (a) Explain Regular Safety Properties with a suitable example. **(7)**

(b) Illustrate an algorithm for verifying Regular Safety Properties. **(7)**

OR

16. (a) Explain ω-Regular Properties. **(4)**

(b) Illustrate how ω-Regular Properties are verified. **(10)**

17. (a) Explain the syntax of Linear Temporal Logic (LTL). **(7)**

(b) Explain the semantics of LTL. **(7)**

OR

18. (a) With an example, give the difference between until and weak until in LTL. **(4)**

(b) With a suitable example, explain automata based LTL model checking. **(10)**

19. (a) Explain Peterson's protocol. What are the LTL properties to be verified to ensure its correctness? **(8)**

(b) Write a SAL script for the verification of Peterson's protocol. **(6)**

OR

20. (a) Show the SAL model corresponding to Bakery protocol. **(8)**

(b) List any three Linear Time properties of this model and show their LTL **(6)**

Teaching Plan

Module 1 (Introduction to Model Checking)		**4 Hours**
1.1	**System Verification** – Hardware and Software Verification, Model Checking, Model Checking	1 Hour
1.2	**Transition Systems** – Transition System, Direct Predecessors and Successors, Terminal State, Deterministic Transition System	1 Hour
1.3	**Executions** - Execution Fragment, Maximal and Initial Execution Fragment	1 Hour
1.4	Execution, Reachable States	1 Hour
Module 2 (Linear Time Properties)		**8 Hours**
2.1	Linear-Time (LT) Properties - Deadlock	1 Hour
2.2	Linear-Time Behavior - Paths and State Graph, Path Fragment, Maximal and Initial Path Fragment, Path	1 Hour
2.3	Traces - Trace and Trace Fragment	1 Hour
2.4	LT Property, Satisfaction Relation for LT Properties, Trace Equivalence and LT Properties	1 Hour
2.5	Invariants	1 Hour
2.6	Safety Properties, Trace Equivalence and Safety properties	1 Hour
2.7	Liveness Property, Safety vs. Liveness Properties	1 Hour
2.8	Fairness, Unconditional, Weak and Strong Fairness, Fairness Strategies, Fairness and Safety	1 Hour
Module 3 (Regular Properties)		**9 Hours**
3.1	**Regular Properties** - Model Checking Regular Safety properties - Regular Safety property	1 Hour
3.2	Verifying Regular Safety Properties	1 Hour
3.3	Automata on Infinite Words - ω -Regular Languages and Properties	2 Hour

3.4	Nondeterministic Buchi Automata (NBA), Deterministic Buchi Automata (DBA), Generalised Buchi Automata	1 Hour
3.5	Model Checking ω-Regular Properties - Persistence Properties and Product - Lecture 1	1 Hour
3.6	Persistence Properties and Product - Lecture 2	1 Hour
3.7	Nested Depth-First Search (Lecture 1)	1 Hour
3.8	Nested Depth-First Search (Lecture 2)	1 Hour
Module 4 (Linear Time Logic)		**7 Hours**
4.1	**Linear Temporal Logic –** Linear Temporal Logic (LTL) - Syntax	1 Hour
4.2	Semantics - Lecture 1	1 Hour
4.3	Equivalence of LTL Formulae, Weak Until	1 Hour
4.4	Release and Positive Normal Form	1 Hour
4.5	Fairness, Safety and Liveness in LTL	1 Hour
4.6	Automata Based LTL Model Checking (Lecture 1)	1 Hour
4.7	Automata Based LTL Model Checking (Lecture 2)	1 Hour
Module 5 (Model Checking in SAL)		**7 Hours**
5.1	Introduction - Introduction to the tool Symbolic Analysis Laboratory (SAL).	1 Hour
5.2	The Language of SAL - The expression language, The transition Language	1 Hour
5.3	The module language, SAL Contexts.	1 Hour
5.4	SAL Examples - Mutual Exclusion	1 Hour
5.5	Peterson's Protocol, Synchronous Bus Arbiter	1 Hour
5.6	Bounded Bakery protocol, Bakery Protocol	1 Hour
5.7	Traffic Signalling System	1 Hour

CST 362	PROGRAMMING IN PYTHON	Category	L	T	P	Credit	Year of Introduction
		PEC	2	1	0	3	2019

Preamble: The objective of the course is to equip the learners to develop multi-module software solutions for real world computational problems using Python. It encompasses the Python programming environment, syntax, data representations, intermediate level features, GUI programming, Object Oriented Programming and data processing. This course lays the foundation to develop modular software solutions including complex interactive applications, network applications, and data-driven intelligent applications.

Prerequisite: Basic knowledge in Computational Problem Solving, A course in any programming language.

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 and if-elif-else) and iterative (while and for) statements in Python programs. **(Cognitive Knowledge level: Apply)**
CO3	Develop programs by utilizing the Python programming constructs such as Lists, Tuples, Sets and Dictionaries. **(Cognitive Knowledge level: Apply)**
CO4	Develop graphical user interface for solutions using Python libraries. (**Cognitive Knowledge level: Apply)**
CO5	Implement Object Oriented programs with exception handling. **(Cognitive Knowledge level: Apply)**
CO6	Write programs in Python to process data stored in files by utilizing 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	35	35	35
Apply	45	45	45
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 (Programming Environment and Python Basics) (6 hours)

Getting started with Python programming – Interactive shell, IDLE, iPython Notebooks, Detecting and correcting syntax errors, How Python works. The software development process – A case study. Basic coding skills – strings, assignment, and comments, Numeric data types and character sets, Expressions, Using inbuilt functions and modules. Control statements – Iteration with for/while loop, Formatting text for output, A case study, Selection structure (if-else, switch-case), Conditional iteration with while, A case study, Testing control statements, Lazy evaluation.

Module -2 (Building Python Programs) (8 hours)

Strings and text files – Accessing characters, substrings, Data encryption, Strings and number system, String methods, Text files, A case study on text analysis. Design with Functions – Functions as Abstraction Mechanisms, Problem solving with top-down design, Design with recursive functions, Managing a program's namespace, Higher-Order Functions. Lists - Basic list Operations and functions, List of lists, Slicing, Searching and sorting list, List comprehension. Work with tuples. Sets. Work with dates and times, A case study with lists. Dictionaries - Dictionary functions, dictionary literals, adding and removing keys, accessing and replacing values, traversing dictionaries, reverse lookup. Case Study – Data Structure Selection.

Module -3 (Graphics) (7 hours)

Graphics – Terminal-based programs, Simple Graphics using Turtle, Operations, 2D Shapes, Colors and RGB Systems, A case study. Image Processing – Basic image processing with inbuilt functions. Graphical User Interfaces – Event-driven programming, Coding simple GUI-based programs : Windows, Labels, Displaying images, Input text entry, Popup dialog boxes, Command buttons, A case study.

Module -4 (Object Oriented Programming) (7 hours)

Design with classes - Objects and Classes, Methods, Instance variables, Constructor, Accessor and Mutator, Data-Modeling Examples, Structuring classes with inheritance and polymorphism. Abstract classes, Interfaces, Exceptions - Handle a single exception, handle multiple exceptions.

Module -5 (Data Processing) (9 hours)

The os and sys modules, NumPy - Basics, Creating arrays, Arithmetic, Slicing, Matrix Operations, Random numbers. Plotting and visualization. Matplotlib - Basic plot, Ticks, Labels, and Legends. Working with CSV files. – Pandas - Reading, Manipulating, and Processing Data. Introduction to Micro services using Flask.

Text Books:

1. Kenneth A Lambert., Fundamentals of Python : First Programs, 2/e, Cengage Publishing, 2016
2. Wes McKinney, Python for Data Analysis, 2/e, Shroff / O'Reilly Publishers, 2017
3. Flask: Building Python web services, Jack Stouffer, Shalabh Aggarwal, Gareth Dwyer, PACKT Publishing Limited, 2018

Reference Books:

1. Zed A Shaw, Learn Python 3 The Hard Way, Addison-Wesley, 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. Charles Severance. Python for Informatics: Exploring Information,

Sample Course Level Assessment Questions

Course Outcome1(CO1):

1. What is type conversion? How is it done in Python?
2. Write a note on the Python editors.

Course Outcome 2(CO2):

1. Write a Python program which takes a positive integer **n** as input and finds the sum of cubes all positive even numbers less than or equal to the number.
2. What is printed when the below code is executed?

```
mysum = 0
   for i in range(5, 11, 2):
               mysum += i
               if mysum == 5:
                         break
                    mysum += 1
   print(mysum)
```

What would be the output if 'break' is replaced with 'continue' in the above code fragment?

Course Outcome 3(CO3):

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.

2. What is the value of L after you run the code below?

```
L = ["life", "answer", 42, 0]
for thing in L:
      if thing == 0:
        L[thing] = "universe"
      elif thing == 42:
          L[1] = "everything"
```

Course Outcome 4(CO4):

1. A bouncy program is defined as follows – The program computes and displays the total distance traveled by a ball, given three inputs—the initial height from which it is dropped, its bounciness index, and the number of bounces. Given the inputs write a GUI-based program to compute the total distance traveled.
2. Write a Python program to find the quadrant of a point, say (x,y).

Course Outcome 5(CO5):

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.
2. Explain inheritance in Python using suitable examples.

Course Outcome 6(CO6):

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
    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.
2. Given two matrices A and B, write a program to find the product of A and B^T.

Model Question Paper

QP CODE: PAGES:

Reg No:_____

Name:_____

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

Course Code: CST 362

Course name : PROGRAMMING IN PYTHON

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

PART-A

(Answer All Questions. Each question carries 3 marks)

1. Write a Python program to reverse a number and also find the sum of digits of the number. Prompt the user for input.
2. Explain the concept of scope and lifetime of variables in Python programming language, with a suitable example.
3. Illustrate format specifiers and escape sequences with examples.
4. Compare tuples, lists, and dictionaries with examples.
5. Describe the following dictionary methods with an example.
   i. *get( )* ii. *Keys( )* iii. *pop( )* iv. *update( )* v. *values( )* vi. *items( )*
6. Differentiate the terminal-based and GUI-based programming in Python.
7. What is polymorphism? Give an example in the context of OOP in Python.
8. How is exception handling accomplished in Python programs?
9. Explain the **os** and **os.path** modules in Python with examples. Also, discuss the *walk( )* and *getcwd( )* methods of the **os** module.
10. What are the important characteristics of CSV file format.

PART-B

(Answer any one full question from each module)

11. (a) Write a Python code to check whether a given year is a leap year or not [An year is a leap year if it's divisible by 4 but not divisible by 100 except for those divisible by 400]**.** (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) Write a Python program to find the value for *sin(x)* up to *n* terms using the series (6)

$$\sin(x) = \frac{x}{1!} - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots$$ where x is in degrees

(b) Write a Python code to determine whether the given string is a Palindrome or not using slicing. Do not use any string function. (8)

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. (6)

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. (8)

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**

15. (a) Write a program to draw a hexagon using turtle. (5)

(b) Write a note on the image processing function in Python. (9)

OR

16. (a) Describe the features of event driven programming. (4)

(b) Write a GUI-based program that allows the user to convert temperature values between degrees Fahrenheit and degrees Celsius. The interface should have labeled entry fields for these two values. These components should be arranged in a grid where the labels occupy the first row and the corresponding fields occupy the second row. At start-up, the Fahrenheit field should contain 32.0, and the Celsius field should contain 0.0. The third row in the window contains two command buttons, labeled >>>> and <<<<. When the user presses the first button, the program should use the data in the Fahrenheit field to compute the Celsius value, which should then be output to the Celsius field. The second button should perform the inverse function. (10)

17. (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. (10)

(b) Explain inheritance in Python. Give examples for each type of inheritance. (4)

OR

18. (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) Write Python program to create a class called as **Complex** and implement *__add__( )* method to add two complex numbers. Display the result by overloading the + Operator. (8)

19. (a) Write a Python program to add two matrices and also find the transpose of the resultant matrix. (8)

(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 (6)

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.

OR

20. (a) Write Python program to write the data given below to a CSV file. (5)

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) 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 (9)

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.

(14X5=70)

Teaching Plan

Module 1: Programming Environment and Python Basics		**(6 hours)**
1.1	Getting started with Python programming – Interactive shell, IDLE, iPython Notebooks, Detecting and correcting syntax errors, How Python works.	**1 hour**
1.2	The software development process – A case study.	**1 hour**
1.3	Basic coding skills – strings, assignment, and comments, Numeric data types and character sets	**1 hour**
1.4	Expressions, Using inbuilt functions and modules.	**1 hour**
1.5	Control statements – Definite Iteration with for loop, Formatting text for output, Selection structure (if-else, switch-case), Conditional iteration with	**1 hour**

	while loop, A case study	
1.6	Testing the control statements, Lazy evaluation.	**1 hour**
Module 2: Building Python Programs		**(8 hours)**
2.1	Strings – Accessing characters, substrings, Data encryption, Strings and number system, String methods,	**1 hour**
2.2	Text files, A case study on text analysis.	**1 hour**
2.3	Design with Functions – Functions as Abstraction Mechanisms, Problem solving with top-down design,	**1 hour**
2.4	Design with recursive functions, Managing a program's namespace, Higher-Order Functions.	**1 hour**
2.5	Lists - Basic list Operations and functions, List of lists, Slicing, Searching and sorting list, List comprehension.	**1 hour**
2.6	Work with tuples. Sets. Work with dates and times, A case study with lists.	**1 hour**
2.7	Dictionaries - Dictionary functions, dictionary literals, adding and removing keys, accessing and replacing values, traversing dictionaries, reverse lookup.	**1 hour**
2.8	Case Study - Data Structure Selection.	**1 hour**
Module 3: Graphics		**(7 hours)**
3.1	Graphics – Simple Graphics using Turtle, Operations, 2D Shapes,	**1 hour**
3.2	Colors and RGB Systems, A case study.	**1 hour**
3.3	Image Processing – Basic image processing with inbuilt functions.	**1 hour**
3.4	Graphical User Interfaces – Event-driven programming	**1 hour**
3.5	Coding simple GUI-based programs : Windows, Labels, Displaying images,	**1 hour**
3.6	Coding simple GUI-based programs : Input text entry, Popup dialog boxes, Command buttons	**1 hour**
3.7	A case study - GUI	**1 hour**

Module 4: Object Oriented Programming		**(7 hours)**
4.1	Design with classes : Objects and Classes, Methods, Instance Variables	**1 hour**
4.2	Constructor, Accessors, and Mutators	**1 hour**
4.3	Structuring classes with Inheritance	**1 hour**
4.4	Polymorphism	**1 hour**
4.5	Abstract Classes	**1 hour**
4.6	Interfaces	**1 hour**
4.7	Exceptions : Handle a single exception, handle multiple exceptions	**1 hour**
Module 5: Data Processing		**(9 hours)**
5.1	The *os* and *sys* modules, NumPy : Basics, Creating arrays, Arithmetic, Slicing	**1 hour**
5.2	Matrix Operations, Random numbers.	**1 hour**
5.3	Matplotlib : Basic plot, Ticks, Labels, and Legends	**1 hour**
5.4	Working with CSV files	**1 hour**
5.5	Pandas : Reading, Manipulating	**1 hour**
5.6	Pandas : Processing Data and Visualize.	**1 hour**
5.7	Introduction to Microservices using Flask	**1 hour**
5.8	Introduction to Microservices using Flask	**1 hour**
5.9	Introduction to Microservices using Flask	**1 hour**

CST 372	DATA AND COMPUTER COMMUNICATION	Category	L	T	P	Credits	Year of Introduction
		PEC	2	1	0	3	2019

Preamble:

The purpose of this course is to prepare learners to understand the communication entities and the associated issues in data transmission. This course covers fundamental concepts of data transmission in digital and analog form, transmission media, concepts of encoding, multiplexing, spread spectrum and switching methods. This course helps the learner to gain insight into the important aspects of data communication and computer networking systems and enables to apply in practical applications.

Prerequisite: NIL

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

CO#	Course Outcomes
CO1	Identify the characteristics of signals for analog and digital transmissions (**Cognitive knowledge: Apply**)
CO2	Identify the issues in data transmission (**Cognitive knowledge: Apply**)
CO3	Select transmission media based on characteristics and propagation modes (**Cognitive knowledge: Apply**)
CO4	Choose appropriate signal encoding techniques for a given scenario (**Cognitive knowledge: Apply**)
CO5	Illustrate multiplexing and spread spectrum technologies (**Cognitive knowledge: Apply**)
CO6	Use error detection, correction and switching techniques in data communication (**Cognitive knowledge: 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**	Lifelong learning

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination (%)
	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 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 maximum 2 sub-divisions and carries 14 marks.

Syllabus

Module-1 (Data Transmission Basics)

Communication model - Simplex, Half duplex, Full duplex transmission. Periodic analog signals - Sine wave, Amplitude, Phase, Wavelength, Time and frequency domain, Bandwidth. Analog & digital data and signals. Transmission impairments - Attenuation, Delay distortion, Noise. Data rate limits - Noiseless channel, Nyquist bandwidth, Noisy channel, Shannon's capacity formula.

Module-2 (Transmission Media)

Guided transmission media - Twisted pair, Coaxial cable, Optical fiber. Unguided media - Radio waves, Terrestrial microwave, Satellite microwave, Infrared. Wireless propagation - Ground wave propagation, Sky wave propagation, Line-of-Sight (LoS) propagation.

Module-3 (Digital Transmission and Analog Transmission)

Digital data to digital signal – Non-Return-to-Zero (NRZ), Return-to-Zero (RZ), Multilevel binary, Biphase. Analog data to digital signal - Sampling theorem, Pulse Code Modulation (PCM), Delta Modulation (DM). Digital data to analog signal - Amplitude Shift Keying (ASK), Frequency Shift Keying (FSK), Phase Shift Keying (PSK). Analog data to analog signal - Amplitude Modulation (AM), Frequency Modulation (FM), Phase Modulation (PM).

Module-4 (Multiplexing and Spread Spectrum)

Multiplexing - Frequency Division Multiplexing (FDM), Wavelength Division Multiplexing (WDM), Time Division Multiplexing (TDM), Characteristics, Synchronous TDM, Statistical TDM. Spread spectrum techniques - Direct Sequence Spread Spectrum (DSSS), Frequency Hopping Spread Spectrum (FHSS), Code Division Multiplexing, Code Division Multiple Access (CDMA).

Module-5 (Error Detection, Correction and Switching)

Digital data communication techniques - Asynchronous transmission, Synchronous transmission. Detecting and correcting errors - Types of errors, Parity check, Checksum, Cyclic Redundancy Check (CRC), Forward Error Correction (FEC), Hamming distance, Hamming code. Basic principles of switching - Circuit switching, Packet switching, Message switching.

Text Books

1. Forouzan B. A., Data Communications and Networking, 5/e, McGraw Hill, 2013.

2. William Stallings, Data and Computer Communication 9/e, Pearson Education, Inc.

References

1. Schiller J., Mobile Communications, 2/e, Pearson Education, 2009.
2. Curt M. White, Fundamentals of Networking and Communication 7/e, Cengage learning.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. If the spectrum of a channel is between 3 MHz and 4 MHz and SNR_{dB} = 24 dB, calculate the Shannon capacity.
2. Assume that a periodic signal is composed of five sine waves with frequencies 200, 400, 600, 800 and 1000 Hz. Determine the bandwidth. Draw the spectrum assuming all components have a maximum amplitude of 5 V.

Course Outcome 2 (CO2):

1. Given a receiver with an effective noise temperature of 294 K and a bandwidth of 10 MHz. Find the thermal noise level at the receiver side in dBW.
2. The loss in a cable is usually defined in decibels per kilometer (dB/km). If the signal at the beginning of a cable with -0.3 db/km has a power of 2 mW, determine the power of the signal at 5 km.

Course Outcome 3 (CO3):

1. Explain the reflective property of a parabolic antenna.
2. Two separate frequencies are used for uplink and downlink transmission in satellite communication. Give reason.

Course Outcome 4 (CO4):

1. Encode the data sequence 101011100 using Multilevel binary and Biphase schemes.
2. Encode the data bits 00101101110001 using 2B1Q encoding scheme. Assume negative original level.

Course Outcome 5 (CO5):

1. The frequency spectrum of input signals will move to high frequency bands by the FDM process. Justify.
2. Four channels are multiplexed using TDM. If each channel sends 100 bytes/sec and we multiplex one byte per channel, determine the frame size, duration of a frame, frame rate and bit rate of link.

Course Outcome 6 (CO6):

1. Using the divisor polynomial $x^4 + x + 1$, determine the Cyclic Redundancy Check (CRC) for the dataword 10110100. Also, perform the checking at the receiver side.
2. How many redundancy bits are required to generate the Hamming code for a 7-bit data? Assuming even parity, generate the Hamming code for the 7-bit dataword 1001101. If the fifth bit from the left of the received codeword is changed to 0, can

this be detected? Give reasons for your answer.

Model Question Paper

QP CODE: **PAGES: 3**

Reg No:_____
Name:_____

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SIXTH SEMESTER B.TECH DEGREE EXAMINATION, MONTH & YEAR

Course Code: CST 372

Course Name : Data and Computer Communication

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

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

1. Define bandwidth. Find the lowest frequency, if a periodic signal has a bandwidth of 20 Hz and the highest frequency is 60 Hz. Draw the spectrum if the signal contains all frequencies of the same amplitude.

2. Assume that a TV picture is to be transmitted over a channel with 4.5 MHz bandwidth and a 35 dB Signal-to-Noise-Ratio. Find the capacity of the channel.

3. How does twisting affect the performance in a twisted pair cable?

4. Which wireless propagation method is suitable for satellite communication? Justify your answer.

5. Explain the two main distortions that can occur in a delta modulated waveform. How can it be avoided?

6. Illustrate the equivalent square wave pattern of the bit string 01001101 using Non-Return-to-Zero (NRZ) - Level and NRZ-Invert encoding schemes.

7. Apply Direct Sequence Spread Spectrum to the data 101 using the Barker sequence 10110111000. Show the encoding and decoding steps.

8. Compare synchronous and statistical time division multiplexing.

9. Find the minimum hamming distance for the following cases:
    a) Detection of two errors

b) Correction of two errors
c) Detection of three errors

10. Find the parity bit for simple even parity check for the following. **(10x3=30)**
    a) 1001010
    b) 0001100
    c) 1000000

Part B

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

11. (a) With the help of suitable figures, distinguish between time and frequency domain representations. **(4)**

    (b) Describe the different types of transmission impairments. **(10)**

OR

12. (a) Calculate the bandwidth, if a periodic signal is decomposed into 4 sine waves with frequencies 50 Hz, 100 Hz, 150 Hz and 200 Hz. Draw the spectrum, assuming all components having an amplitude in the range 6-12 V and all are multiples of two in the increasing order. **(6)**

    (b) Distinguish between Nyquist bandwidth and Shannon capacity. Consider a noiseless channel with a bandwidth of 3000 Hz transmitting a signal with (i) Two signal levels (ii) Four signal levels. Determine the maximum bit rate in both cases. **(8)**

13. (a) For a parabolic reflective antenna operating at 12 GHz with a diameter of 2 m, calculate the effective area and the antenna gain. **(6)**

    (b) List any four advantages and disadvantages of twisted pair, coaxial cable and fiber optic cable. **(8)**

OR

14. (a) Compare the features of terrestrial microwave and satellite microwave. **(6)**

    (b) With the help of suitable diagrams, differentiate Multi-mode and Single-mode optical fibres. How are the rays propagated in Step-index and Graded-index Multi-mode fibres? **(8)**

15. (a) Distinguish between data rate and signal rate. **(4)**

(b) How is polar encoding done? Encode the pattern 010011001110 using the two Biphase schemes. **(10)**

OR

16. (a) Show the equivalent analog sine wave pattern of the bit string 010011010 using Amplitude Shift Keying, Frequency Shift Keying and Phase Shift Keying. **(4)**

(b) State Sampling theorem. Explain Pulse Code Modulation with suitable figures. **(10)**

17. (a) Four channels are multiplexed using Time Division Multiplexing. If each channel sends 100 bytes/sec and we multiplex one byte per channel, determine the frame size, duration of a frame, frame rate and bit rate of the link. **(6)**

(b) Explain the working of Frequency Hopping Spread Spectrum with an example. **(8)**

OR

18. (a) Explain any three techniques by which the disparity in input data rate is handled by Time Division Multiplexing. Give examples. **(4)**

(b) Suppose Alice and Bob are communicating using Code Division Multiple Access. Alice uses the code [+1 +1] and Bob uses the code [+1 -1]. Alice sends a data bit 0 and Bob sends a data bit 1. Show the data in the channel and how they can detect what the other person has sent. **(10)**

19. (a) Explain two-dimensional parity check with examples. **(4)**

(b) Describe the need for a switch in a communication system. What are the different phases in circuit switching? **(10)**

OR

20. (a) Explain the virtual circuit approach of packet switching with a suitable example. **(6)**

(b) Find the Hamming code for the data word 1011001. Assume odd parity. **(8)**

Teaching Plan

No	Contents	No.of Lecture Hrs (35 hrs)
Module-1 (Data Transmission Basics) (6 hrs)		
1.1	Introduction, Communication model, Simplex, Half duplex, Full duplex transmission, Periodic analog signals, Sine wave, Amplitude, Phase, Wavelength	1
1.2	Time and frequency domain, Bandwidth	1
1.3	Analog & digital data and signals	1
1.4	Transmission impairments, Attenuation, Delay distortion, Noise	1
1.5	Data rate limits, Noiseless channel, Nyquist bandwidth	1
1.6	Noisy channel, Shannon's capacity formula	1
Module-2 (Transmission Media) (6 hrs)		
2.1	Guided transmission media, Twisted pair, Coaxial cable	1
2.2	Optical fiber	1
2.3	Unguided media, Radio waves	1
2.4	Terrestrial microwave, Satellite microwave	1
2.5	Infrared	1
2.6	Wireless Propagation, Ground wave, Sky wave, Line-of-Sight (LoS) propagation	1
Module-3 (Digital Transmission and Analog Transmission) (8 hrs)		
3.1	Digital data to digital signal, Non-Return-to-Zero (NRZ), Return-to-Zero (RZ)	1
3.2	Multilevel binary and Biphase	1
3.3	Analog data to digital signal, Sampling theorem, Pulse Code Modulation (PCM)	1

3.4	Delta Modulation (DM)	1
3.5	Digital data to analog signal, Amplitude Shift Keying (ASK)	1
3.6	Frequency Shift Keying (FSK), Phase Shift Keying (PSK)	1
3.7	Analog data to analog signal, Amplitude Modulation (AM)	1
3.8	Frequency Modulation (FM), Phase Modulation (PM)	1
Module-4 (Multiplexing and Spread Spectrum) (7 hrs)		
4.1	Multiplexing, Frequency Division Multiplexing (FDM), Wavelength Division Multiplexing (WDM)	1
4.2	Time Division Multiplexing (TDM), Characteristics, Synchronous TDM, Statistical TDM	1
4.3	Spread spectrum techniques, Direct Sequence Spread Spectrum (DSSS)	1
4.4	Frequency Hopping Spread Spectrum (FHSS)	1
4.5	Code Division Multiplexing	1
4.6	Code Division Multiple Access (CDMA) (Lecture 1)	1
4.7	CDMA (Lecture 2)	1
Module-5 (Error Detection, Correction and Switching) (8 hrs)		
5.1	Digital data communication techniques, Asynchronous & Synchronous transmission	1
5.2	Detecting and correcting errors, Types of errors, Parity check, Checksum	1
5.3	Cyclic Redundancy Check (CRC)	1
5.4	Forward Error Correction (FEC), Hamming distance	1
5.5	Hamming code	1
5.6	Basic principles of switching, Circuit switching	1
5.7	Packet switching	1
5.8	Message switching	1

SEMESTER VI

MINOR

CST 382	INTRODUCTION TO SOFTWARE TESTING	Category	L	T	P	Credits	Year of Introduction
		VAC	3	1	0	4	2019

Preamble:

This is a course in theoretical computer science that includes test cases for white-box, black-box, and grey-box approaches. This course describes the various techniques for test case design used to test software artifacts, including requirements, design, and code. The course includes different techniques for test case design based on graphs, programming language syntaxes and inputs. The course also covers symbolic execution using PEX tool.

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

CO1	List a range of different software testing techniques and be able to apply specific unit testing method to the projects using Junit**. (Cognitive Knowledge Level: Understand)**
CO2	Explain mutation testing method for a given piece of code to identify hidden defects that can't be detected using other testing methods. **(Cognitive Knowledge Level: Understand)**
CO3	Explain graph coverage criteria in terms of control flow graph and data flow graph for a given program. **(Cognitive Knowledge Level: Understand**)
CO4	Demonstrate the importance of black-box approaches in terms of domain and functional testing. **(Cognitive Knowledge Level: Understand)**
CO5	Illustrate the use of PEX tool with symbolic execution. **(Cognitive Knowledge Level: Apply)**

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO 9	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 (Marks)	**Test 2 (Marks)**	
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 - 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 (Introduction to Software Testing)

Some Popular Errors – Ariane 5, Therac 25, Intel Pentium Bug. What is Software testing? Why should it be tested? Software Quality, Role of Testing. Testing Process - Level 0 thinking, Level 1 thinking, Level 2 thinking, Level 3 thinking, Level 4 thinking. Software Testing Terminologies - Verification, Validation and Testing, Faults, Error and Bug, Test cases, Coverage Criteria. Types of Testing- Unit testing, integration testing, System testing, Acceptance testing, Beta testing, Functional testing, Stress testing, Performance testing, Usability testing and Regression testing. Testing Methods - Black Box testing, White Box testing, Grey Box testing.

Module - 2 (Unit Testing)

Concept of Unit testing. Static Unit testing. Dynamic Unit testing - Control Flow testing, Data Flow testing, Domain testing, Functional Program testing. Mutation testing - Mutation and Mutants, Mutation operators, Mutation score. Junit - Framework for Unit testing. Case Study - Mutation testing using Junit and Muclipse.

Module - 3 (Unit Testing - White Box Approaches)

Overview of Graph Coverage Criteria. Structural Graph Coverage Criteria - Node/vertex coverage, Edge coverage, Edge pair coverage, Path coverage, Complete path coverage, Prime path coverage, Complete round trip coverage, Simple round trip coverage. Data Flow Criteria - du paths, du pairs. Subsumption Relationships among Graph Coverage Criteria. Graph Coverage for Source Code - Control flow graphs for code, CFG: If statement, CFG: If statement with return, CFG: Switch-case, CFG: Loops, CFG: Exceptions (try-catch). Example program – Statistics. Graph Coverage for Design Elements - Call graphs and classes, Class inheritance testing: Coverage criteria, Coverage criteria on inheritance graph, Data flow at the design level, Inter-procedural DU pairs, Coupling du-pairs example. Example - Quadratic Root. Case Study - Graph Based testing using JUnit Framework.

Module - 4 (Unit Testing - Black Box Approaches)

Domain Testing / Input Space Partitioning - Partitions of a set. Input domain modelling - Interface-based approach, Functionality-based approach. Identifying values. Multiple partitions of the inputdomain - All Combinations Coverage (ACoC), Each Choice Coverage (ECC), Pair-wise Coverage, T-wise Coverage, Base Choice Coverage, Multiple Base Choices Coverage. TriTyp example. Functional Testing - Functional Testing Concepts of Howden. Functional testing - Important Steps. Types of Functional testing - Equivalence Class Partitioning, Boundary Value Analysis, Decision Tables, Random Testing. Case Study - Black Box testing approaches using JUnit.

Module - 5 (Grey Box Testing Approaches)

Introduction to Grey Box testing - Why Grey Box testing, Gray Box Methodology, Advantages and Disadvantages. Techniques of Grey Box Testing - Matrix Testing, Regression Testing, Orthogonal Array Testing or OAT, Pattern Testing. An Introduction to PEX - Parameterized Unit Testing, The Testing Problem. Symbolic Execution – Example, Symbolic execution tree. PEX application Case Study – PEX.

Text Books

1. Paul Ammann and JeffOffutt ,Introduction to Software Testing.
2. KshirasagarNaik and PriyadarshiTripathy, Software Testing And Quality Assurance: Theory And Practice.

Reference Materials

1. https://www.csc.ncsu.edu/academics/undergrad/honors/thesis/muclipsebinder.pdf - Muclipse tutorial.
2. King, James C, "Symbolic Execution and Program Testing", Association for Computing Machinery, July 1976.
3.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1): Explain the following types of testing methods with examples.

(i) Balck-box testing.

(ii) White-box testing.

(iii) Grey-box testing.

Course Outcome 2 (CO2): Define 12 mutants for the following method *power*() using effective mutation operators. Try to use each mutation operator at least once. Approximately, how many mutants do you think there would be, if all mutants for *power*() were created?

```
public static int power (int left, int right)
{
//*************************************
// Raises Left to the power of Right
// precondition : Right >= 0
// postcondition: Returns Left**Right
//*************************************
        intrslt;
        rslt = Left;
```

```
if (Right == 0)
{
        rslt = 1;
}
else
{
        for (int i = 2; i <= Right; i++)
        rslt = rslt * Left;
}
return (rslt);
}
```

Course Outcome 3 (CO3): Draw the control flow graph and data flow graph of given piece of code.

```
public static double ReturnAverage(int value[],int AS, int MIN, int MAX){
/*
Function: ReturnAverageComputes the averageof all those numbers in the input array in
the positive range [MIN, MAX]. The maximumsize of the array is AS. But, the array size
could be smaller than AS in which case the endof input is represented by -999.
*/
int i, ti, tv, sum;
doubleav;
i = 0; ti = 0; tv = 0; sum = 0;
while (ti< AS && value[i] != -999) {
ti++;
if (value[i] >= MIN && value[i] <= MAX) {
tv++;
sum = sum + value[i];
}
i++;
}
if (tv> 0)
av = (double)sum/tv;
```

```
else
av = (double) -999;
return (av);
}
```

Course Outcome 4 (CO4): Explain the following with examples.

1. Input domain modelling.
2. All Combinations Coverage (ACoC)
3. Each Choice Coverage (ECC)
4. Pair-wise Coverage
5. T-wise Coverage
6. Base Choice Coverage
7. Multiple Base Choices Coverage.

Course Outcome 5 (CO5): Draw the symbolic execution tree for the following program code and explain the symbolic execution of testme (α1, α2).

```
1. int twice (int v) {
2.  return 2 ∗ v;
3.  }
4.  void testme (int x, int y ) {
5.  z = twice ( y);
6.  if ( z == x ){
7.  if ( x > y + 10)
8. ERROR;
9.  }
10. }
11. int main() {
12.  x = sym input();
13.  y = sym input();
14. testme ( x , y);
15.  return(0);
16. }
```

Model Question Paper

QP CODE: **PAGES: 4**

Reg No:_____ **Name :_____**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SIXTH SEMESTER B.TECH DEGREE EXAMINATION(MINOR), MONTH & YEAR

Course Code: CST 382

Course Name: Introduction to Software Testing

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

PART A

Answer all Questions. Each question carries 3 Marks

1. Explain the differences between Validation and Verification.
2. Explain the differences between Fault, Error, and Bug?
3. Define Ground string, Mutation score, and Mutants.
4. What are the functions of Test driver and Test stubs in dynamic unit testing?
5. Define Node coverage, Edge coverage and Prime path coverage in a control flow graph.
6. What are du paths and du pairs in a data flow graph?
7. Explain the two approaches in input domain modelling.
8. Explain the difference between Equivalence Class Partitioning and Boundary Value Analysis.
9. Briefly explain three techniques of Grey box testing.
10. Explain the concept of symbolic execution with the help of a toy example.

(10x3=30)

Part B

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

11. (a) Explain the following types of testing
(i) Black Box testing (ii) White Box testing (iii) Grey Box testing **(14)**

(iv) Unit testing (v) Integration testing (vi) System testing (vii) Acceptance testing

OR

12. (a) Explain the following coverage criterias based on the code fragment given below. (i) Functional coverage (ii) Statement coverage (iii)Conditional coverage (iv)Branch coverage **(8)**

```
int foo (int x, int y){
    int z = 0;
    if ((x > 0) && (y > 0)){
    z = x;}
    return z;
    }
```

(b) Write positive and negative test cases for an ATM Machine? **(6)**

13. (a) Explain Dynamic unit test environment with a neat figure. **(8)**

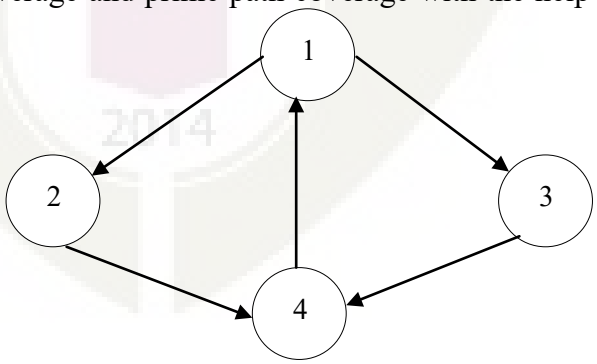
(b) Explain the major difference between control flow testing and data flow testing. **(6)**

OR

14. Explain seven types of mutation operators with neat examples. **(14)**

15. (a) Explain touring, side trips and detours with a neat example. **(7)**

(b) Explain simple path coverage and prime path coverage with the help of CFG given below. **(7)**



OR

16. (a) Draw CFG fragment for

(i) Simple *if* (ii) Simple *while* loop (iii) Simple *for* loop **(7)**

(b) Explain the following concepts with examples. **(7)**

(i)Call graph (ii) Inheritance graph (iii) Coupling du-pairs

17. (a) What are the four important steps in functional testing? **(7)**

(b) Briefly explain input domain modelling approaches. **(7)**

OR

18. (a) Consider the triangle classification program with a specification: **(6)**

The program reads floating values from the standard input. The three values *A*, *B*, and *C* are interpreted as representing the lengths of the sides of triangle. The program then prints a message to the standard output that states whether the triangle, if it can be formed, is scalene, isosceles, equilateral, or right angled. Determine the following for the above program:

(i) For the boundary condition $A + B > C$ case (scalene triangle), identify test cases to verify the boundary.

(ii) For the boundary condition $A = C$ case (isosceles triangle), identify test cases to verify the boundary.

(iii) For the boundary condition $A = B = C$ case (equilateral triangle), identify test cases to verify the boundary.

(b) Develop a decision table to generate test cases for this specification. **(8)**

19. (a) Explain the importance of grey box testing, its advantages and disadvantages. **(9)**

(b) Explain the concept of symbolic execution tree. **(5)**

OR

20. (a) Consider the code fragment given below: - **(7)**

1. POWER: PROCEDURE(X, Y);
2. Z ← 1;
3. J ← 1;
4. LAB: IF Y ≥ J THEN

5. DO; Z← Z * X;
6. J ← J + 1;
7. GO TO LAB; END;
8. RETURN (Z) ;
9. END;

a) Explain Symbolic execution of POWER (αl, α2).

(b) Explain Execution tree for POWER (αl, α2) in the above code fragment. **(7)**

TEACHING PLAN

Index	Topics	No. of Hours (45)
	Module 1 (Introduction to Software Testing) 9 Hours	
1.1	Some Popular Errors– Ariane 5, Therac 25, Intel Pentium Bug.	1 Hour
1.2	What is Software testing? Why should it be tested? Software Quality, Role of Testing.	1 Hour
1.3	Testing Process - Level 0 thinking, Level 1 thinking, Level 2 thinking, Level 3 thinking, Level 4 thinking.	1 Hour
1.4	Software Testing Terminologies- Verification, Validation and Testing, Faults, Error and Bug, Test cases, Coverage Criteria.	1 Hour
1.5	Types of Testing- Unit testing, integration testing, System testing, Acceptance testing, Beta testing	1 Hour
1.6	Functional testing, Stress testing	1 Hour
1.7	Performance testing, Usability testing and Regression testing.	1 Hour
1.8	Testing Methods - Black Box testing	1 Hour
1.9	Grey Box testing.	1 Hour
Module 2 (Unit testing) 8 Hours		

2.1	Concept of Unit testing.	1 Hour
2.2	Static Unit testing.	1 Hour
2.3	Dynamic Unit testing - Control Flow testing, Data Flow testing	1 Hour
2.4	Domain testing, Functional Program testing.	
2.5	Mutation testing - Mutation and Mutants, Mutation operators, Mutation score.	1 Hour
2.6	Junit - Framework for Unit testing.	1 Hour
2.7	Case Study - Mutation testing using Junit	1 Hour
2.8	Case Study - Mutation testing using Muclipse	1 Hour
Module 3 (Unit Testing:- White Box Approaches) 10 Hours		
3.1	Overview of Graph Coverage Criteria	1 Hour
3.2	Structural Graph Coverage Criteria - Node/vertex coverage, Edge coverage, Edge pair coverage, Path coverage	1 Hour
3.3	Complete path coverage, Prime path coverage, Complete round trip coverage, Simple round trip coverage.	1 Hour
3.4	Data Flow Criteria - du paths, du pairs	1 Hour
3.5	Subsumption Relationships among Graph Coverage Criteria.	1 Hour
3.6	Graph Coverage for Source Code - Control Flow Graphs (CFG) for code, CFG: If statement, CFG: If statement with return, CFG: Switch-case, CFG: Loops, CFG: Exceptions (try-catch). Example program - Statistics	1 Hour
3.7	Graph Coverage for Design Elements - Call graphs and classes, Class inheritance testing: Coverage criteria, Coverage criteria on inheritance graph,	1 Hour

3.8	Data flow at the design level, Inter-procedural DU pairs, Coupling du-pairs example. Example - Quadratic Root	1 Hour
3.9	Case Study - Graph Based testing using JUnit Framework. (Lecture 1)	1 Hour
3.10	Case Study - Graph Based testing using JUnit Framework. (Lecture 2)	1 Hour
Module 4 (Unit Testing:- Black Box Approaches) 9 Hours		
4.1	Domain Testing / Input Space Partitioning - Partitions of a set.	1 Hour
4.2	Input domain modelling - Interface-based approach, Functionality-based approach.	1 Hour
4.3	Identifying values.	1 Hour
4.4	Multiple partitions of the input domain - All Combinations Coverage (ACoC), Each Choice Coverage (ECC), Pair-wise Coverage, T-wise Coverage, Base Choice Coverage, Multiple Base Choices Coverage.	1 Hour
4.5	TriTyp example.	1 Hour
4.6	Functional Testing - Functional Testing Concepts of Howden. Important Steps.	1 Hour
4.7	Types of Functional testing - Equivalence Class Partitioning, Boundary Value Analysis	1 Hour
4.8	Decision Tables, Random Testing.	1 Hour
4.9	Case Study - Black Box testing approaches using JUnit.	1 Hour
Module 5 (Grey Box Testing Approaches) 9 Hours		
5.1	Introduction to Grey Box testing - Why Grey Box testing, Gray Box Methodology, Advantages and Disadvantages.	1 Hour
5.2	Techniques of Grey Box Testing - Matrix Testing, Regression Testing,	1 Hour

5.3	Orthogonal Array Testing or OAT, Pattern Testing.	1 Hour
5.4	An Introduction to Pex - Parameterized Unit Testing, The Testing Problem.	1 Hour
5.5	Symbolic Execution – Example, Symbolic execution tree.	1 Hour
5.6	PEX application.	1 hour
5.7	Case Study – PEX (Lecture 1)	1 Hour
5.8	Case Study – PEX (Lecture 2)	1 Hour
5.9	Case Study – PEX (Lecture 3)	1 Hour

CST 384	CONCEPTS IN DEEP LEARNING	Category	L	T	P	Credits	Year of Introduction
		VAC	3	1	0	4	2019

Preamble:

This course aims to introduce the learner to an overview of the concepts and algorithms involved in deep learning. Deep learning is a subfield of machine learning, a subfield of artificial intelligence. Basic concepts and application areas of machine learning, deep networks, convolutional neural network and recurrent neural network are covered here. 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: Sound knowledge in Basics of linear algebra and probability theory.

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**	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
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 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

INTRODUCTION TO DEEP LEARNING

(General Instructions: Instructors are to introduce students to any one software platform and demonstrate the working of the algorithms in the syllabus using suitable use cases and public datasets to give a better understanding of the concepts discussed. Tutorial hour may be used for this purpose)

Module-1 (Introduction)

Key components - Data, models, objective functions, optimization algorithms, Learning algorithm. 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, Structure of CNN, Convolution and Pooling as an infinitely strong prior, variants of convolution functions, structured outputs, data types, efficient convolution algorithms. Practical challenges of common deep learning architectures- early stopping, parameter sharing, dropout. Case study: AlexNet, VGG, ResNet.

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, common word embedding: continuous Bag-of-Words, Word2Vec, global vectors for word representation (GloVe). Research Areas – autoencoders, representation learning, boltzmann machines, deep belief networks.

Text Book

1. Ian Goodfellow, YoshuaBengio, 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, Aggarwal, Charu C., c Springer International Publishing AG, part of Springer Nature 2018

Reference Books

1. Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks by Russell Reed, Robert J MarksII, A Bradford Book,2014
2. Practical Convolutional Neural Networks by MohitSewak, Md. Rezaul Karim, PradeepPujari,Packt Publishing 2018
3. Hands-On Deep Learning Algorithms with Python by SudharsanRavichandran,Packt Publishing 2019
4. Deep Learning with Python by Francois Chollet,Manning Publications Co.,2018

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

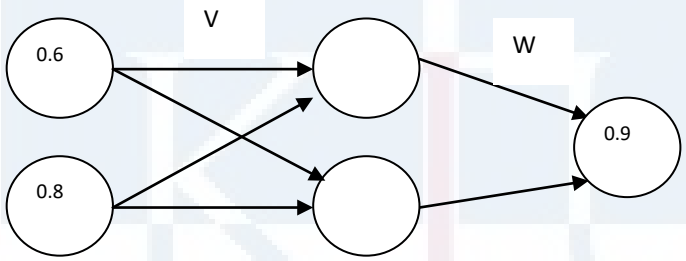
1. Compare regression and classification.
2. Define supervised learning? Distinguish between regression and classification.
3. Discuss the different learning approaches used in machine learning.

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?

Course Outcome 3 (CO3):

1. Update the parameters V_{11} in the given MLP using back propagation with learning rate as 0.5 and activation function as sigmoid. Initial weights are given as V_{11}= 0.2, V_{12}=0.1, V_{21}=0.1, V_{22}=0.3, V_{11}=0.2, W_{11}=0.5, W_{21}=0.2



2. Draw the architecture of a multi-layer perceptron.
3. Derive update rules for parameters in the multi-layer neural network through the gradient descent.

Course Outcome 4 (CO4):

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. Explain how the cell state is updated in the LSTM model from Ct-1 to Ct
4. Show the steps involved in an LSTM to predict stock prices.

Course Outcome 5 (CO5):

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.

Course Outcome 6 (CO6):

1. Development a deep learning solution for problems in the domain i) natural language processing or ii Computer vision (Assignment
2. Illustrate the workings of the RNN with an example of a single sequence defined on a vocabulary of four words.

Model Question Paper

QP CODE: PAGES:4

Reg No:_____

Name:_____

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
SIXTH SEMESTER B.TECH DEGREE EXAMINATION(MINOR), MONTH & YEAR

Course Code: CST 384
Course Name: CONCEPTS IN 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 of tasks 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 **(4)**

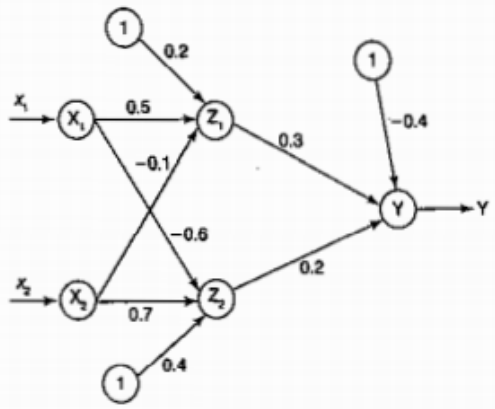
traditional machine learning models

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 backpropgation 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

CONCEPTS IN DEEP LEARNING (45 Hours)		
Module 1 : Introduction (9 hours)		
1.1	Key components - Data, models, objective functions, optimization algorithms. (TB2: Section 1.1-1.2)	1 hour

1.2	Learning algorithm (TB1: Section 5.1), Supervised learning- regression, classification (TB2: Section 1.3.1)	1 hour
1.3	tagging, web search, page ranking (TB2: Section 1.3.1)	1 hour
1.4	Recommender systems, Sequence learning, Unsupervised learning, Reinforcement learning(TB2: Section 1.3.2-1.3.4)	1 hour
1.5	Historical Trends in Deep Learning (TB1: Section 1.2).	1 hour
1.6	Concepts: over-fitting, under-fitting, hyperparameters and validation sets. (TB1: Section 5.2-5.3)	1 hour
1.7	Concepts: Estimators, bias and variance. (TB1: Section 5.4)	1 hour
1.8	Demonstrate the concepts of supervised learning algorithms using a suitable platform.	1 hour
1.9	Demonstrate the concepts of unsupervised using a suitable platform.	1 hour
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 hour
2.2	Multilayer perceptron (TB3: Section 1.2.2), (TB1: Section 6.1,6.3)	1 hour
2.3	Activation functions- Sigmoid, tanh, Softmax, ReLU, leaky ReLU (TB3: Section 1.2.1.3 - 1.2.1.5)	1 hour
2.4	Architecture design (TB1: Section 6.4, TB3: Section 1.6)	1 hour
2.5	Chain rule, back propagation (TB3: Section 1.3)	1 hour

2.6	Gradient based learning (TB1: Section 6.2)	1 hour
2.7	Gradient based optimization (TB1: Section 4.3)	1 hour
2.8	Linear least squares using a suitable platform. (TB1: Section 4.5)	1 hour
2.9	Building ML Algorithms and Challenges (TB3: 1.4, TB1: 5.10-5.11)	1 hour
Module 3 :Convolution Neural Network (10 hours)		
3.1	Convolution operation, Motivation, pooling (TB1:Section 9.1-9.3)	1 hour
3.2	Structure of CNN (TB3: Section 8.2)	1 hour
3.3	Convolution and Pooling as an infinitely strong prior (TB1: Section 9.4)	1 hour
3.4	Variants of convolution functions – multilayer convolutional network, tensors, kernel flipping, downsampling, strides and zero padding. (TB1: Section 9.5)	1 hour
3.5	Variants of convolution functions - unshared convolutions, tiled convolution, training different networks. (TB1: Section 9.5)	1 hour
3.6	Structured outputs, data types (TB1: Section 9.6-9.7)	1 hour
3.7	Efficient convolution algorithms. (TB1: Section 9.8,9.10)	1 hour
3.8	Practical challenges of common deep learning architectures- early Stopping (TB3: 4.6)	1 hour
3.9	Practical challenges of common deep learning architectures- parameter sharing, drop-out (TB3: Section 4.9, 4.5.4)	1 hour
3.10	Case Study: AlexNet,VGG, ResNet. (TB3: Section 8.4.1-8.4.3,8.4.5)	1 hour

Module 4 :Recurrent Neural Network (8 hours)		
4.1	Computational graphs (TB1: Section 10.1)	1 hour
4.2	RNN (TB1: Section 10.2-10.3)	1 hour
4.3	Encoder – decoder sequence to sequence architectures. (TB1: Section 10.4)	1 hour
4.4	Deep recurrent networks (TB1: Section 10.5)	1 hour
4.5	Recursive neural networks , Modern RNNs, (TB1: Section 10.6, 10.10)	1 hour
4.6	LSTM and GRU (TB1: Section 10.10, TB3: Section 7.5-7.6)	1 hour
4.7	Practical use cases for RNNs. (TB1: Section 11.1-11.4)	1 hour
4.8	Demonstrate the concepts of RNN using a suitable platform.	1 hour
Module 5 : Applications and Research (9 hours)		
5.1	Computer vision. (TB1: Section 12.2)	1 hour
5.2	Speech recognition. (TB1: Section 12.3)	1 hour
5.3	Natural language processing. (TB1: Section 12.4)	1 hour
5.4	Common Word Embedding -: Continuous Bag-of-Words, Word2Vec (TB3: Section 2.6)	1 hour
5.5	Common Word Embedding -: Global Vectors for Word Representation(GloVe) (TB3: Section 2.9.1- Pennigton 2014)	1 hour
5.6	Brief introduction on current research areas- Autoencoders, Representation learning. (TB3: Section 4.10)	1 hour

5.7	Brief introduction on current research areas- representation learning. (TB3: Section 9.3)	1 hour
5.8	Brief introduction on current research areas- Boltzmann Machines, Deep belief networks. (TB1: Section 20.1, TB3 Section 6.3)	1 hour
5.9	Brief introduction on current research areas- Deep belief networks. (TB1: Section 20.3)	1 hour

CST 386	WIRELESS NETWORKS AND IoT APPLICATIONS	Category	L	T	P	Credit	Year of Introduction
		VAC	3	1	0	4	2019

Preamble:

This course equips the learners with fundamental wireless technologies for the Internet of Things(IoT) and the IoT ecosystem. It covers the underlying concepts in wireless networks, communication mechanisms, protocols, hardware, software, and the cloud platforms for IoT. The students will be able to design smart IoT applications for real world problems..

Prerequisite: Sound knowledge in Data Communication, Computer Networks and Programming in C

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

CO1	Recognize wireless technologies required for IoT ecosystem **(Cognitive Knowledge Level : Understand)**
CO2	Perceive the concept of IoT and M2M architecture, IoT examples, and Data Management in IoT **(Cognitive Knowledge Level :Apply)**
CO3	Outline the hardware components used in IoT including Sensors, Actuators and development boards **(Cognitive Knowledge Level : understand)**
CO4	Explain the software components of IoT **(Cognitive Knowledge Level :Understand)**
CO5	Demonstrate the protocols used in IoT and build IoT Programs **(Cognitive Knowledge Level : Apply)**
CO6	Build IoT-based smart real-time applications such as Smart Healthcare, Smart Agriculture, Smart Environment and Smart Home **(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 teamwork
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

Blooms Category	Continuous Assessment Tests		End Semester Examination Marks
	Test 1 (Percentage)	Test 2 (Percentage)	
Remember	**30**	**30**	**30**
Understand	**50**	**40**	**40**
Apply	**20**	**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 anyone. Each question can have a maximum 2 subdivisions and carries 14 marks.

Syllabus

Module- 1 (Introduction to IoT and wireless technologies required for IoT)

Internet of Things, Role of Things and the Internet, Wireless IoT. Wireless Networks - Network Topologies, Types of Networks. Role of Wireless Standards in IoT. Protocol Stack - OSI Model, TCP/IP Model, IEEE 802 Reference Model, Protocols for Wireless IoT. Bluetooth - Transceiver, Frequency Channels, Typical Range, Access and Spread Spectrum, Modulation and Data Rate, Error Correction and Detection, Network Topology. ITU G.9959, Zwave, IEEE 802.15.4, Zigbee Specification, Thread, WiFi, 6LowPAN, IPv6, LoRaWAN.

Module- 2 (IoT architecture, Data and Device management)

Internet of Things - IoT Architectural View, Technology Behind IoT - Server End Technology, Sources of Internet of Things, M2M Communication. IoT Application Areas. IoT Examples. IoT Data Management - Device Management Gateways. Design Principles for Web Connectivity - Web Communication Protocols for Connected Devices, Web Connectivity for Connected Devices using Gateways. Internet Connectivity Principles – Internet Connectivity, Internet based communication, IP addressing in the IoT.

Module- 3 (Data Acquiring and Enabling Technologies)

Data Acquiring and Storage for IoT Sevices- Organization of Data, Big data, Acquiring Methods, Management Techniques, Analytics, Storage Technologies. Cloud Computing for Data storage - IoT Cloud based Services using Xively, Nimbits, and Other Platforms. Sensor Technologies for IoT Devices - Sensor Technology, Participatory Sensing, Industrial IoT and Automotive IoT, Actuators for Various Devices, Sensor Data Communication Protocols, Wireless Sensor network Technology

Module-4 (Prototyping the Embedded Devices for IoT)

Embedded Computing Basics, Embedded Hardware Unit. Embedded Platforms for Prototyping - Arduino, Intel Galileo, Intel Edison, Raspberry Pi, BeagleBone, mBed. Prototyping and Designing the Software for IoT Applications- Introduction, Prototyping Embedded DeviceSoftware- Programming using Arduino, Programming for an Arduino Controlled Traffic Control Lights at a Road Junction, Basic Arduino Programs to Blink LED, Find the Distance using Ultrasonic Sensor, Estimate Room Temperature, Measuring Soil Moisture Level

Module 5 (Business Models and Case Studies)

Business Models and Processes using IoT. Value Creation in the Internet of Things. Cloud PaaS- Xively, Nimbits, IBM Bluemix, CISCO IoT, AWS IoT, TCS Connected AWS Platform, Case studies- Smart Home, Smart Environment, Smart healthcare, Smart agriculture

Text Books

1. Daniel Chew, "Wireless Internet of Things -A Guide to the lower layers", IEEE Standards and Association, IEEE Press, Wiley
2. Rajkamal, "Internet of Things : Architecture and Design Principles", McGraw Hill (India) Private Limited.

References

1. ArshadeepBahga, Vijay Madisetti, "Internet of Things: A hands-on approach", University Press, 2015 (First edition)
2. Dieter Uckelmann, Mark Harrison, Michahelles Florian (Ed.), Architecting the internet of things, Springer, 2011
3. Dr. Ovidiu Vermesan, Dr. Peter Friess, Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems, River Publishers, 2013
4. Simon Monk, "Programming Arduino: Getting Started with Sketches", McGraw Hill Publications

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Compare Bluetooth and Bluetooth LE power classes
2. Demonstrate Zigbee Specification Protocol Stack

Course Outcome 2 (CO2):

1. What are the major components of IOT system? Briefly explain each
2. Correlate M2M architectural Levels with IOT architectural Levels

Course Outcome 3 (CO3):

1. Describe the use of GPIO pins ?
2. What are actuators ? Mention the roles of actuators in IoT systems

Course Outcome 4(CO4):

1. Identify the role of HBase in Hadoop File System
2. Differentiate Edge computing and Distributed computing
3. Illustrate open protocols, tools and frameworks generally used in M2M

Course Outcome 5(CO5):

1. What do you mean by Arduino sketches?
2. Write an Arduino program to blink LED

Course Outcome 6(CO6):

1. How IoT technology helps TELEMEDICINE in India?
2. How soil moisture can be detected in Smart Agriculture?

Model Question Paper

QP CODE: PAGES :2

Reg No: _____

Name: _____

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SIXTH SEMESTER B.TECH DEGREE EXAMINATION(MINOR), MONTH & YEAR

Course Code: CST 386

Course Name: WIRELESS NETWORKS AND IoT APPLICATIONS

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

PART A

Answer All Questions. Each Question Carries 3 Marks

1. Illustrate Role of *things* and *internet* in IoT
2. What is Bluetooth? Explain the range and frequency channels of Bluetooth?
3. List any three the features of Constrained Application Protocol (COAP).
4. Compare Raspberry Pi and BeagleBoard boards.
5. Identify the role of HBase in Hadoop File System.
6. Differentiate Edge computing and Distributed computing.
7. Give an example of Raspberry Pi applications for Industrial IoT.
8. What are the on-board functional units in Intel Galileo?
9. Interpret the concept of value creation in IoT.

10. Explain the use of PaaS in IoT Smart applications with any three examples. **(10x3=30)**

Part B

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

11. (a) Compare various Network topologies used in Wireless Networks. **(8)**

(b) Describe the following wireless technologies on i) *Zigbee* ii)*WiFi* iii) *Thread.* **(6)**

OR

12. (a) Explain protocol stacks used in wireless networks for IoT applications. **(8)**

(b) Illustrate the Architectural design of LoRaWAN. **(6)**

13. (a) Define M2M. Explain M2M architecture. Correlate M2M architectural levels with IoT architectural levels. **(8)**

(b) Compare SOAP and REST protocols. **(6)**

OR

14. (a) Summarize different Online Transactions and Processing techniques. **(8)**

(b) Identify the functions of Device-Management Gateway . **(6)**

15. (a) Define actuators ? Describe the roles of actuators in IoT systems. **(8)**

(b) Explain the usage contexts of analog sensors and digital sensors. **(6)**

OR

16. (a) How data collection, storage & computing services done using Nimbits? **(10)**

(b) List any four features of Xively. **(4)**

17. (a) What do you mean by Arduino sketches? **(4)**

(b) Write an Arduino program to blink LED **(10)**

OR

18. (a) Demonstrate an example of Raspberry Pi applications for Industrial IoT. **(10)**

(b) Compare the features of Arduino-R3 and Arduino Yun boards. **(4)**

19. (a) Explain various tasks of a smart irrigation monitoring service. **(8)**

(b) Demonstrate the tasks of Soil-Moisture monitoring service. **(6)**

OR

20. (a) a) Mr. Kiran Mathew has been a chronic diabetic patient for the past few years. He was under regular check up at the hospital every two weeks. All of a sudden the pandemic like COVID-19 arises in the country and the government issues a lockdown for a period of two months. Illustrate how Mr. Kiran can be monitored by the health care worker using intelligent healthcare techniques. **(10)**

(b) Mention any four sensors used in smart healthcare **(4)**

TEACHING PLAN

No	Contents	No of Lecture Hrs(45)
Module – 1 (Introduction to IoT and wireless technologies required for IoT) (8 hrs) (TB-1, Chapter 1...)		
1.1	Internet Of Things, Role of things and internet ,Wireless IoT	1
1.2	Wireless Networks- Network Topologies-Types of Networks,Role of	1

	Wireless standards in IoT	
1.3	Protocol Stack-OSI Model- TCP/IP Model-IEEE 802 reference model	1
1.4	Protocols for Wireless IoT-Bluetooth-Transceiver, Frequency Channels-Typical Range, Access and Spread Spectrum, Modulation and Data Rate	1
1.5	Error Correction and Detection-Network Topology.	1
1.6	ITU G.9959, Zwave, IEEE 802.15.4, Zigbee Specification	1
1.7	Thread, Wifi, 6LowPAN, IPv6	1
1.8	LoRaWAN	1
Module- 2 (IOT architecture, Data and Device management) (9hrs)		
2.1	Internet of Things -IoT Architectural view	1
2.2	Technology Behind IOT-Server End Technology,Sources of Internet of Things	1
2.3	M2M Communication.	1
2.4	IoT Application Areas. IOT Examples.	1
2.5	IoT Data Management, Device Management Gateways.	1
2.6	Design Principles for Web Connectivity	1
2.7	Web communication protocols for connected devices,	1
2.8	Web connectivity for connected devices using Gateways.	1
2.9	Internet connectivity Principles – Internet Connectivity, Internet based communication, IP addressing in the IoT.	1
Module- 3 (Data Acquiring and Enabling Technologies (8 hrs)		
3.1	Data acquiring and storage for IoT devices- Organization of Data, Big data	1
3.2	Acquiring methods, management techniques, Analytics, Storage technologies.	1
3.3	Cloud computing for Data storage-IoT Cloud based services using Xively,	1

	Nimbits, and other platforms.	
3.4	Cloud computing-Nimbits	1
3.5	Sensor Technologies for IoT Devices-Sensor Technology, Participatory sensing	1
3.6	Industrial IoT and Automotive IoT	1
3.7	Actuators for various devices, Sensor data communication protocols	1
3.8	Wireless Sensor network Technology	1
Module 4(Prototyping the Embedded Devices for IoT)(9hrs)		
4.1	Introduction, Embedded Computing Basics, Embedded Hardware Unit.	1
4.2	Embedded Platforms for Prototyping-Arduino, Intel Galileo	1
4.3	Intel Edison, Raspberry Pi, BeagleBone, mBed	1
4.4	Prototyping and designing the software for IoT applications-Introduction, Prototyping embedded device software	1
4.5	Prototyping and designing the software for IoT applications-Introduction, Prototyping embedded device software	1
4.6	Programming concepts in Arduino	1
4.7	Programming for an arduino controlled traffic control lights at a road junction	1
4.8	Basic Arduino programs to blink LED, Find the distance using ultrasonic sensor	1
4.9	Estimate room temperature, Measuring soil moisture level	1
Module 5 (higher level protocols and case studies)(9 hrs)		
5.1	Business Models and Processes using IOT, Value creation in the Internet of Things.	1

5.2	Xively, Nimbits, IBM Bluemix	1
5.3	CISCO IoT, AWS IoT, TCS Connected AWS Platform	1
5.4	Case Study- Smart Environment	1
5.5	Case Study- Smart Environment	1
5.6	Case study Smart Home	1
5.7	Case study Smart Home	1
5.8	Case study Smart healthcare (Lecture I)	1
5.9	Case study Smart healthcare (Lecture II)	1
5.10	Case study -Smart agriculture (Lecture I)	1
5.11	Case study -Smart agriculture (Lecture II)	1

SEMESTER VI

HONOURS

CST 394	NETWORK SECURITY	Category	L	T	P	Credits	Year of Introduction
		VAC	3	1	0	4	2019

Preamble:

The purpose of this course is to create a better understanding of the network security concepts. This course covers network security standards, email security services, web security mechanisms, firewalls and wireless security mechanisms. This course helps the learner to gain insight into the key aspects of secure network communication and enables to apply in real-life scenarios.

Prerequisite: A sound background in Number Theory and Cryptographic Algorithms.

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

CO#	Course Outcomes
CO1	Identify the key aspects of security, intrusion detection systems and digital signature schemes **(Cognitive Knowledge Level: Apply)**
CO2	Explain the security standards used in network communication (**Cognitive Knowledge Level:Understand)**
CO3	Identify the mechanisms in email security services (**Cognitive Knowledge Level: Apply)**
CO4	Summarize the protocols used to provide web security (**Cognitive Knowledge Level: Understand)**
CO5	Explain the fundamental concepts of wireless network security and firewalls **(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**	Lifelong learning

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination (%)
	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 **: 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 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 (Network Security Basics)

Introduction to network security - Security requirements, Challenges of security, Network security model. Malicious programs – Worms, Viruses, Trojans, Spyware, Adware. Intrusion Detection Systems (IDS) - Uses, Techniques. Digital signatures - ElGamal, Schnorr, Digital Signature Standard (DSS).

Module – 2 (Network Security Standards)

Kerberos v4 – Configuration, Authentication, Encryption, Message formats. Kerberos v5 – Cryptographic algorithms, Message formats. Public Key Infrastructure (PKI) – Trust models, Revocation. Real-time communication security – Perfect Forward Secrecy (PFS), Denial-of-Service protection, Endpoint identifier hiding, Live partner reassurance. Internet Protocol Security (IPSec) - Authentication Header (AH), Encapsulating Security Payload (ESP), Internet Key Exchange (IKE) phases.

Module – 3 (Email Security)

Introduction to email security - Security services for email, Establishing keys, Privacy, Authentication, Message integrity, Non-repudiation. Privacy Enhanced Mail (PEM) – Encryption, Source authentication and integrity protection, Message formats. Secure/Multipurpose Internet Mail Extensions (S/MIME) – Messages, Differences from PEM. Pretty Good Privacy (PGP) - Encoding, Certificate and key revocation, Anomalies, Object formats.

Module – 4 (Web Security)

Introduction to web security - Web security considerations, Threats. Secure Sockets Layer (SSL) – Architecture, Protocols, Transport Layer Security (TLS) – Differences from SSL. Hypertext Transfer Protocol Secure (HTTPS) – Connection initiation, Closure. Secure Shell (SSH) – Transport layer protocol, User authentication protocol, Connection protocol.

Module – 5 (Wireless Network Security and Firewalls)

IEEE 802.11 Wireless LAN - Network components, Architectural model, Services. IEEE 802.11i wireless LAN security - Services, Phases of operation. Wired Equivalent Privacy (WEP), Wi-Fi Protected Access (WPA), WPA2, Wireless Application Protocol (WAP) – Services, Protocol architecture. Firewalls – Need for firewalls, Packet filters, Circuit-level firewalls, Application layer firewalls.

Text Books

1. C. Kaufman, R. Perlman and M. Speciner, “Network Security: Private Communication in a Public World”, 2/e, PHI.
2. William Stallings, “Cryptography and Network Security Principles and Practice”, 5/e, Pearson

Education Asia.

References

1. Behrouz A. Forouzan, Debdeep Mukhopadhyay, "Cryptography and Network Security", 3/e, Tata McGraw Hill.
2. Tyler Wrightson, "Wireless Network Security A Beginner's Guide", 2012, Tata McGraw Hill.
3. William Stallings, "Network Security Essentials: Applications and Standards", 4/e, Prentice Hall.
4. Schiller J., Mobile Communications, 2/e, Pearson Education.
5. Roberta Bragg et. al., "Network Security: The Complete Reference", Tata McGraw Hill.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Using the Schnorr digital signature scheme, let $q = 83$, $p = 997$ and $d = 23$. Find values for e_1 and e_2.
2. The Digital Signature Algorithm (DSA) specifies that if the signature generation process results in a value of zero, a new value of k should be generated and the signature should be recalculated. Give reason.

Course Outcome 2 (CO2):

1. In Kerberos v4, the authenticator field is not of security benefit when asking the Key Distribution Center (KDC) for a ticket for Bob, but useful when logging in as Bob. Give reasons for your answer.
2. How does the stateless cookie protocol provide clogging protection?

Course Outcome 3 (CO3):

1. If Alice is sending an ENCRYPTED message, she first signs the message digest with her private key and then encrypts the message digest with the pre-message secret key. Why this last encryption was considered necessary for encrypted messages and not for MIC-CLEAR or MIC-ONLY?
2. Which security services are considered desirable in the following cases? (i) Sending a purchase order (ii) Sending a ransom note. (iii) Sending a mission description to security officials.
3. Explain the security mechanism used in Gmail communication.

Course Outcome 4 (CO4):

1. Is it possible in SSL for the receiver to reorder SSL record blocks that arrive out of order? If so, how it can be done? If not, why?
2. Describe any five web security threats, their consequences and countermeasures.

Course Outcome 5 (CO5):

1. Explain the security areas addressed by IEEE 802.11i.
2. Describe the advantages and disadvantages of application layer firewalls.

Model Question Paper

QP CODE:
Reg. No:_____
Name:_____ **PAGES : 3**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SIXTH SEMESTER B.TECH. DEGREE (HONORS) EXAMINATION, MONTH &YEAR
Course Code: CST 394

Course Name: Network Security

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

PART A

Answer all Questions. Each question carries 3 Marks

1. Distinguish between signature-based and anomaly-based intrusion detection techniques.

2. A trusted third party is considered as a main component in a network security model. Why?

3. How is endpoint identifier hiding achieved in real-time communication?

4. Show how encryption is used to provide privacy and integrity in Kerberos v5.

5. End-to-end privacy is essential for e-mail security. How is this achieved?

6. List the four steps for preparing an EnvelopedData MIME entity.

7. Show the operation of a Secure Sockets Layer (SSL) Record protocol.

8. For Secure Shell (SSH) packets, what is the advantage of not including the MAC in the scope of packet encryption?

9. List the three security services provided by IEEE 802.11i.

10. Define the terms Access Point, Basic Service Set, Extended Service Set.

(10x3=30)

Part B

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

11. (a) Using the ElGamal scheme, let p = 881 and d = 700, find values for e1 and e2. Choose r = 17. Find the value of S1 and S2 if M = 400. **(8)**

(b) Explain the requirements and challenges of network security. **(6)**

OR

12. (a) In ElGamal, Schnorr and DSS, what happens if an attacker can find the value of random secret key used by the signer? Also, what happens if a user uses the same value of random secret key to sign two messages? Explain your answer for each scheme separately. **(8)**

(b) Explain the network security model with the help of a neat diagram. **(6)**

13. (a) Alice wishes to log into Bob's workstation remotely. List the steps involved in this communication if Kerberos v4 is used. **(7)**

(b) How does Diffie-Hellman technique provide perfect forward secrecy using signature keys? **(7)**

OR

14. (a) Explain the algorithm for Message Authentication Code (MAC) calculation and verification in Kerberos v5 rsa-md5-des. **(8)**

(b) Compare the aggressive mode and main mode of Phase 1 Internet Key Exchange (IKE). **(6)**

15. (a) Describe the different methods by which authentication of source is performed in email communication. **(7)**

(b) Explain the Signed data and Clear-signed data functions provided by S/MIME. **(7)**

OR

16. (a) Explain the advantages of Pretty Good Privacy (PGP) over Privacy Enhanced Mail (PEM). **(7)**

(b) Define non-repudiation. Describe the different ways by which it is implemented in email communication. **(7)**

17. (a) Describe the significance of pseudo-random function of Transport Layer Security. **(7)**

(b) Explain the four different phases of Secure Sockets Layer (SSL) HandshakeProtocol. **(7)**

OR

18. (a) Describe how connection initiation and connection closure is done in Hyper Text Transfer Protocol Secure (HTTPS). **(7)**

(b) Illustrate the sequence of events in Secure Shell (SSH) transport layer protocol packet exchanges. **(7)**

19. (a) Explain the Discovery phase and Authentication phase of IEEE 802.11i operation. **(7)**

(b) Why are firewalls needed? Compare the features of packet filters and circuit level firewalls. **(7)**

OR

20. (a) Explain the two authentication methods used in Wired Equivalent Privacy (WEP). **(7)**

(b) Describe the three transaction classes provided by Wireless Transaction Protocol. **(7)**

Teaching Plan

No	Contents	No of Lecture Hrs
Module - 1 (Network Security Basics) (7 hrs)		
1.1	Security requirements, Challenges of security	1
1.2	Network security model	1
1.3	Worms, Viruses, Trojans, Spyware, Adware	1
1.4	Intrusion Detection Systems (IDS) uses, Techniques	1
1.5	ElGamal digital signature	1
1.6	Schnorr digital signature	1
1.7	Digital Signature Standard (DSS)	1
Module - 2 (Network Security Standards) (12 hrs)		
2.1	Kerberos v4 configuration, Authentication	1
2.2	Kerberos v4 encryption	1
2.3	Kerberos v4 message formats	1
2.4	Kerberos v5 cryptographic algorithms – rsa-md5-des, des-mac, des-mac-k	1
2.5	Kerberos v5 cryptographic algorithms - rsa-md4-des, rsa-md4-des-k, Encryption for privacy and integrity	1
2.6	Kerberos v5 message formats	1
2.7	Public Key Infrastructure (PKI) trust models	1
2.8	PKI revocation	1
2.9	Perfect Forward Secrecy (PFS), Denial-of-Service protection	1
2.10	Endpoint identifier hiding, Live partner reassurance	1
2.11	Internet Protocol Security (IPSec) Authentication Header (AH), Encapsulating Security Payload (ESP)	1

2.12	Internet Key Exchange (IKE) phases	1
Module - 3 (Email Security) (9 hrs)		
3.1	Security services for email, Establishing keys, Privacy	1
3.2	Authentication, Message integrity, Non-repudiation	1
3.3	Privacy Enhanced Mail (PEM) encryption, Source authentication	1
3.4	PEM integrity protection, Message formats (Lecture 1)	1
3.5	PEM message formats (Lecture 2)	1
3.6	Secure/Multipurpose Internet Mail Extensions (S/MIME) – Messages, Differences from PEM	1
3.7	Pretty Good Privacy (PGP) encoding, Certificate and key revocation, Anomalies	1
3.8	PGP Object formats (Lecture 1)	1
3.9	PGP Object formats (Lecture 2)	1
Module – 4 (Web Security)(9 hrs)		
4.1	Web security considerations, Threats, Secure Sockets Layer (SSL) architecture	1
4.2	SSL protocols (Lecture 1)	1
4.3	SSL protocols (Lecture 2)	1
4.4	Transport Layer Security (TLS) differences from SSL (Lecture 1)	1
4.5	TLS differences from SSL (Lecture 2)	1
4.6	Hypertext Transfer Protocol Secure (HTTPS) connection initiation, Closure	1
4.7	Secure Shell (SSH) transport layer protocol	1
4.8	SSH user authentication protocol	1
4.9	SSH connection protocol	1

Module - 5 (Wireless Security and Firewalls) (8 hrs)		
5.1	IEEE 802.11 Wireless LAN network components, Architectural model, Services	1
5.2	IEEE 802.11i wireless LAN security services, Phases of operation (Lecture 1)	1
5.3	IEEE 802.11i phases of operation (Lecture 2)	1
5.4	Wired Equivalent Privacy (WEP), Wi-Fi Protected Access (WPA), WPA2	1
5.5	Wireless Application Protocol (WAP) services, Protocol architecture (Lecture 1)	1
5.6	WAP protocol architecture (Lecture 2)	1
5.7	Need for firewalls, Packet filters	1
5.8	Circuit-level firewalls, Application layer firewalls	1

CST 396	ADVANCED TOPICS IN MACHINE LEARNING	Category	L	T	P	Credit	Year of Introduction
		VAC	3	1	0	4	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, auto encoders, sampling methods and PAC learning. This course helps the students to provide machine learning based solutions to real world problems.

Prerequisite: Basic understanding of probability theory, linear algebra, multivariate calculus and multivariate probability theory.

CO1	Illustrate the concepts of regression and classification techniques **(Cognitive Knowledge Level: Apply)**
CO2	Demonstrate various unsupervised learning techniques **(Cognitive Knowledge Level: Apply)**
CO3	Choose suitable model parameters for different machine learning techniques and to evaluate a model performance **(Cognitive Knowledge Level: Apply)**
CO4	Explain the framework of PAC learning, basic concepts of VC dimension and non-uniform learnability **(Cognitive Knowledge Level: Understand)**
CO5	Construct Bayesian models for data and apply computational techniques to draw inferences **(Cognitive Knowledge Level: Apply)**
CO6	Illustrate the concepts of sampling algorithms, auto encoder, generative adversarial networks **(Cognitive Knowledge Level: Apply)**

Mapping of course outcomes with program outcomes

	PO1	PO 2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO1 2
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	**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 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 2 sub-divisions and carry 14 marks.

Syllabus

Module -1 (Supervised Learning)

Overview of machine learning - supervised, semi-supervised, unsupervised learning, reinforcement learning Regression algorithms: least squares linear regression, gradient descent, closed form, normal equations, regularization techniques (LASSO, RIDGE), polynomial regression. Discriminative Methods - Logistic Regression, Decision Tree Learning. Generative Methods - Naive Bayes Classifier, Gaussian Discriminant Analysis (GDA).

Module -2 (Unsupervised Learning)

Clustering - Similarity measures, Hierarchical Agglomerative Clustering, K-means partitional clustering, K-medoids clustering, Gaussian mixture models: Expectation Maximization (EM) algorithm for Gaussian mixture model.

Module -3 (Practical aspects in machine learning)

Classification Performance measures - Precision, Recall, Accuracy, F-Measure, ROC, AUC, generalisation and overfitting, cross-validation, bias-variance tradeoff, error estimation, parameter and model selection. Ensemble Methods - Bagging, Boosting, Adaboost, Random Forests.

Module -4 (Statistical Learning Theory)

Models of learnability- learning in the limit, probably approximately correct (PAC) learning. Sample complexity- quantifying the number of examples needed to PAC learn, Computational complexity of training, Sample complexity for finite hypothesis spaces, PAC results for learning conjunctions, Sample complexity for infinite hypothesis spaces, Vapnik-Chervonenkis(VC) dimension.

Module -5 (Advanced Machine Learning Topics)

Graphical models - Bayesian belief networks, Markov random fields(MRFs), Inference on chains and factor graphs, inference on clique trees. Monte Carlo methods – Basic sampling algorithms, rejection sampling, importance sampling, Markov chain Monte Carlo(MCMC), Gibbs sampling. Variational methods. Auto Encoder, Variational AutoEncoder, Generative Adversarial Networks

Textbook

1. Christopher M. Bishop. Pattern recognition and machine learning. Springer 2006.
2. Ethem Alpaydin, Introduction to Machine Learning, 2nd edition, MIT Press 2010.
3. Mohammed J. Zaki and Wagner Meira, Data Mining and Analysis: Fundamental Concepts and Algorithms, Cambridge University Press, First South Asia edition, 2016.
4. Ian Goodfellow, Yoshua Bengio and Aaron Courville. Deep Learning. MIT Press 2016.
5. Mehryar Mohri, Afshin Rostamizadeh and Ameet Talwalkar. Foundations of Machine Learning. Second edition. MIT Press 2018.
6. Tom Mitchell. Machine Learning. McGraw Hill 1997.
7. Richard O. Duda, Peter E . Hart, David G. Stork. Pattern classification, Second Edition. Wiley.
8. Jiawei Han, Micheline Kamber, Jian Pei. Data Mining Concepts and Techniques, Third Edition. Morgan Kaufmann.
9. David Foster. Generative Deep Learning - Teaching Machines to Paint, Write, Compose, and Play. O'Reilly Media, Inc., June 2019.

Reference Books

1. Kevin P. Murphy. Machine Learning: A Probabilistic Perspective. MIT Press 2012
2. Carl Edward Rasmussen and Christopher K. I. Williams. Gaussian Processes for Machine Learning. MIT Press 2005.

Sample Course Level Assessment Questions

Course Outcome1 (CO1):

1. Consider a naive Bayes classifier with 3 boolean input variables, $\mathbf{X_1}$, $\mathbf{X_2}$ and $\mathbf{X_3}$, and one boolean output, **Y**. How many parameters must be estimated to train such a naive Bayes classifier? How many parameters would have to be estimated to learn the above classifier if we do not make the naive Bayes conditional independence assumption?
2. Describe the ID3 algorithm. Is the order of attributes identical in all branches of the decision tree?
3. Explain the difference between (batch) gradient descent and stochastic gradient descent. Give an example of when you might prefer one over the other.
4. 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?
5. Suppose you have a three class problem where class label y $\in$ 0, 1, 2 and each training example $\boldsymbol{X}$ has 3 binary attributes $\boldsymbol{X_1, X_2, X_3} \in \mathbf{0, 1}$. How many parameters (probability distribution) do you need to know to classify an example using the Naive Bayes classifier?

Course Outcome 2(CO2):

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 3(CO3):

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.

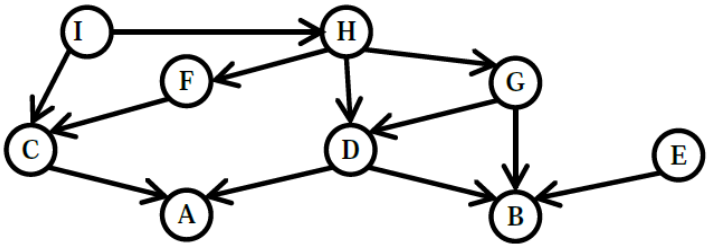
Course Outcome 4(CO4): .

1. A monotone conjunction is a conjunction of the variables such that no variable is negated. Show that the concept class of monotone conjunction is probably approximately correct (PAC)-learnable.
2. Consider a Boolean classification problem with n binary variables and a hypothesis space H, where each hypothesis is a decision tree of depth 2, using only two variables. How many training examples, m suffice to assure that with probability at least 0.99, any consistent learner using H will output a hypothesis with true error at most 0.05
3. Show that the concept class C containing the set of all boolean functions on n variable is not PAC-learnable.

4. What is the Vapnik-Chervonenkis(VC)-dimension of a circle centered at the origin.
5. A hypothesis space that has a high VC dimension is good, bad, or neither? Explain in terms of both (a) richness or expressive power of the hypotheses, and (b) sample complexity.

Course Outcome 5(CO5):

1. Write down the factored conditional probability expression that corresponds to the graphical Bayesian Network shown below.



2. How do we learn the conditional probability tables(CPT) in Bayesian networks if information about some variables is missing? How are these variables called?

Course Outcome 6 (CO6):

1. Derive an algorithm using the inverse transform method to generate a random sample from the exponential distribution.
2. Explain the pros and cons of importance sampling versus rejection sampling.
3. Sketch the core idea of the Monte Carlo method. What is a sample? What is a direct sampling method? Why can't it be used directly to do any inference? What is rejection sampling? What is its major disadvantage?
4. Generative Adversarial Networks(GANs) include a generator and a discriminator. Sketch a basic GAN using those elements, a source of real images, and a source of randomness.
5. The word “adversarial” in the acronym for GANs suggests a two-player game. What are the two players, and what are their respective goals?

Model Question Paper

QP CODE:

Reg No: _____

Name: _____ **PAGES : 5**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SIXTH SEMESTER B.TECH DEGREE EXAMINATION (HONORS), MONTH & YEAR

Course Code: CST 396

Course Name: Advanced Topics in Machine Learning

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

PART A

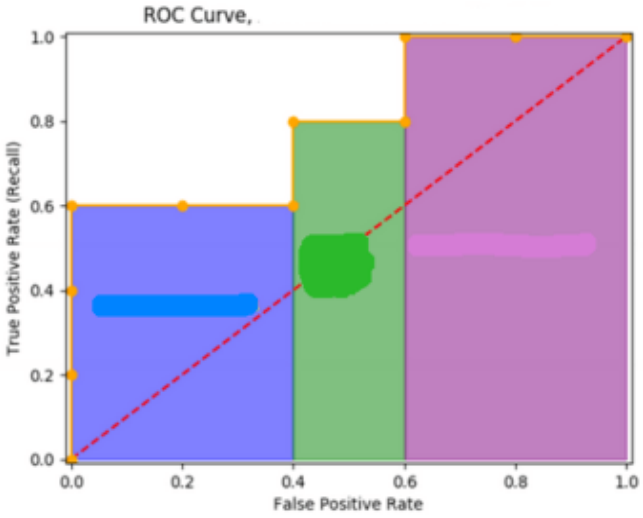
Answer All Questions. Each Question Carries 3 Marks

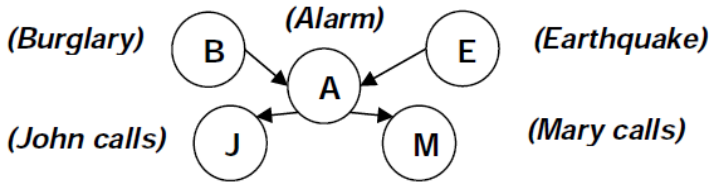
1.	Suppose you have a dataset with m = 1000000 examples and n = 200000 features for each example. You want to use multivariate linear regression to fit the parameters to our data. Should you prefer gradient descent or the normal equation? Justify your answer.	
2.	Define Information gain? How is that different from Gain ratio? Give the advantage of using Gain ratio measure?	
3.	What is cluster analysis? Identify two applications where cluster analysis can be applied to multimedia data?	
4.	Given two objects represented by the tuples (22, 1, 42, 10) and (20, 0, 36, 8): (i) Compute the Euclidean distance between the two objects. (ii) Compute the Manhattan distance between the two objects.	
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.	
6.	How Bias-Variance Tradeoff affects machine learning algorithms?	
7.	For a particular learning task, if the requirement of error parameter ε changes from 0.1 to 0.01. How many more samples will be required for probably approximately correct(PAC) learning?	

8.		Suppose we have a hypothesis set that labels all points inside an interval ***[a, b]*** as class 1. Find its Vapnik-Chervonenkis(VC)- dimension?	
9.		Given a density function ***f(x)***, the rejection sampling is a method that can generate data points from the density function ***f***. List the three steps to generate a random sample from f using rejection sampling.	
10.		How does the variational auto-encoder(VAE) architecture allow it to generate new data points, compared to auto-encoder, which cannot generate new data points?	**(10x3=30)**
		Part B **(Answer any one question from each module. Each question carries 14 Marks)**	
11.	(a)	Consider the hypothesis for the linear regression $h_\theta(x) = \theta_0 + \theta_1 x$, and the cost function $J(\theta_0, \theta_1) = 1/2m \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$ where m is the number of training examples. Given the following set of training examples. x | y 3 | 2 1 | 2 0 | 1 4 | 3 Answer the following questions : 1) Find the value of $h_\theta(2)$ 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.	**(5)**
	(b)	Write a gradient descent algorithm for multivariate regression? Compare the gradient and analytical solution to the multivariate regression?	**(9)**
		OR	
12.	(a)	Consider the collection of training samples (S) in the Figure given below. Drug is the target attribute which describes the Drug suggested for each patient. Find the value of the following . i) Gain(S, Sex) ii) Gain (S, Cholesterol)	**(9)**

		<table><tr><th>Patient ID</th><th>Age</th><th>Sex</th><th>BP</th><th>Cholesterol</th><th>Drug</th></tr><tr><td>p1</td><td>Young</td><td>F</td><td>High</td><td>Normal</td><td>Drug A</td></tr><tr><td>p2</td><td>Young</td><td>F</td><td>High</td><td>High</td><td>Drug A</td></tr><tr><td>p3</td><td>Middle-age</td><td>F</td><td>Hiigh</td><td>Normal</td><td>Drug B</td></tr><tr><td>p4</td><td>Senior</td><td>F</td><td>Normal</td><td>Normal</td><td>Drug B</td></tr><tr><td>p5</td><td>Senior</td><td>M</td><td>Low</td><td>Normal</td><td>Drug B</td></tr><tr><td>p6</td><td>Senior</td><td>M</td><td>Low</td><td>High</td><td>Drug A</td></tr><tr><td>p7</td><td>Middle-age</td><td>M</td><td>Low</td><td>High</td><td>Drug B</td></tr><tr><td>p8</td><td>Young</td><td>F</td><td>Normal</td><td>Normal</td><td>Drug A</td></tr><tr><td>p9</td><td>Young</td><td>M</td><td>Low</td><td>Normal</td><td>Drug B</td></tr><tr><td>p10</td><td>Senior</td><td>M</td><td>Normal</td><td>Normal</td><td>Drug B</td></tr><tr><td>p11</td><td>Young</td><td>M</td><td>Normal</td><td>High</td><td>Drug B</td></tr><tr><td>p12</td><td>Middle-age</td><td>F</td><td>Normal</td><td>High</td><td>Drug B</td></tr><tr><td>p13</td><td>Middle-age</td><td>M</td><td>High</td><td>Normal</td><td>Drug B</td></tr><tr><td>p14</td><td>Senior</td><td>F</td><td>Normal</td><td>High</td><td>Drug A</td></tr></table>	
	(b)	Explain how LASSO regression helps to reduce the overfitting problem?	**(5)**
13.	(a)	Suppose that we have the following data: <table><tr><th>a</th><th>b</th><th>c</th><th>d</th><th>e</th><th>f</th><th>g</th><th>h</th><th>i</th><th>j</th></tr><tr><td>(2,0)</td><td>(1,2)</td><td>(2,2)</td><td>(3,2)</td><td>(2,3)</td><td>(3,3)</td><td>(2,4)</td><td>(3,4)</td><td>(4,4)</td><td>(3,5)</td></tr></table> Identify the cluster by applying the k-means algorithm, with k = 2. Try using initial cluster centers as far apart as possible.	**(9)**
	(b)	Describe EM algorithm for Gaussian mixtures.	**(5)**
		OR	
14.	(a)	Illustrate the strength and weakness of k-means in comparison with the k-medoids algorithm.	**(4)**
	(b)	Suppose that we have the following data . Use single linkage Agglomerative clustering to identify the clusters. Draw the Dendogram. Use Euclidean distance measure	**(10)**

		<table><tr><th></th><th>X</th><th>Y</th></tr><tr><td>P1</td><td>0.4</td><td>0.53</td></tr><tr><td>P2</td><td>0.22</td><td>0.38</td></tr><tr><td>P3</td><td>0.35</td><td>0.32</td></tr><tr><td>P4</td><td>0.26</td><td>0.19</td></tr><tr><td>P5</td><td>0.08</td><td>0.41</td></tr><tr><td>P6</td><td>0.45</td><td>0.30</td></tr></table>	
15.	(a)	Define Precision, Recall, Accuracy and F-measure?	**(4)**
	(b)	What does it mean for a classifier to have a high precision but low recall?	**(3)**
	(c)	Fill in the missing values in the accompanying three class confusion matrix. Given that model accuracy is 72% and classification error for class 2 is 20%. Find also the precision and recall for class1 <table><tr><th colspan="2" rowspan="2"></th><th colspan="3">Predicted</th></tr><tr><th>Class 1</th><th>Class 2</th><th>Class 3</th></tr><tr><td rowspan="3">Actual</td><td>Class 1</td><td>14</td><td>2</td><td>5</td></tr><tr><td>Class 2</td><td>?(X)</td><td>40</td><td>2</td></tr><tr><td>Class 3</td><td>1</td><td>?(Y)</td><td>18</td></tr></table>	**(7)**
		OR	

16.	(a)	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?	**(4)**
	(b)	Given the following ROC Curve? Find the AUC? 	**(3)**
	(c)	How does random forest classifier work? Why is a random forest better than a decision tree?	**(7)**
17.	(a)	Show that the concept class Cn of the conjunction of boolean literals up to n variables is probably approximately correct(PAC)-learnable.	**(8)**
	(b)	Explain the concept of Vapnik-Chervonenkis (VC) dimension using shattering. How the number of training examples required to train the model is related to the VC dimension and what is its relation with training and test errors.	**(6)**
		OR	
18.	(a)	Consider a Boolean classification problem with ***n*** binary variables and a hypothesis space ***H***, where each hypothesis is a decision tree of depth 1. How many training examples, ***m*** suffice to assure that with probability at least 0.99, any consistent learner using ***H*** will output a hypothesis with true error at most 0.05?	**(7)**
	(b)	Prove that VC(H) ≤ log2 |H|, where H is a hypothesis space. (|H| denotes the	**(7)**

		cardinality of the hypothesis space)	
19.	(a)	Shown below is the Bayesian network corresponding to the Burglar Alarm problem, P(J | A) P(M | A) P(A | B, E) P(B) P(E). The probability tables show the probability that variable is True, e.g., P(M) means P(M = t). Find P(J = t ∧ M = f ∧ A = f ∧ B = f ∧ E = t).  P(E): .002 A | P(M): t | .70; f | .01 B | E | P(A): t | t | .95; t | f | .94; f | t | .29; f | f | .001 P(B): .001 A | P(J): t | .90; f | .05	**(7)**
	(b)	Derive an algorithm using the inverse transform method to generate a random sample from the distribution with density $f_X(x) = 3x^2,\ 0 < x < 1$.	**(7)**
		OR	
20.	(a)	Draw the Bayesian Network that corresponds to this conditional probability: ***P(A | B,C,E) P(B | D,E) P(C | F,H) P(D | G) P(E| G,H) P(F | H) P(G) P(H)***	**(6)**
	(b)	What is effective sample size (ESS)? Why is a large ESS necessary but not sufficient for good MCMC mixing?	**(3)**
	(c)	Describe the overall Gibbs sampling algorithm briefly	**(5)**

Teaching Plan

Module 1 : (Supervised Learning)(10 hours)		
1.1	Supervised, semi-supervised, unsupervised learning, reinforcement learning (TB 2: Ch 1)	1 hour
1.2	Least squares linear regression (TB 2: Section 2.6)	1 hour
1.3	Gradient descent, closed form, normal equations (TB 2: Section 5.8)	1 hour
1.4	Regularization techniques (LASSO, RIDGE) (TB 4: Section 7.1)	1 hour
1.5	Polynomial regression (TB 2: Section 2.6)	1 hour
1.6	Logistic Regression (TB 6: Section 3.3)	1 hour
1.7	Decision Tree Learning (ID3) (TB 8: Section 8.2)	1 hour
1.8	Decision Tree Learning (C4.5) (TB 8: Section 8.2)	1 hour
1.9	Naive Bayes Classifier (TB 8: Section 8.3)	1 hour
1.10	Gaussian Discriminant Analysis (GDA) (TB 7: Section 5.2,5.3)	1 hour
Module 2 : (Unsupervised Learning)(8 hours)		
2.1	Similarity measures (TB 8: Section 2.4)	1 hour
2.2	Hierarchical Agglomerative Clustering (TB 3: Chapter 14)	1 hour
2.3	Hierarchical Agglomerative Clustering (TB 3: Chapter 14)	
2.4	K-means partitional clustering (TB 3: Chapter 13)	1 hour
2.5	K-medoids partitional clustering	
2.6	Gaussian mixture models (TB 3: Chapter 13)	1 hour
2.7	Expectation Maximization (EM) algorithm for Gaussian mixture model Lecture-1 (TB 3: Chapter 13)	1 hour
2.8	Expectation Maximization (EM) algorithm for Gaussian mixture model Lecture-2 (TB 3: Chapter 13)	1 hour
Module 3 : (Practical aspects in machine learning) (6 hours)		

3.1	Precision, Recall, Accuracy, F-Measure, ROC, AUC (TB8.5/TB 3: Chapter 22.1)	1 hour
3.2	Generalisation and overfitting, cross-validation (TB 2: Section 2.7,4.8)	1 hour
3.3	Bias-variance tradeoff (TB 2: Chapter 22.3)	1 hour
3.4	Error estimation, parameter and model selection (TB 3: Chapter 8.5)	1 hour
3.5	Bagging, Boosting (TB 8: Chapter 8.6)	1 hour
3.6	Adaboost, Random Forests (TB 8: Chapter 8.6)	1 hour
Module 4 : (Statistical Learning Theory) (TB 5 – Chapter 2, 3.3)(7 hours)		
4.1	Learning in the limit, probably approximately correct (PAC) learning	1 hour
4.2	Quantifying the number of examples needed to PAC learn	1 hour
4.3	Computational complexity of training	1 hour
4.4	Sample complexity for finite hypothesis spaces	1 hour
4.5	PAC results for learning conjunctions	1 hour
4.6	Sample complexity for infinite hypothesis spaces	1 hour
4.7	Vapnik-Chervonenkis(VC) dimension	1 hour
Module 5 : (Advanced Machine Learning Topics) (13 hours)		
5.1	Bayesian belief networks (TB 1 – Chapter 8)	1 hour
5.2	Markov random fields (TB 1 – Chapter 8)	1 hour
5.3	Inference on chains and factor graphs (TB 1 – Chapter 8)	1 hour
5.4	Inference on clique trees (TB 1 – Chapter 8)	1 hour
5.5	Basic sampling algorithms (TB 1 – Chapter 11)	1 hour
5.6	Rejection sampling (TB 1 – Chapter 11)	1 hour
5.7	Importance sampling (TB 1 – Chapter 11)	1 hour
5.8	Markov chain Monte Carlo(MCMC) (TB 1 – Chapter 11)	1 hour
5.9	Gibbs sampling (TB 1 – Chapter 11)	1 hour

5.10	Variational method (TB 1 – Chapter 10)	1 hour
5.11	Auto Encoder (TB 4 – Chapter 14)	1 hour
5.12	Variational AutoEncoder (TB 9 – Chapter 3)	1 hour
5.13	Generative Adversarial Networks (TB 9 – Chapter 4)	1 hour

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
KTU
Estd. 2014

CST 398	THEORY OF COMPUTABILITY AND COMPLEXITY	Category	L	T	P	Credit	Year of Introduction
		VAC	3	1	0	4	2019

Preamble:

This is a theoretical course in computer science to enable the learners to know the fundamentals of computability and complexity theories. It covers the notions of computability/decidability, the process of reduction to prove decidability/undecidability and the classification of problems into class P, class NP and class NP Complete based on the time complexity of solving the problems. This course helps the learner to identify whether a real life problem is decidable/undecidable and also to classify a decidable problem into tractable or intractable, based on the time complexity class it belongs.

Prerequisite: Sound knowledge in Data Structures and Formal Languages and Automata Theory.

Mapping of course outcomes with program outcomes

CO1	Illustrate relative computing powers of Finite State Automata, Push Down Automata, Linear Bounded Automata and Turing Machines.**(Cognitive Knowledge Level: Apply)**
CO2	Prove that a given language is undecidable/not semi-decidable by using the reduction process.**(Cognitive Knowledge Level: Apply)**
CO3	Describe the time complexity of a given problem as a function of the number of steps required by a Turing machine to solve it. **(Cognitive Knowledge Level: Understand)**
CO4	Utilize polynomial time reduction to prove that a given problem is NP Complete. **(Cognitive Knowledge Level: Apply)**

Mapping of course outcomes with program outcomes

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

CO3	✓	✓	✓	✓								✓
CO4	✓	✓	✓	✓								✓

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 (Introduction to Formal Language Theory and Regular Languages)

Finite State Automata, Push Down Automata, Linear Bounded Automata, Turing Machines, Recursive Languages, Recursively Enumerable Languages, Universal Turing Machine, Enumeration Machine, Two Counter Machine.

Module– 2 (Undecidability)

Halting Problem, Language representation of a problem, Reduction - applications, Rice's First and Second Theorem with proof.

Module - 3 (Overview of Complexity Classes)

Measuring time complexity, Asymptotic notations - Big O and small-o, Analysing algorithms, Complexity relationship among models. Complexity classes- Class P, example problems in class P, Class NP, Polynomial time verification, example problems in class NP.

Module- 4 (NP Completeness)

Satisfiability problem, Polynomial time reducibility, Overview of Graphs, NP Complete Problems, Cook-Levin theorem (SAT is NP Complete).

Module- 5 (More NP Complete Problems)

CLIQUE, Vertex Cover and Hamiltonian path with proof of correctness of NP Completeness.

Text Books

1. Dexter C. Kozen, Automata and Computability, Springer (1999)
2. Michael Sipser, Introduction to the Theory of Computation, Second Edition

Reference Books

1. Douglas B. West, Introduction to Graph Theory, Second Edition

Course Level Assessment Questions

Course Outcome1 (CO1):

Identify the class of the following languages in Chomsky Hierarchy:

1. Design a Finite State Automaton for the language $L = \{axb | x \in \{a, b\}^*\}$
2. Design a Push Down Automaton for the language $L = \{a^n b^n | n \geq 0\}$
3. Design a Linear Bounded Automaton for the language $L = \{a^n b^n c^n | n \geq 0\}$
4. Design a Turing Machine for the language $L = \{ww | w \in \{a, b\}^*\}$

Course Outcome 2(CO2):

Without using Rice's Theorem prove that neither the set FIN (refer Text Book 1) nor its complement is recursively enumerable.

Course Outcome 3(CO3):

Show that the language $L = \{a^n b^n | n \geq 0\}$ can be decided by a deterministic Turing Machine in quadratic time.

Course Outcome 4(CO4): .

Using polynomial time reduction, prove that SUBSET-SUM (refer Text Book 2) problem is NP Complete.

Model Question Paper

QP CODE:

Reg No: _____

Name: _____ **PAGES : 4**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

SIXTH SEMESTER B.TECH. DEGREE EXAMINATION(HONORS), MONTH & YEAR

Course Code: CST 398

Course Name: Theory of Computability and Complexity

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

PART A

Answer All Questions. Each Question Carries 3 Marks

1. Design a Deterministic Finite state Automaton (DFA) for the language**:** $L = \{x \in \{0,1\}^* | x \textit{does not contain consecutive zeros}\}$**.**

2. Design a Pushdown Automaton (PDA) for the language $L = \{a^m b^n | m \geq 0 and n > m\}$ (no explanation is required, just list the transitions in the PDA).

3. List any *six* undecidable problems.

4. Illustrate how a problem can be represented as a language.

5. Describe the term time complexity class.

6. Define the term polynomial time verification. Describe its usage.

7. Define the term polynomial time reduction. Describe its usage.

8. Define vertex cover. Illustrate with the help of a graph.

9. Illustrate CLIQUE problem with an example.

10. State Hamiltonian path problem. Show an example.

(10x3=30)

Part B

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

11. (a) Why the family of languages recognized by Turing machines is called Recursively Enumerable? Explain the working of an enumeration machine. **(8)**

(b) Illustrate the functioning of a Universal Turing Machine. **(6)**

OR

12. (a) Illustrate the functioning of a two counter machine. **(4)**

(b) Prove that Turing Machines and Two Counter Machines are equivalent in power. **(10)**

13. (a) Prove using Diagonalisation that halting problem is undecidable. **(8)**

(b) Prove Using reduction that state entry problem of Turing machines is undecidable. **(6)**

OR

14. (a) State and prove Rice's first theorem. **(8)**

(b) Prove Using reduction that whether a Turing Machine accepts empty string (or null string) is undecidable. **(6)**

15. (a) Show that the language $L = \{a^n b^n | n \geq 0\}$ can be decided by a deterministic Turing Machine in $O(n \cdot logn)$ time. **(7)**

(b) Let $t(n)$ be a function, where $n \in \mathbb{N}$ and $t(n) \geq n$. Then, prove that every $t(n)$ time nondeterministic single-tape Turing machine has an equivalent $2^{O(t(n))}$ time single-tape deterministic Turing machine. **(7)**

OR

16. (a) Prove that every context free language is a member of class P. **(8)**

(b) When is a problem said to be in class NP? **(6)**
Prove that Hamiltonian path problem of a directed graph is in class NP.

17. (a) Define Independent set in a graph. Prove that a graph G of n vertices with an independent set of size k contains a vertex cover of size $n - k$. **(8)**

(b) Define the complexity class NP Complete. Explain the significance of an NP Complete problem. **(6)**

OR

18. (a) Define the complement of a graph. Prove that the complement of a graph G of n vertices with a CLIQUE of size k contains an independent set of size k. **(7)**

(b) What is satisfiability problem. Prove that satisfiability problem is in class NP. **(7)**

19. (a) Illustrate Hamiltonian path in a Graph. **(4)**

(b) Prove that Hamiltonian path problem is in the class NP Complete. **(10)**

OR

20. (a) Prove that Vertex Cover problem is in the class NP Complete. **(8)**

(b) Why is it useful to identify that a problem is in the class NP Complete? **(6)**

Teaching Plan

No	Contents	No. of Lecture Hours (45 hrs)
Module-1(Overview of Automata Theory) (10 hours)		
1.1	Finite State Automata	1 hour
1.2	Push Down Automata	1 hour
1.3	Linear Bounded Automata	1 hour
1.4	Turing Machines	1 hour
1.5	Recursive Languages	1 hour
1.6	Recursively Enumerable Languages	1 hour
1.7	Universal Turing Machine	1 hour
1.8	Enumeration Machine	1 hour
1.9	Two Counter Machines	1 hour
1.10	Proof that two Counter Machines and Turing machines are equivalent	1 hour
Module-2 (Undecidability) (10 hours)		
2.1	Halting problem of Turing machine	1 hour
2.2	Proof of undecidability of Halting Problem	1 hour
2.3	Language representation of a problem	1 hour
2.4	Reduction	1 hour
2.5	Applications of reduction - Lecture 1	1 hour

2.6	Applications of reduction - Lecture 2	1 hour
2.7	Rice's First Theorem	1 hour
2.8	Proof of Rice's First Theorem	1 hour
2.9	Rice's Second Theorem	1 hour
2.10	Proof of Rice's Second Theorem	1 hour
Module-3 (Overview of Complexity Classes) (10 hours)		
3.1	Measuring time complexity, Asymptotic notations - Big O and small-o	1 hour
3.2	Analysing algorithms - time complexity class	1 hour
3.3	Complexity relationship among models - Single tape Turing Machine	1 hour
3.4	Multi-tape Turing Machine, Nondeterministic Turing Machine	1 hour
3.5	Class P	1 hour
3.6	Example problems in Class P	1 hour
3.7	Class NP	1 hour
3.8	Polynomial time verification	1 hour
3.9	Example problems in Class NP - Lecture 1	1 hour
3.10	Example problems in Class NP - Lecture 2	1 hour
Module-4 (NP Completeness) (9 hours)		
4.1	Satisfiability problem	1 hour
4.2	Polynomial time reducibility	1 hour
4.3	Overview of Graphs, CLIQUE, INDEPENDENT SET	1 hour
4.4	Vertex Cover	1 hour
4.5	Reducing 3SAT problem to CLIQUE - Lecture 1	1 hour
4.6	Reducing 3SAT problem to CLIQUE - Lecture 2	1 hour
4.7	NP Complete Problems	1 hour
4.8	Cook-Levin theorem, Proof - Lecture 1	1 hour
4.9	Proof - Lecture 2	1 hour
Module-5 (More NP Complete Problems) (6 hours)		

5.1	CLIQUE	1 hour
5.2	Vertex Cover - Lecture 1	1 hour
5.3	Vertex Cover - Lecture 2	1 hour
5.4	Hamiltonian path - Lecture 1	1 hour
5.5	Hamiltonian path - Lecture 2	1 hour
5.6	Hamiltonian path - Lecture 3	1 hour

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
KTU
Estd.
2014