APJ ABDUL KALAM
TECHNOLOGICAL
UNIVERSITY
KTU
Estd.
2014

SEMESTER 1

SEMESTER – I

20MCA101	MATHEMATICAL FOUNDATIONS FOR COMPUTING	CATEGORY	L	T	P	CREDIT
		GENERAL	3	1	0	4

Preamble: This course introduces students to some basic mathematical ideas and tools which are at the core of MCA course. It introduces the concepts of graph theory, set theory and statistics.

Prerequisite: A basic course in set theory and statistics.

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

CO 1	Understand mathematical reasoning in order to read, comprehend and construct mathematical arguments
CO 2	Count or enumerate objects and solve counting problems and analyze algorithms
CO 3	Solve problems in almost every conceivable discipline using graph models
CO 4	Solve the linear system of equations and Calculate the eigen values and eigen vectors of matrices.
CO 5	Apply the principles of correlation and regression in practical problems.

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	3	3	3	3			3					
CO 2	3	3	3	3			3					
CO 3	3	3	3	3			3					
CO 4	3	3	3	3			3					
CO 5	3	3	3	3			3					

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
Estd. 2014

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember(K1)	10	10	10
Understand(K2)	20	20	20
Apply(K3)	20	20	30
Analyse(K4)			
Evaluate(K5)			
Create(K6)			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question caries 3 marks. Part B contains 2 questions from each module of which student should answer any one.Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Define equivalence relation with suitable example. (K1)

2. Write Warshall's algorithm. Use to find the transitive closure of the relation

   {(1,3), (3,2), (2,4), (3,1), (4,1) on (1,2,3,4)} (K2)

3. Let f,g:R□R be defined by f(x)= x+1, g(x)= $2x^2+3$, find fog and gof. Is fog = gof ? (K3)

Course Outcome 2 (CO2)

1. Solve the linear Diophantine equation 24x+138y = 18 (K5)
2. Find the GCD (12378,3054) (K3)
3. Solve a_{n+2}- $4a_{n+1}$+$3a_n$= -200,n □0 given that a_o= 3000, a_1=3300 (K5)

Course Outcome 3(CO3):

1. Define Hamilton cycle and Euler circuit with example. (K1)
2. Show that $K_{3,3}$ is non-planar. Define planar graph. State Kuratowski's theorem. (K4)
3. Prove that a connected graph G is an Euler graph if all vertices of G are of even degree. (K4)

Course Outcome 4 (CO4):

1. Find the rank of the matrix $\begin{bmatrix} 0 & 3 & 4 \\ -3 & 0 & -5 \\ -4 & 5 & 0 \end{bmatrix}$ (K3)
2. Find the Eigen values and Eigen vectors of $\begin{bmatrix} 4 & 2 & -2 \\ 2 & 5 & 0 \\ -2 & 0 & 3 \end{bmatrix}$ (K3)
3. Find out what type of conic sections the quadratic form $Q = 17x_1^2 - 30x_1x_2 + 17x_2^2 = 128$ represents and transform it into principal axes form (K3)

Course Outcome 5 (CO5):

1. State the principle of least squares. (K1)
2. Fit a parabola by the method of least squares, to the following data. (K3)

x:	1	2	3	4	5
y:	5	12	26	60	97

3. Compute the correlation coefficient from the following data. (K3)

x:	77	54	27	52	14	35	90	25	96	60
y:	35	58	60	40	50	40	35	56	34	42

Syllabus

Module 1

Sets, Set Operations, Relations, Classification of relations, Equivalence Relations, Closures of Relations, Matrix Representation of Relations, Partial Ordering, n-ary Relations, Functions.

Module 2

Division Algorithm, GCD, Primes, Euclidean Algorithm, Congruences, Properties of Congruences, Solutions of Linear Congruences.

First Order Linear Recurrence Relation, Second Order Linear Homogeneous Recurrence Relations with Constant coefficients, Non Homogeneous Recurrence Relation.

Module 3

Graphs and Graph Models, Graph Terminology and Special Types of Graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths, Shortest-Path Problems, Planar Graphs, Directed Graph, Multigraph, Connected graph, Euler circuit and trail, Planar and Non-planar Graphs.

Module 4

Linear system of equations, coefficient matrix, augmented matrix, Gauss elimination method and back substitution, elementary row operations, row equivalent systems, Gauss elimination- three possible cases, Row Echelon form and information from it, Linear independence- rank of a matrix. Solution of linear system, fundamental theorem of non- homogeneous linear system (without proof). Homogeneous linear system (theory only), Matrix eigen value problem- determination of eigen values and eigen vectors, Basis of eigen vectors- diagonalization of matrix- Quadratic form- principle axis theorem (without proof).

Module 5

Bivariate data – Scatter Diagram – Interpretation of the nature and degree of relation using scattered diagram - Curve fitting – Principle of least squares – fitting a straight line – fitting a parabola – linear correlation and regression – Karl's Pearson's Coefficient of Correlation – Spearman's rank correlation coefficient (problems based on the formula).

Text Books

1. David M. Burton, "Elementary Number Theory", McGraw-Hill, 7th Edition (2012).

2. Ralph P Grimaldi, "Discrete and Computational Mathematics: An applied introduction", Pearson Education, 5th Edition, (2007).

3. Erwin Kreyszig, "Advanced Engineering Mathematics", 10th ed., Wiley.
4. Gupta S.C and Kapoor V .K, "Fundamentals of Mathematical Statistics", Sultan Chand and Sons 11th edition.

Reference Books

1. C. Liu, “Elements of Discrete Mathematics: A Computer Oriented Approach”, McGraw-Hill, 4th Edition (2012).
2. Jean-Paul Tremblay ,“Discrete Mathematical Structures with applications to Computer science”, ”, McGraw-Hill, 1st Edition (2001).
3. Kenneth H. Rosen , ”Discrete mathematics and its applications”, McGraw-Hill, (7th Edition), (Smartbook available).
4. Marty Lewinter, Jeanine Meyer, “Elementary Number Theory with Programming”, Wiley- Blackwell (2015).
5. David S. Moore and George P. McCabe, “Introduction to practice of statistics”, W.H. Freeman & Company, 5th Edition (2005).
6. Douglas C. Montgomery and George C. Runger, “Applied Statistics and Probability for Engineers”, Wiley India, 5th Edition (2012).
7. Veerarajan T, “Probability and Random Process”, 3rd Edition, Tata McGraw-Hill (2002)
8. G. Jay Kerns, “Introduction to Probability and Statistics Using R”, Chapman & Hall (2010).
9. B.S Grewal. Higher Engineering Mathematics, Khanna Publishers, New Delhi.

Web Resources

1. Probability and statistics EBook
   http://wiki.stat.ucla.edu/socr/index.php/EBook
2. https://www.openintro.org/stat/textbook.php
3. http://www.math.uah.edu/stat/index.html
4. Statistics Online Computational Resource
   http://www.socr.ucla.edu/

Course Contents and Lecture Schedule

Topic	No. of lectures
Module 1	9 hrs.
Sets, Set Operations	2
Relations, Classification of relations, Equivalence Relations	2
Closures of Relations, Matrix Representation of Relations, Partial Ordering, n-ary Relations	3
Functions	2

Module 2	9 hrs.
Division Algorithm, GCD, Primes, Euclidean Algorithm	2
Congruences, Properties of Congruences, Solutions of Linear Congruences	2
First Order Linear Recurrence Relation	1
Second Order Linear homogeneous Recurrence Relations with Constant coefficients	2
Non Homogeneous Recurrence Relation	2
Module 3	8 hrs.
Graphs and Graph Models, Graph Terminology and Special Types of Graphs	1
Representing Graphs and Graph Isomorphism, Connectivity	2
Euler and Hamilton Paths, Shortest-Path Problems, Planar Graphs	2
Directed Graph, Multigraph, Connected graph	1
Euler circuit and trail, Planar and Non-Planar Graphs	2
Module 4	11 hrs.
Linear system of equations, coefficient matrix, augmented matrix, Gauss elimination method and back substitution, elementary row operations, row equivalent systems	2
Gauss elimination- three possible cases, Row Echelon form and information from it	2
Linear independence- rank of a matrix. Solution of linear system, fundamental theorem of non- homogeneous linear system (without proof). Homogeneous linear system (theory only), fundamental theorem of non- homogeneous linear system (without proof). Homogeneous linear system (theory only)	3
Matrix eigen value problem- determination of eigen values and eigen vectors, Basis of eigen vectors	2
diagonalization of matrix, Quadratic form-principle axis theorem (without proof).	2
Module 5	8 hrs.
Bivariate data – Scatter Diagram – Interpretation of the nature and degree of relation using scattered diagram	2
Curve fitting – Principle of least squares – fitting a straight line – fitting a parabola	2
linear correlation and regression – Karl's Pearson's Coefficient of Correlation	2
Spearman's rank correlation coefficient	2

20MCA103	DIGITAL FUNDAMENTALS & COMPUTER ARCHITECTURE	CATEGORY	L	T	P	CREDIT
		GENERAL	3	1	0	4

Preamble:

The primary aim of this course is to understand the fundamentals behind the digital logic design and gain the experience to design digital circuits and systems. Students should also acquire some understanding and appreciation of a computer system's functional components, their characteristics, performance and interactions. They need to understand the computer architecture in order to make best use of the software tools and computer languages they use to create programs.

Prerequisite: NIL

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

CO 1	Apply the basics of digital electronics to design and realize simple combinational logic circuits
CO 2	Apply the digital electronics principles to design sequential logic circuits.
CO 3	Understand the different design features of computer architecture, Five key components of a computer, processor and memory making technologies, addressing modes & instruction formats.
CO 4	Understand Processor logic design conventions and data path, pipelining and hazards, I/O organization, Interrupts and direct memory access
CO 5	Understand and different types of memories - RAM, ROM, Cache memory, virtual memory etc. Apply the different memory design techniques.
CO 6	Understand the concept of single board computers like Arduino, Raspberry Pi etc. and apply the same in practical applications.

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	3	3	2	1	-	-	1	-	-	-	-	-
CO 2	3	3	2	1	-	-	1	-	-	-	-	-
CO 3	1	1	-	1	-	-	1	-	-	-	-	-
CO 4	1	1	-	-	-	-	1	-	-	-	-	-
CO 5	2	2	1	1	-	-	1	-	-	-	-	-
CO 6	1	1	2	-	2	-	2	2	2	-	2	2

Assessment Pattern

Bloom's Category	**Continuous Assessment Tests**		**End Semester Examination**
	1	**2**	
Remember	10	10	20
Understand	20	20	20
Apply	20	20	20
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	**CIE**	**ESE**	**ESE Duration**
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance	: 8 marks
Continuous Assessment Test (2 numbers)	: 20 marks
Assignment/Quiz/Course project	: 12 marks

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 any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Minimize the Boolean Expression f(A,B,C) = Σm(1,3,5,6,7) using K-map.

2. Convert the decimal number 3.257 * 10^4 into single precision floating point binary representation

3. Express -31 in sign magnitude,1's complement and 2's complement notations

Course Outcome 2 (CO2)

1. Explain J-K flipflop with its truth table

2. Design an asynchronous decade counter.

3. Describe the working of a Parallel in Serial Out register.

Course Outcome 3 (CO3):

1. Describe the key components of a computer.

2. Define addressing mode. List 5 addressing modes with examples.

3. Differentiate between fixed length encoding and variable length encoding.

Course Outcome 4 (CO4):

1. Define pipeline, describe how pipeline improves the performance of the machine.

3. List different types of pipeline hazards with examples.

2. Explain how interrupts from multiple devices handled?

Course Outcome 5 (CO5):

1. Illustrate different cache mapping techniques with neat diagrams.

2. Discuss about Read Only Memories

3. Design 2M*32 memory module using 512K *8 static memory chips.

Course Outcome 6 (CO6):

No questions for university examination, for internal assessments practical assignment for configuring a PC / arduino or raspberry and programming assignments using HDL like Verilog or VHDL can be given.

Model Question paper

Reg No.:________________ Name:_________________________________

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

FIRST SEMESTER M.C.A.DEGREE EXAMINATION, MODEL QUESTION PAPER

Course Code: 20MCA103

Course Name: DIGITAL FUNDAMENTALS & COMPUTER ARCHITECTURE

Max. Marks: 60 Duration: 3 Hours

PART A

		Answer all questions, each carries 3 marks.	Marks
1		Represent +45,-45 in 1’s complement and 2’s complement form.	(3)
2		Implement a full adder using 8:1 MUX	(3)
3		How could you convert RS flip flop to D flip flop?	(3)
4		What is meant by modulus of a counter? Realize a mod-8 synchronous counter.	(3)
5		Suppose we have two implementations of the same instruction set architecture. Computer A has a clock cycle time of 250 ps and a CPI of 2.0 for some program, and computer B has a clock cycle time of 500 ps and a CPI of 1.2 for the same program. Which computer is faster for this program and by how much?	(3)
6		Describe about little endian and big endian byte ordering.	(3)
7		Explain 4 stage pipelining with a diagram.	(3)
8		Differentiate between memory mapped I/O and Isolated I/O	(3)
9		What is static RAM ?	(3)
10		Define temporal locality and spatial locality.	(3)

PART B

Answer any one question from each module. Each question carries 6 marks.

Module I

11		Explain about single precision floating point representation with an example	(6)
		OR	
12		Minimize the Boolean expression f(A,B,C,D)=Σm(1,5,6,7,9,15)+d(2,3,11,13) using Karnaugh map and realize it using NAND gates.	(6)

Module II			
13		Demonstrate the working of a JK flip flop. How does it eliminate the invalid condition in SR flip flop? List out its applications.	(6)
OR			
14		Design a mod-12 asynchronous counter.	(6)
Module III			
15		Explain the five classic components of a computer with diagram.	(6)
OR			
16		Describe the code sequence of C=A+B in different types of instruction set architecture.	(6)
Module IV			
17		Draw a single datapath representation for memory instructions and R-type instructions	(6)
OR			
18		What is Direct Memory Access? Explain two types of bus arbitration schemes	(6)
Module V			
19		Elaborate the various cache memory mapping techniques with an example for each.	(6)
OR			
20		Explain the internal organization of memory chips and design a 1K*1 memory chip using decoder.	(6)

Syllabus

Module I (11 Hours)

Representation of signed numbers – 1's complement and 2's complement ,Logic gates – AND - OR – NOT - NAND- NOR - XOR , Boolean algebra - Basic laws and theorems , Boolean functions - truth table, Standard forms of Boolean Expressions – Sum of Products and Product of Sums - minimization of Boolean function using Karnaugh map method - Realization using logic gates, Floating point numbers
Combinational Circuits - Half adder - Full Adder- Decoder -Encoder- Multiplexer – Demultiplexer

Module II (10 Hours)

Sequential circuit - Clocking, Flip flops - SR – JK- D -T flip flops, Counters - Synchronous and asynchronous counters - UP/DOWN counters , Registers - Serial in serial out - Serial in parallel out - Parallel in serial out - Parallel in parallel out registers

A practical assignments may be given in configuring a PC / configuring arduino - Implementing simple programs for blinking an LED - Input from an external switch - fading an LED - serial monitor and debugging / installing & configuring Raspberry pi.

Module III (10 Hours)

Computer abstractions and technology - Introduction, Computer architecture -8 Design features, Application program - layers of abstraction, Five key components of a computer, Technologies for building processors and memory, Performance, Instruction set principles – Introduction, Classifying instruction set architectures, Memory addressing, Encoding an instruction set.

Module IV (9 Hours)

The Processor - Introduction, Logic design conventions, Building a datapath, A simple implementation scheme, An overview of pipelining - Pipelined datapath and control - Structural hazards - Data hazards - Control hazards
I/O organization - Accessing I/O devices, interrupts - handling multiple devices, Direct memory access
Programming assignments may be given in any HDL like Verilog or VHDL to create gate level/ Dataflow/Behavioural level models of gates, multiplexers, adders, flip-flops, registers etc. No detailed teaching of HDL is necessary. The students can be given a basic tutorial write up on gate level modelling.

Module V (8 Hours)

The Memory System – basic concepts, semiconductor RAM memories - organization – static and dynamic RAM, Structure of larger memories, semiconductor ROM memories, Speed, Size and cost ,Cache memory – mapping functions – replacement algorithms , Virtual memory – paging and segmentation.

Text Books

1. Floyd, "***Digital Fundamentals***", Pearson Education, 10th Edition (2011).(Module 1 & 2)
2. J. Hennessy and D. Patterson, "***Computer Organization and Design: The Hardware/Software Interface***", 5th Edition. (Module 3 & 4)
3. J. Hennessy and D. Patterson, "***Computer Architecture, A quantitative approach***", 5th Edition. (Module 3)
4. Hamacher, Vranesic & Zaky, "***Computer Organization***" (5th Ed), McGraw Hill. (Module 4 & 5)

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
Estd. 2014

References

1. William Stallings, "***Computer Organization and Architecture: Designing for Performance***", Pearson, 9/e, 2013.
2. R.P.Jain ,"***Modern Digital Electronics***", McGraw Hill.,Fourth Edition,2009
3. Mano, "***Digital Design : With an Introduction to Verilog HDL***", Pearson Education, 5th Edition (2014)

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
	Module 1	**11**
1	Representation of signed numbers – 1's complement and 2's complement, Logic gates - AND, OR, NOT, NAND, NOR, XOR	2
1.1	Boolean algebra - Basic laws and theorems, Boolean functions - truth table.	2
1.2	Standard forms of Boolean Expressions – Sum of Products and Product of Sums - minimization of Boolean function using Karnaugh map method - Realization using logic gates.	2
1.3	Floating point numbers	1
1.4	Combinational Circuits - Half adder - Full Adder	2
1.5	Decoder – Encoder - Multiplexers – Demultiplexers	2
	Module 2	**10**
2.1	Sequential circuit - Clocking, Flip flops -RS – JK- D -T flip flops	3
2.2	Counters - Synchronous and asynchronous counters - UP/DOWN counters.	3
2.3	Registers - Serial in serial out - Serial in parallel out - Parallel in serial out - Parallel in parallel out registers	2
2.4	Introduction to arduino and raspberry pi	2
	Module 3	**10**
3.1	Computer abstractions and technology - Introduction, Computer architecture	4
3.2	Technologies for building processors and memory, Performance, instruction	4
3.3	Classifying instruction set architectures, Memory addressing, Encoding an	2
	Module 4	**9**
4.1	The Processor - Introduction, Logic design conventions, Building a datapath, A simple implementation scheme.	3
4.2	An Overview of pipelining - Pipelined datapath and control - Structural hazards - Data hazards - Control hazards	3
4.3	I/O organization - Accessing I/O devices, Interrupts - Handling multiple devices- Direct memory access	3

No	Topic	No. of Lectures
	Module 5	**8**
5.1	The memory system – basic concepts, semiconductor RAM memories, organization	2
5.2	Static and dynamic RAM, Structure of larger memories, semiconductor ROM memories, Speed, size and cost	2
5.3	Cache memory – mapping functions – replacement algorithms,	2
5.4	Virtual memory – paging and segmentation.	2

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY KTU Estd. 2014

20MCA105	**ADVANCED DATA STRUCTURES**	**CATEGORY**	**L**	**T**	**P**	**CREDIT**
		GENERAL	**3**	**1**	**0**	**4**

Preamble: A graduate course in Computer Applications should give due exposure to the recent developments. Since Data structures is a central pillar of any program on Computer Science/ Applications, this course is designed to build upon the knowledge acquired at the undergraduate level and familiarise students with a bunch of modern data structures which are quite useful to solve, in the most effective manner, the modern, real life problems.

Prerequisite: Basic Data Structures

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

CO 1	Remember the Basic Data Structures and understand the Set Data Structure and its implementation.
CO 2	Understand Advanced Tree Structures for the design of efficient algorithms
CO 3	Understand Advanced Heap Structures suitable for solving Computational problems involving Optimisation and analysing these data structures using amortised analysis.
CO 4	Understand Advanced Graph algorithms suitable for solving advanced computational problems
CO 5	Understand the basic operation of Blockchaining along with the data structures used in it and the challenges in Blockchain data.

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	3	2	2		1							
CO 2	2	2	3	2	1		1					
CO 3	2	3	3	2	1		1					
CO 4	3	3	2	1	2		1					
CO 5	3	2	2	2	3		1					

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	10	10	10
Understand	20	20	20
Apply	20	20	30
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance	: 8 marks
Continuous Assessment Test (2 numbers)	: 20 marks
Assignment/Quiz/Course project	: 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

(a) Review the basic data structures such as array, linked list, stack, queue etc.
(b) Understand the set data structure and its implementation
(c) Understand the Disjoint set data structure
(d) Learn the basics of Amortised Analysis and its important types

Course Outcome 2 (CO2)

(a) Understand Balanced Binary Search Trees and the idea of Rotations
(b) Understand Red Blak Trees and their operations
(c) Understand B Trees and operations
(d) Obtain a basic awareness of Splay Trees and Suffix Trees.

Course Outcome 3(CO3):

(a) Understand the concepts of Mergeable Heaps and their operations.
(b) Understand the Binomial Heaps and its operations along with their amortised analysis
(c) Understand the Fibonacci Heaps and its operations along with their amortised analysis

Course Outcome 4 (CO4):

(a) Understand Graphs traversal techniques and topological sorting using these
(b) Understand the algorithms for finding the strongly connected components and biconnected components in a graph.
(c) Understand the Prim's and Krusksl's algorithms and their implementation
(d) Understand the Dijikstra's Single Source Shortest path algorithm and implementing it using Advanced Heap Structures.

Course Outcome 5 (CO5):

(a) Understand a basic overview of the Blockchain system architecture.
(b) Understand the Blockchain Data Structures and Data Types.
(c) Understand the problems and challenges in Blockchain data.

Model Question paper

Reg No.:________________		Name:____ _
APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY FIRST SEMESTER M.C.A.DEGREE EXAMINATION, MODEL QUESTION PAPER		
Course Code: 20MCA105		
Course Name: ADVANCED DATA STRUCTURES		
Max. Marks: 60		Duration: 3 Hours
PART A		

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
Estd.
2014

		Answer all questions, each carries 3 marks.	Marks
1		What is meant by Hashing ?	(3)
2		How does Amortised Analysis differ from Average Case Analysis?	(3)
3		What is meant by Balanced Binary Search Tree? Give an example for a balanced binary search tree and an unbalanced one.	(3)
4		What is meant by Suffix Tree?	(3)
5		Give a valid Binomial heap with nodes 3,5,7,10,12,15.	(3)
6		Explain how fibonacci heaps are implemented?	(3)
7		What do you mean by Minimum Costs Spanning Tree?	(3)
8		What is meant by Strongly Connected Components? Illustrate with an example	(3)
9		What is meant by Block Chaining?	(3)
10		What is Contract Data ?	(3)
		PART B	
		Answer any one question from each module. Each question carries 6 marks.	
		Module I	
11		How do you perform Amortised Analysis using Accounting method? Illustrate with Multipop Stack example.	(6)
		OR	
12		Explain any three Hashing functions.	(6)
		Module II	
13		What is meant by Red Black Tree? Explain how insertion is done in a Red Black Tree.	(6)
		OR	
14		Give notes on B-Trees and Splay Trees.	(6)
		Module III	
15		Explain how the Decrease-Key operation is performed on Binomial Heaps. What is the Amortised Cost of this operation?	(6)
		OR	
16		Describe how the Delete-Key operation is performed in a Fibonacci heap? Illustrate with an example.	(6)
		Module IV	
17		Explain the Breadth First Search algorithm with a suitable example.	(6)
		OR	
18		Explain the Prim's algorithm with an example.	(6)

Module V			
19		Explain the Blockchain architecture in detail.	(6)
OR			
20		Explain the problems to be solved in Blockchain Data Analysis.	(6)
****			

Syllabus

Module 1 [12 hrs]

Review of basic data structures- Array, linked list and its variants, Stack ,Queue and Trees
Set Data Structure:- Representation of sets, Set implementation using bit string.
Hashing :- SImple hash functions, Collision and Collision Resolution techniques
Amortised Analysis - Aggregate, Accounting and Potential Methods (using the examples Multipop Stack and Incrementing Binary Counter only)
Disjoint sets- representations, Union, Find algorithms

Module 2 [10 hrs]

Advanced Tree Structures:- Balanced Binary Search trees, Red-Black trees- Properties of Red Black trees, Rotations, Insertion, Deletion. B-Trees- Basic operations on B-Trees – Insertion and Deletion, Introduction to Splay Trees and Suffix Trees

Module 3 [10 hrs]

Advanced Heap Structures:- Mergeable Heaps and operations on Mergeable Heaps. Binomial Heaps, Binomial Heap operations and Analysis, Fibonacci Heaps, Fibonacci Heap operations and Analysis**.**

Module 4 [14 hrs]

Advanced Graph Structures : Representation of graphs, Depth First and Breadth First Traversals, Topological Sorting, Strongly connected Components and Biconnected Components Minimum Cost Spanning Tree algorithms- Prim's Algorithm, Kruskal' Algorithm,. Shortest Path Finding algorithms – Dijikstra's single source shortest paths algorithm

Module 5[8 hrs]

Blockchain Data Structure:- Blockchain Architecture, Blockchain Data Structures and Data types, Contract Data, Problems to be solved in Blockchain data analysis

Text Books

1. Cormen T.H., Leiserson C.E, Rivest R.L. and Stein C, ***Introduction to Algorithms***, Prentice Hall India, New Delhi, 2004 [Modules 1 to 4]

2. Yang, Xiaojing, Jinshan Liu, and Xiaohe Li. "***Research and Analysis of Blockchain Data." Journal of Physics: Conference Series. Vol. 1237***. No. 2. IOP Publishing, 2019.

Reference Books

1. Kleinberg, Jon, and Eva Tardos. ***Algorithm design***. Pearson Education India, 2006.

2. Aho A.V., Hopcroft J.E., and Ullman J.D., ***Data Structures and Algorithms***, Pearson Education, New Delhi, 1983.

3. Sahni S., ***Data Structures, Algorithms, and Applications in C++,*** Mc Graw Hill, Singapore, 1998.

Course Contents and Lecture Schedule

No	Topic	No. of Lecture Hours
1	**Review of basic data structures**	**10Hrs**
1.1	Array, Stack and Queue	
1.2	Linked list and its variants	
1.3	Representation of sets, Set implementation using bit string.	
1.4	Hashing – SImple hash functions	
1.5	Collision and Collision Resolution techniques	
1.6	Amortised Analysis	
1.7	Aggregate Method (Multipop Stack and Incrementing Binary Counter)	
1.8	Accounting Method (Multipop Stack and Incrementing Binary Counter)	
1.9	Potential Method (Multipop Stack and Incrementing Binary Counter)	
1.10	Disjoint sets- representations	
1.11	Union, Find algorithms	
2	**Advanced Tree Structures**	**10Hrs**
2.1	Balanced Binary Search trees	
2.2	Red-Black trees	
2.3	Properties of Red Black trees	
2.4	Rotations	
2.5	Insertion	

2.6	Deletion	
2.7	B-Trees	
2.8	Insertion and Deletion	
2.9	Splay Trees	
2.10	Suffix Trees	
3	**Advanced Heap Structures**	**8Hrs**
3.1	Mergeable Heaps	
3.2	Operations on Mergeable Heaps	
3.3	Binomial Heaps	
3.4	Binomial Heaps operations and Analysis	
3.5	Fibonacci Heaps	
3.6	Fibonacci Heap operations and Analysis.	
4	**Advanced Graph Structures**	**12Hrs**
4.1	Representation of graphs	
4.2	Depth First and Breadth First Traversals	
4.3	Topological Sorting	
4.4	Strongly connected Components	
4.5	Biconnected Components	
4.6	Minimum Cost Spanning Tree	
4.7	Prim's Algorithm	
4.8	Kruskal’s Algorithm	
4.9	Dijikstra’s single source shortest paths algorithm	
5	**Blockchain Data Structure**	**8Hrs**
5.1	Blockchain Architecture	
5.2	Blockchain Data Structures	
5.3	Blockchain Data types	
5.4	Contract Data	
5.5	Problems to be solved in Blockchain data analysis	

20MCA107	ADVANCED SOFTWARE ENGINEERING	CATEGORY	L	T	P	CREDIT
		GENERAL	3	1	0	4

Preamble:

Most of the programs on Computer Applications do not give due importance to teach Software Engineering in an Industry perspective. But this course, built upon the tools and techniques prevalent in Industry today, is supposed to make students Industry-ready.

Prerequisite: Programming proficiency in at least one of C, C++, Java, Python or PHP programming languages.

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

CO 1	Get a full view of the Software life cycle
CO 2	Gain a deep knowledge of Software Planning, Analysis and Design and Software Engineering Models
CO 3	Have a great comprehension of Coding Practices, Version Control using 'git' and Software Quality
CO 4	Acquire ample grasp of Design Patterns
CO 5	Get deeply familiarised with Software Testing and its automation
CO 6	Start using Agile Methodology
CO 7	Begin to apply CI/CD techniques in Software development

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		2	2					3			1	1
CO 2		3	3					3				
CO 3					3				3	2	2	
CO 4			3		3							
CO 5					3					2	3	
CO 6					2			2	2		2	3
CO 7					3			1		2		

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	10	10	10
Understand	20	20	20
Apply	10	10	20
Analyse			
Evaluate			
Create	10	10	10

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance	: 8 marks
Continuous Assessment Test (2 numbers)	: 20 marks
Assignment/Quiz/Course project	: 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

(a) Understand the software development as an engineering process and its stages.

(b) Understand Software development lifecycle (SDLC).

(c) Understand software engineering models.

(d) Learn how to prepare software requirements specification, approaches and methodologies to prepare requirement specifications document.

Course Outcome 2 (CO2)

(a) Understand writing industry-grade software programs, following style guides and coding standards.

(b) Learn core concepts of software version control system and common operations with Git distributed version control system.

(c) Understanding software quality concepts with respect to software requirement specifications document, what to conform to at various stages of SDLC.

(d) Understand what to ensure at various stage of SDLC to ensure quality of developed software system.

Course Outcome 3(CO3):

(a) Learn Object Oriented Programming concepts comprehensively.

(b) Learn the concept of Design Patterns, category of patterns, and how to select appropriate design patterns.

(c) Understand Unit testing concepts and xUnit architecture.

(d) Learn Unit testing frameworks and writing unit testing for Java and one of PHP or Python.

(e) Understand the concepts Continuous Integration and Continuous Delivery (CICD).

Course Outcome 4 (CO4):

(a) Knowledge of Git distributed version control system to use in a product environment.

(b) Knowledge of OOP paradigm and software Design Patterns to design the software system.

(c) Knowledge of unit testing frameworks such as Junit, uniitest, phpdbg for wiring units tests in a software production environment.

(d) Knowledge of software testing CI/CD practices.

Course Outcome 5 (CO5):

(a) Understand software testing concepts and principles.

(b) Learn common approaches to ensure software quality through testing.

(c) In-depth understanding of various types of testing methodologies.

(d) Learn about testing automation and understand commonly used test automation types.

(e) Learn to use Robot framework.

Course Outcome 6 (CO5):

(a) Understand the concepts of Agile methodology.

(b) Learn to use Scrum framework for implementing Agile methodology for executing a software development process.

(c) Learn to monitor a software development project using a Scrum tool.

Course Outcome 7 (CO5):

(a) Understand the concepts of Software Configuration Management.

(b) Learn about build and deployment environments.

(c) Understand the concepts of Continuous Integration and essential practices.

(d) Understand the concepts of deployment automation and learn to use Ansible.

Model Question paper

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
First Semester MCA Degree Examination (R&S)

Course Code: 20MCA107
Course Name: ADVANCED SOFTWARE ENGINEERING

Total Marks: 60 Duration: 3 Hours

PART A

Answer all questions, each carries3 marks. Marks

1. Why is Software Engineering important? (3)
2. What are the desired requirements of a good software engineering model? (3)
3. What is the purpose of a version control system? (3)
4. Explain the different ways to fix commits in Git (3)
5. What is anti-pattern? (3)
6. What is an abstract test? (3)
7. Distinguish between black box testing and white box testing. (3)
8. Draw a model Sprint Backlog for the login module of a simple web portal (3)

9. Write a short note on release candidate (3)
10. Differentiate continuous delivery and continuous deployment (3)

PART B

Required to answer one question from each module in full.
Each module carries 6 marks for either of the questions.

Module I

11. Prepare a basic Software Requirements Specification for Savings Bank accounts. (6)

OR

12. How is Use Case different from User Stories? Enlist the advantage of each. (6)

Module II

13. How do you create, switch and view branches in Git? explain how to merge commits between branches. (6)

OR

14. You have cloned a repository which was then modified by another developer. You make changes locally and try to execute push. What are the possible outputs? How will you solve the problems, if any? (6)

Module III

15. Explain the important design patterns. (6)

OR

16. When are assertions and expected error tests used in Unit tests? (6)

Module IV

17. Write down the scrum. (6)

OR

18. Differentiate Black box testing and White box testing. Give appropriate example for each for “only black box testing is possible” and “necessary to do white box testing” scenarios. (6)

Module V

19. Explain the strategy for implementing Continuous integration. (6)

OR

20. What is a deployment pipeline? Explain the anatomy of a deployment pipeline with a (6) neat diagram. Comment on the various stages of a deployment pipeline.

Syllabus

Module 1 [8 hrs]

Introduction to Software Engineering: What is Software Engineering, Characteristics of Software.

Life cycle of a software system: software design, development, testing, deployment, Maintenance.

Project planning phase: project objectives, scope of the software system, empirical estimation models, COCOMO, staffing and personnel planning.

Software Engineering models: Predictive software engineering models, model approaches, prerequisites, predictive and adaptive waterfall, waterfall with feedback (Sashimi), incremental waterfall, V model; Prototyping and prototyping models.

Software requirements specification, Eliciting Software requirements, Requirement specifications, Software requirements engineering concepts, Requirements modelling, Requirements documentation. Use cases and User stories.

Module 2 [10 hrs]

Programming Style Guides and Coding Standards; Literate programming and Software documentation; Documentation generators, Javadoc, phpDocumentor.

Version control systems basic concepts; Concept of Distributed version control system and Git; Setting up Git; Core operations in Git version control system using command line interface (CLI): Clone a repository; View history; Modifying files; Branching; Push changes, Clone operation, add, commit, log, diff commands, conflict resolution. Pushing changes to the master; Using Git in IDEs and UI based tools.

Software Quality: Understanding and ensuring requirements specification quality, design quality, quality in software development, conformance quality.

Module 3 [10 hrs]

OOP Concepts; Design Patterns: Basic concepts of Design patterns, How to select a design pattern, Creational patterns, Structural patterns, Behavioural patterns. Concept of Anti-patterns.

Unit testing and Unit Testing frameworks, The xUnit Architecture, Writing Unit Tests using at least one of Junit (for Java), unittest (for Python) or phpdbg (PHP). Writing tests with Assertions, defining and using Custom Assertions, single condition tests, testing for expected errors, Abstract test.

Module 4 [10 hrs]

Concepts of Agile Development methodology; Scrum Framework.

Software testing principles, Program inspections, Program walkthroughs, Program reviews; Blackbox testing: Equivalence class testing, Boundary value testing, Decision table testing, Pairwise testing, State transition testing, Use-case testing; White box testing: control flow testing, Data flow testing.

Testing automation: Defect life cycle; Regression testing, Testing automation; Testing non-functional requirements.

Module 5[10 hrs]

Software Configuration Management: Using version control, Managing dependencies, Managing software configuration, Managing build and deployment environments.

Continuous Integration: Prerequisites for continuous integration, Essential practices.

Continuous Delivery: Principles of Software delivery, Introduction and concepts.

Build and deployment automation, Learn to use Ansible for configuration management.

Test automation (as part of continuous integration), Learn to set up test automation cases using Robot Framework.

Notes

1. At the end of Module 1, conduct the following class work with appropriate evaluation points: Prepare Software Specification Document for a moderately complex process flow system (*e.g.* Broadband fault booking and resolution system covering technical, operational and commercial aspects, covering organizational and subscriber use cases).

2. At the end of Module 2, clone an open source project using Git and perform all based operations.

Reference Books

1. Philip A. Laplante, *What Every Engineer Should Know about Software Engineering*, CRC Press [Module 1]
2. Murali Chemuturi, *Mastering Software Quality Assurance: Best Practices, Tools and Technique for Software Developers*, J Ross Publishing [Module 2]
3. Ben Straub, Scott Chacon, *Pro Git*, 2nd Edition, Apress [Module 2]
4. Erich Gamma et. al., *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley [Moule 3]
5. Vaskaran Sarcar, *Java Design Patterns: A Hands-On Experience with Real-World Examples*, Apress [Module 3]
6. Alistair Cockburn and Robert Cecil Martin, *Agile Software Development: The Cooperative Game (2nd edition)*, Addition Wesley [Module 4]

7. Ken Schwaber , *Agile Software Development with Scrum*, Pearson [Module 4]
8. Lisa Crispin, Agile Testing: *A Practical Guide for Testers and Agile Teams*, Adison Wesley
9. Paul Hamill, *Unit Test Frameworks*, O'Reilly Media [Module 4]
10. Glenford J. Myers, et. al., *The Art of Software Testing*, Wiley [Module 4, 5]
11. Lee Copeland, *A Practitioner's Guide to Software Test Design*, Artech House Publishers [Module 4, 5]
12. Jez Humble and David Farley, *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation*, Pearson Education [Module 5]

Web-based Resources

1. *Git Handbook* https://guides.github.com/introduction/git-handbook/ Retrieved 8 July 2020 [Module 2]
2. *Git User Manual* https://mirrors.edge.kernel.org/pub/software/scm/git/docs/user-manual.html Retrieved 8 July 2020 [Module 2]
3. *Introduction to Software Engineering/Quality* https://en.wikibooks.org/wiki/Introduction_to_Software_Engineering/Quality Retrieved 8 July 2020 [Module 2]
4. *Understanding software design patterns* https://opensource.com/article/19/7/understanding-software-design-patterns Retrieved 8 July 2020 [Module 3]
5. *The Scrum Guide* https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf Retrieved 8 July 2020 [Module 4]
6. *unittest — Unit testing framework* https://docs.python.org/3/library/unittest.html Retrieved 8 July 2020 [Module 4]
7. What is CI/CD? https://www.redhat.com/en/topics/devops/what-is-ci-cd Retrieved 8 July 2020 [Module 5]

Course Contents and Lecture Schedule

No	Topic	No. of Lecture Hours
1	**Software Engineering**	
1.1	What is Software Engineering, Characteristics of Software Engineering	1
1.2	Life cycle of a software system	1
1.3	Project planning	1
1.4	Software Engineering Models	2
1.5	Software Requirements Specification	3
2	**Industry Best Practices**	
2.1	Programming style guides and coding standards	1
2.2	Software version control systems, basic concepts	1
2.3	Git distributed version control system, introduction	2
2.4	Common operations in Git	4

Estd. 2014

No	Topic	No. of Lecture Hours
2.5	Software quality, achieving	2
3	**System Design Methodologies**	
3.1	Object Oriented Programming	1
3.2	Software Design Patterns	4
3.3	Unit Testing concepts and xUnit architecture	1
3.4	Unit testing frameworks: Junit, unittest, phpdbg	2
3.5	Writing unit test code	2
4	**Agile Development Methodology**	
4.1	Agile Development methodology, introduction	2
4.2	Scrum framework	5
4.3	Automated testing	3
5	**Continuous Integration and Continuous Development (CI/CD)**	
5.1	Configuration Management	2
5.2	Continuous Integration, concepts and practices	2
5.3	Continuous Delivery, concepts and practices	2
5.4	Build and deployment automation	2
5.5	Test automation for CI/CD	2

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY KTU Estd. 2014

20MCA131	PROGRAMMING LAB	CATEGORY	L	T	P	CREDIT
		PRACTICAL	0	1	3	2

Preamble: This course introduces a basic step towards program writing and develops the logical ability and problem-solving skill using Python Programming Language. Students are able to do testing and debugging of code written in Python.

Prerequisite: None

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

CO 1	Understands basics of Python Programming language including input/output functions, operators, basic and collection data types
CO 2	Implement decision making, looping constructs and functions
CO 3	Design modules and packages - built in and user defined packages
CO 4	Implement object-oriented programming and exception handling.
CO 5	Create files and form regular expressions for effective search operations on strings and files.

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	2	2	2	1	2							
CO 2	3	3	3	2	2							
CO 3	3	3	3	3	3						1	
CO 4	3	3	3	3	3						1	
CO 5	3	3	3	3	3						1	

Assessment Pattern

Bloom's Category	**Continuous Assessment Tests**		**End Semester Examination**
	1	**2**	
Remember (K1)			
Understand (K2)			
Apply (K3)	10	10	10
Analyse (K4)	10	10	10
Evaluate (K5)	10	10	10
Create(K6)	20	20	20

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY KTU Estd. 2014

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 hours

Continuous Internal Evaluation Pattern:

Maximum Marks: 50	
Attendance	15%
Maintenance of daily lab record and GitHub management	20%
Regular class viva	15%
Timely completion of day to day tasks	20%
Tests/Evaluation	30%

End Semester Examination Pattern:

Maximum Marks: 50			
Verification of Daily program record and Git Repository			5 marks
Viva			10 marks
Problem solving (Based on difficulty level, one or more questions may be given)	Flowchart / Algorithm / Structured description of problem to explain how the problem can be solved / Interface Design	15%	35 marks
	Program correctness	50%	
	Code efficiency	15%	
	Formatted output and Pushing to remote Git repository	20%	
Total Marks			50 marks

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Familiarizing Text Editor, IDE, Code Analysis Tools etc // Use any IDE like PyCharm, PyDev…
2. Display future leap years from current year to a final year entered by user.
3. List comprehensions:
   (a) Generate positive list of numbers from a given list of integers
   (b) Square of N numbers
   (c) Form a list of vowels selected from a given word
   (d) List ordinal value of each element of a word (Hint: use ord() to get ordinal values)
4. Count the occurrences of each word in a line of text.
5. Prompt the user for a list of integers. For all values greater than 100, store ‘over’ instead.
6. Store a list of first names. Count the occurrences of ‘a’ within the list
7. Enter 2 lists of integers. Check (a) Whether list are of same length (b) whether list sums to same value (c) whether any value occur in both
8. Get a string from an input string where all occurrences of first character replaced with ‘$’, except first character.
   [eg: onion -> oni$n]
9. Create a string from given string where first and last characters exchanged. [eg: python -> nythop]
10. Accept the radius from user and find area of circle.
11. Find biggest of 3 numbers entered.
12. Accept a file name from user and print extension of that.
13. Create a list of colors from comma-separated color names entered by user. Display first and last colors.
14. Accept an integer n and compute n+nn+nnn.
15. Print out all colors from color-list1 not contained in color-list2.
16. Create a single string separated with space from two strings by swapping the character at position 1.
17. Sort dictionary in ascending and descending order.

18. Merge two dictionaries.
19. Find gcd of 2 numbers.
20. From a list of integers, create a list removing even numbers.

Course Outcome 2 (CO2)

1. Program to find the factorial of a number
2. Generate Fibonacci series of N terms
3. Find the sum of all items in a list
4. Generate a list of four digit numbers in a given range with all their digits even and the number is a perfect square.
5. Display the given pyramid with step number accepted from user.

```
Eg: N=4
1
2  4
3  6  9
4  8 12  16
```

6. Count the number of characters (character frequency) in a string.
7. Add ‘ing’ at the end of a given string. If it already ends with ‘ing’, then add ‘ly’
8. Accept a list of words and return length of longest word.
9. Construct following pattern using nested loop

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

10. Generate all factors of a number.

11. Write lambda functions to find area of square, rectangle and triangle.

Course Outcome 3(CO3):

1. Work with built-in packages
2. Create a package graphics with modules rectangle, circle and sub-package 3D-graphics with modules cuboid and sphere. Include methods to find area and perimeter of respective figures in each module. Write programs that finds area and perimeter of figures by different importing statements. (Include selective import of modules and import * statements)

Course Outcome 4 (CO4):

1. Create Rectangle class with attributes length and breadth and methods to find area and perimeter. Compare two Rectangle objects by their area.
2. Create a Bank account with members account number, name, type of account and balance. Write constructor and methods to deposit at the bank and withdraw an amount from the bank.
3. Create a class Rectangle with private attributes length and width. Overload '<' operator to compare the area of 2 rectangles.
4. Create a class Time with private attributes hour, minute and second. Overload '+' operator to find sum of 2 time.

5. Create a class Publisher (name). Derive class Book from Publisher with attributes title and author. Derive class Python from Book with attributes price and no_of_pages. Write a program that displays information about a Python book. Use base class constructor invocation and method overriding.

Course Outcome 5 (CO5):

1. Write a Python program to read a file line by line and store it into a list.
2. Python program to copy odd lines of one file to other
3. Write a Python program to read each row from a given csv file and print a list of strings.
4. Write a Python program to read specific columns of a given CSV file and print the content of the columns.
5. Write a Python program to write a Python dictionary to a csv file. After writing the CSV file read the CSV file and display the content.

Syllabus:

Input, Output and Import Functions, Operators, Data Types, Decision Making & Loops, Functions, Modules and Packages, File Handling, Object Handling, Exception Handling, Regular Expressions

Reference Books

1. Wesley J. Chun, “***Core Python Applications Programming***”, 3rd Edition , Pearson Education, 2016
2. Charles Dierbach, “***Introduction to Computer Science using Python***”, Wiley, 2015
3. Jeeva Jose, “***Taming Python by Programming***”, Khanna Publishers, New Delhi, 2018
4. Downey, A. et al., "***How to think like a Computer Scientist: Learning with Python***", John Wiley, 2015

Web References

1. https://archive.org/details/MIT6.00SCS11
2. https://www.coursera.org/course/pythonlearn
3. http://www.learnerstv.com/Free-Computer-Science-Video-lectures-ltv163-Page1.htm
4. https://www.coursera.org/learn/python-databases

Course Contents and Lab Schedule

Topic	**No. of hours**
1. Input, Output and Import Functions	3
2. Operators	5
3. Data Types	6
4. Decision Making & Loops	6
5. Functions	5
6. Modules and Packages	6
7. File Handling	5
8. Object Handling	5
9. Exception Handling	2
10. Regular Expressions	4

20MCA133	WEB PROGRAMMING LAB	CATEGORY	L	T	P	CREDIT
		PRACTICAL	0	1	3	2

Preamble: With a dynamic learn-by-doing focus, this laboratory course encourages the students to explore the designing of web application by implementing the relevant and recent techniques. This course challenges the students to exercise their creativity in both programming and designing.

Prerequisite: Basic understanding of computer programming, Internet and Database etc. is very helpful.

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

CO 1	Explore markup languages features and create interactive web pages using them.
CO 2	Learn and design client-side validation using scripting languages.
CO 3	Design front end web page and connect to the back-end databases.
CO 4	Do Client-side & Server-side scripting
CO 5	Develop Web Applications

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	3	3	3	2	2		3	3				
CO 2	3	3	3	2	2		3	3	1			
CO 3	3	3	3	2	2		3	3				
CO 4	3	3	3	2	2		3	3				2
CO 5	3	3	3	3	3		3	3			2	2

Assessment Pattern

Bloom's Category	**Continuous Assessment Tests**		**End Semester Examination**
	1	**2**	
Remember(K1)			
Understand(K2)			
Apply(K3)	10	10	10
Analyse(K4)	10	10	10

Evaluate(K5)	10	10	10
Create(K6)	20	20	20

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 hours

Continuous Internal Evaluation Pattern:

Maximum Marks: 50	
Attendance	15%
Maintenance of daily lab record and GitHub management	20%
Regular class viva	15%
Timely completion of day to day tasks	20%
Tests/Evaluation	30%

End Semester Examination Pattern:

<table>
<tr><th colspan="4">Maximum Marks: 50</th></tr>
<tr><td colspan="3">Verification of Daily program record and Git Repository</td><td>5 marks</td></tr>
<tr><td colspan="3">Viva</td><td>10 marks</td></tr>
<tr><td rowspan="4">Problem solving
(Based on difficulty level, one or more questions may be given)</td><td>Flowchart / Algorithm / Structured description of problem to explain how the problem can be solved / Interface Design</td><td>15%</td><td rowspan="4">35 marks</td></tr>
<tr><td>Program correctness</td><td>50%</td></tr>
<tr><td>Code efficiency</td><td>15%</td></tr>
<tr><td>Formatted output and Pushing to remote Git repository</td><td>20%</td></tr>
<tr><td colspan="3">Total Marks</td><td>50 marks</td></tr>
</table>

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Model a simple HTML file to demonstrate the use of different tags. (K3)
2. Create a HTML file to link to different HTML page which contains images, tables, and also link within a page. (K6)
3. Create a HTML page with different types of frames such as floating frame, navigation frame & mixed frame. (K6)
4. Analyze CSS by applying the different styles using inline, external & internal style sheets in a HTML file. (K4)
5. Demonstrate a registration form using HTML. (K3)

Course Outcome 2 (CO2)

1. Create a HTML page to explain the use of various predefined functions in a string and math object in java script. (K6)
2. Generate the calendar using JavaScript code by getting the year from the user. (K6)
3. Create a HTML registration form and to validate the form using JavaScript code. (K6)
4. Evaluating JavaScript Event Handling for every click of a button to change the background color of a HTML page. (K5)
5. Create a HTML page to display a new image and text when the mouse comes over the existing content in the page using JavaScript Event Handling. (K6)
6. Create a HTML page to show online exam using JavaScript. (K6)

Course Outcome 3(CO3):

1. Develop a PHP program to connect to a database and retrieve data from a table and show the details in a neat format. (K6)

Course Outcome 4 (CO4):

1. Outline a registration form using PHP and do necessary validations. (K4)
2. Compose Electricity bill from user input based on a given tariff using PHP. (K6)

3. Build a PHP code to store name of students in an array and display it using print_r function. Sort and Display the same using asort & arsort functions. (K6)
4. Build a PHP code to store name of Indian Cricket players in an array and display the same in HTML table. (K6)

Course Outcome 5 (CO5):

1. Develop Web applications using HTML and PHP and deploy. (K6)
2. Using PHP and MySQL, develop a program to accept book information viz. Accession number, title, authors, edition and publisher from a web page and store the information in a database and to search for a book with the title specified by the user and to display the search results with proper headings. (K6)
3. Develop a web application for Airline Reservation System using any PHP framework (Laravel, CodeIgniter, Symfony, CakePHP etc.). (K6)
4. Test the application on an Application Server. (K5)

Syllabus

Introduction To Web: Client/Server concepts, Components of Web Application, Types of Web Content, Overview of HTTP - HTTP request – response, Generation of dynamic web pages, Application Servers, Web Security.

Markup Language (HTML): Formatting and Fonts, Commenting Code, Anchors, Backgrounds, Images, Hyperlinks, Lists, Tables, Frames, HTML Forms.

Cascading Style Sheet (CSS): The need for CSS, Basic syntax and structure, Inline Styles, Embedding Style Sheets, Linking External Style Sheets, Backgrounds, Manipulating text, Margins and Padding, Positioning using CSS.

Client Side Scripting using JavaScript: Core features, Data types and Variables, Operators -Expressions and Statements, Functions, Objects, Array, String - Date and Math related Objects, Document Object Model, Event Handling, Form handling and validations.

An overview of Relational Database Design: Tables, Attributes, Tuples, Primary keys, Foreign keys, Indexes, DDL Commands – CREATE, ALTER, DROP and TRUNCATE; DML Commands – SELECT, INSERT, UPDATE and DELETE.

Server Side Scripting using PHP: Setting up the environment (Example - XAMP server), PHP Programming basics - Print/echo, Variables and constants, Strings and Arrays, Operators, Control structures and looping structures, Functions, Reading Data in Web Pages, Embedding PHP within HTML, Establishing connectivity with database, Debugging with phpdbg.

Web Application development in any PHP framework (Laravel, CodeIgniter, Symfony, CakePHP etc.): Naming convention, MVC model, Connectivity with Database, Database interaction.

Debugging web apps: Browser debugging tools (Any browser web developer tools) - View and change the DOM and CSS, Console, Debug JavaScript, View and debug network activity, Performance tools etc.

Reference Books

1. David Flanagan, "***JavaScript: The Definitive Guide***", 6th Edition", O'Reilly Media
2. Douglas E Comer, "***The Internet Book: Everything You Need to Know About Computer Networking and How the Internet Works***", 4th Edition, Prentice Hall
3. Harvey Deitel and Abbey Deitel, "***Internet and World Wide Web - How To Program***", 5th Edition,

   Pearson Education
4. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, "***Database System Concepts***", McGraw Hill Education, 6th Edition (2011)
5. Steve Suehring, Tim Converse, and Joyce Park, "***PHP6 and MySQL Bible***", Wiley India Pvt Ltd (2009)
6. Steven Holzner, "***PHP-The Complete Reference***", Tata McGraw Hill, 1st Edition (2007)
7. Thomas A Powell, Fritz Schneider, "***JavaScript: The Complete Reference***", 3rd Edition, Tata McGraw Hill

Web Resources

1. http://php.net/manual/
2. https://pepa.holla.cz/wp-content/uploads/2016/08/JavaScript-The-Definitive-Guide-6th-Edition.pdf
3. http://index-of.es/PHP/PHP6%20and%20MySQL%20Bible.pdf
4. https://www.udemy.com/course/html5-fundamentals-for-beginners/
5. https://www.udemy.com/course/programming-in-javascript/
6. https://www.udemy.com/course/php-mysql-tutorial/

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY KTU Estd 2014

List of Lab Experiments/Exercises

1. Create a simple HTML file to demonstrate the use of different tags.
2. Create a HTML file to link to different HTML page which contains images, tables, and also link within a page.
3. Create a HTML page with different types of frames such as floating frame, navigation frame & mixed frame.
4. Create a HTML file by applying the different styles using inline, external & internal style sheets.
5. Create a registration form using HTML.
6. Create a HTML page to explain the use of various predefined functions in a string and math object in java script.
7. Generate the calendar using JavaScript code by getting the year from the user.
8. Create a HTML registration form and to validate the form using JavaScript code.
9. Create a HTML page to change the background color for every click of a button using JavaScript Event Handling.
10. Create a HTML page to display a new image and text when the mouse comes over the existing content in the page using JavaScript Event Handling.
11. Create a HTML page to show online exam using JavaScript.
12. Develop a registration form using PHP and do necessary validations.
13. Compose Electricity bill from user input based on a given tariff using PHP.
14. Build a PHP code to store name of students in an array and display it using print_r function. Sort and Display the same using asort & arsort functions.
15. Build a PHP code to store name of Indian Cricket players in an array and display the same in HTML table.
16. Develop a PHP program to connect to a database and retrieve data from a table and show the details in a neat format.
17. Develop Web applications using HTML and PHP and deploy.
18. Using PHP and MySQL, develop a program to accept book information viz. Accession number, title, authors, edition and publisher from a web page and store the information in a database and to search for a book with the title specified by the user and to display the search results with proper headings.

19. Develop a web application for Airline Reservation System using any PHP framework (Laravel, CodeIgniter, Symfony, CakePHP etc.).
20. Test the application on an Application Server.

Note: *Students can be given a group micro project, so that they learn to work in a team environment. They can also be trained on project management tools.*

Course Contents and Lecture Schedule

Topic	**No. of lectures**
Client/Server concepts, Components of Web Application, Types of Web Content, Overview of HTTP - HTTP request – response, Generation of dynamic web pages, Application Servers, Web Security.	1Hr.
HTML - Formatting and Fonts, Commenting Code, Anchors,Backgrounds, Images, Hyperlinks.	4 Hrs.
HTML - Lists, Tables, Frames, HTML Forms.	4 Hrs.
The need for CSS, Basic syntax and structure, Inline Styles, Embedding Style Sheets, Linking External Style Sheets, Backgrounds.	4 Hrs.
CSS - Manipulating text, Margins and Padding, Positioning using CSS.	4 Hrs.
JavaScript: Core features, Data types and Variables, Operators - Expressions and Statements.	3 Hrs.
JavaScript: Functions, Objects, Array, String - Date and Math related Objects, Document Object Model, Event Handling.	4 Hrs.
JavaScript: Form handling and validations.	4 Hrs.
An overview of Relational Database Design: Tables, Attributes, Tuples, Primary keys, Foreign keys, Indexes, DDL Commands – CREATE, ALTER, DROP and TRUNCATE.	4 Hrs.
DML Commands – SELECT, INSERT, UPDATE and DELETE.	4 Hrs.
PHP: Setting up the environment (Example - XAMP server), PHP Programming basics - Print/echo, Variables and constants.	4 Hrs.
Strings and Arrays, Operators, Control structures and looping structures.	4 Hrs.
Functions, Reading Data in Web Pages, Embedding PHP within HTML, Establishing connectivity with database.	4 Hrs.
PHP framework: naming convention, MVC model, Connectivity with Database, Database Interaction.	6 Hrs.

20MCA135	DATA STRUCTURES LAB	CATEGORY	L	T	P	CREDIT
		PRACTICAL	0	1	3	2

Preamble: This is the companion course of 20MCA105 Advanced Data Structures and provides the students hands-on experience of the advanced data structures which will boost up the knowledge and confidence of students in applying these techniques while dealing with real life computing problems.

Prerequisite: Basic Data Structures, Knowledge of any programming language, preferably ‘C’.

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

CO 1	Use Debuggers, Profilers and advanced Compiler options.
CO 2	Implement the Set and Disjoint Set Data Structures.
CO 3	Understand the practical aspects of Advanced Tree Structures.
CO 4	Realise Modern Heap Structures for effectively solving advanced Computational problems.
CO 5	Implement Advanced Graph algorithms suitable for solving advanced computational problems.

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	2				3		1					
CO 2	**3**	2	2		1							
CO 3	**2**	2	3	2	1		1					
CO 4	**2**	3	3	2	1		1					
CO 5	**3**	3	2	1	2		1					

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember(K1)			
Understand(K2)			
Apply(K3)	10	10	10
Analyse(K4)	10	10	10
Evaluate(K5)	10	10	10
Create(K6)	20	20	20

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 hours

Continuous Internal Evaluation Pattern:

Maximum Marks: 50	
Attendance	15%
Maintenance of daily lab record and GitHub management	20%
Regular class viva	15%
Timely completion of day to day tasks	20%
Tests/Evaluation	30%

End Semester Examination Pattern:

Maximum Marks: 50			
Verification of Daily program record and Git Repository			5 marks
Viva			10 marks
Problem solving (Based on difficulty level, one or more questions may be given)	Flowchart / Algorithm / Structured description of problem to explain how the problem can be solved / Interface Design	15%	35 marks
	Program correctness	50%	
	Code efficiency	15%	
	Formatted output and Pushing to remote Git repository	20%	
Total Marks			50 marks

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Write a C program ‘sum.c’ to add two numbers. Read the input from Standard Input and write output to Standard output. Compile and generate sum.out which is then debug with gdb.
2. Modify ‘sum.c’ by adding a function for finding the sum of two numbers. Then profile the executable with gprof.

Course Outcome 2 (CO2)

1. Create the Set ADT with Add, Remove, Union, Intersection and Difference operations. Implement using Bit Strings.
2. Implement the Disjoint set ADT with Create, Union and Find operations.
3. Implement Kruskal’s algorithm using Disjoint sets.

Course Outcome 3(CO3):

1. Implement B-Tree and its operations..
2. Implement Red Black Tree and the associated operations.

Course Outcome 4 (CO4):

1. Create the Binomial Heap ADT and implement the basic operations.
2. Use any Mergeable Heap to implement Single source shortest path algorithm.

Course Outcome 5 (CO5):

1. Finding the strongly connected components of a directed graph.
2. Prim's Algorithm for Minimum cost spanning tree.

Syllabus:

Based on the syllabus of 20MCA105 Advanced Data Structures.

Text Books

1. Cormen T.H., Leiserson C.E, Rivest R.L. and Stein C, ***Introduction to Algorithms***, Prentice Hall India, New Delhi, 2004

Reference Books

1. Kleinberg, Jon, and Eva Tardos. ***Algorithm design***. Pearson Education India, 2006.
2. Aho A.V., Hopcroft J.E., and Ullman J.D., ***Data Structures and Algorithms***, Pearson Education, New Delhi, 1983.
3. Sahni S., Data Structures, ***Algorithms, and Applications in C++***, Mc Graw Hill, Singapore, 1998.

Web Reference

1. https://gcc.gnu.org/onlinedocs/gcc/Option-Summary.html
2. https://www.gnu.org/software/gdb/documentation/
3. https://ftp.gnu.org/old-gnu/Manuals/gprof-2.9.1/html_mono/gprof.html

Course Contents and Lecture Schedule

Topic	No. of hours
1. Advanced use of gcc : Important Options -o, -c, -D, -l, -I, -g, -O, -save-temps, -pg	1
2. Familiarisation with gdb : Important Commands - break, run, next, print, display, help	1
3. Using gprof : Compile, Execute and Profile	1
4. Review of Basic Data Structures (Array, List, Stack, Queue,Trees) a. Merge two sorted arrays and store in a third array b. Circular Queue - Add, Delete, Search c. Singly Linked Stack - Push, Pop, Linear Search d. Doubly linked list - Insertion, Deletion, Search e. Binary Search Trees- Insertion, Deletion, Search	8
5. Set Data Structure and set operations (Union, Intersection and Difference) using Bit String.	3
6. Disjoint Sets and the associated operations (create, union, find)	3
Topic	**No. of hours**
7. Binomial Heaps and operations (Create, Insert, Delete, Extract-min, Decrease key)	4
8. B Trees and its operations	4
9. Red Black Trees and its operations	4
10. Graph Traversal techniques (DFS and BFS) and Topological Sorting	4
11. Finding the Strongly connected Components in a directed graph	3
12. Prim’s Algorithm for finding the minimum cost spanning tree	3
13. Kruskal’s algorithm using the Disjoint set data structure	3
14. Single Source shortest path algorithm using any heap structure that supports mergeable heap operations	3

APJ ABDUL KALAM
TECHNOLOGICAL
UNIVERSITY
KTU
Estd.
2014

SEMESTER 2

SEMESTER – 2

20MCA102	**ADVANCED DATABASE MANAGEMENT YSTEMS**	**CATEGORY**	**L**	**T**	**P**	**CREDIT**
		GENERAL	**3**	**1**	**0**	**4**

Preamble: This course provides the basic concepts and terminology related to relational and non-relational database management systems. The concept of advanced DBMS techniques and new generation databases like MongoDB, HBase and Cassandra are also introduced. This course serves as a prerequisite for many advanced courses in Data Science and Machine Learning areas.

Prerequisite: Basic knowledge in Database Management Systems.

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

CO 1	Understand the fundamentals of relational database systems including: data models, database architectures and ER features.
CO 2	Analyze and apply the different normalization techniques.
CO 3	Assess the basic issues of transaction processing and concurrency control.
CO 4	Understand the roles that databases play in organizations and familiarize with basic database storage, file organization, database accessing techniques.
CO 5	Understand the basics of query processing, object-oriented, distributed databases.
CO 6	Analyze non-relational database systems and structures and XML.

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				1			1					
CO 2	3	3	3	2			2	2			2	2
CO 3	1	2	2	2		2					2	2
CO 4					1		1					
CO 5	1			1								
CO 6	1											

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember(K1)	10	10	10
Understand(K2)	20	20	20
Apply(K3)	20	20	30
Analyse(K4)			
Evaluate(K5)			
Create(K6)			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance	: 8 marks
Continuous Assessment Test (2 numbers)	: 20 marks
Assignment/Quiz/Course project	: 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Examine why databases are important. (K3)
2. Describe the basic features of the relational data model and discuss their importance to the end user and the designer. (K2)
3. Analyze the graphic depiction of relationships among the entities and examine how these depictions help in the database design process. (K3 & K4)

Course Outcome 2 (CO2):

1. Evaluate and design good table structures to control data redundancies and anomalies. (K5 & K6)

Course Outcome 3(CO3):

1. Explain the database transaction and its properties. (K2)
2. Describe concurrency control and analyze the role it plays in maintaining the database integrity. (K2 & K4)
3. Assess the common algorithms for concurrency control. (K5)
4. Define deadlock and discuss the strategies for managing deadlocks. (K1 & K2)
5. Examine how database recovery management is used to maintain database integrity. (K3)

Course Outcome 4 (CO4):

1. Discuss the various disk-organization techniques. (K2)
2. Describe the various data structures that allow fast access to data. (K2)
3. Analyze and examine the different indexing techniques. (K3 & K4)

Course Outcome 5 (CO5):

1. Describe the basics of query processing and evaluate the query processing cost. (K2 & K5)
2. Analyze the concept of object oriented databases and distributed databases. (K4)

Course Outcome 6 (CO6):

1. Explain the concept of XML. (K2)
2. Describe the various NoSQL databases. (K2)

Model Question Paper

Reg No.:________________		Name:____

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

MODEL QUESTION PAPER M.C.A.DEGREE EXAMINATION

Course Code: 20MCA102

Course Name: ADVANCED DATABASE MANAGEMENT SYSTEMS

Max. Marks: 60		Duration: 3 Hours

PART A

		Answer all questions, each carries3 marks.	Marks
1		Define weak entity set with an example.	(3)
2		With the help of a diagram explain the different levels of data abstraction?	(3)
3		Differentiate between BCNF and 3NF with an example.	(3)
4		Explain functional dependency with suitable example.	(3)
5		Discuss the ACID properties of transaction.	(3)
6		Define deadlock and discuss the strategies for managing deadlocks.	(3)
7		Diagrammatically represent the basic steps in query processing.	(3)
8		Differentiate static and dynamic hashing.	(3)
9		Illustrate the different types of Distributed Databases.	(3)
10		Define collection and document in MongoDB.	(3)

PART B

Answer any one question from each module. Each question carries 6 marks.

Module I

11		Draw an E-R diagram of a college database with entities student, staff, course, teacher, clerk, department & hostel? Relationship names must be meaningful	(6)

		and there should be an ISA relationship also in diagram.	
		OR	
12		Explain the relational model concept and discuss the different relational model constraints.	(6)
		Module II	
13		Analyse the common anomalies found in databases? How can we eliminate it through normalization?	(6)
		OR	
14		Define Normalization. Explain 1NF, 2NF and 3NF in detail.	(6)
		Module III	
15		Define deadlock. How can we deal with deadlocks?	(6)
		OR	
16		Explain concurrency control with locking methods.	(6)
		Module IV	
17		Explain the various RAID levels with appropriate diagrams.	(6)
		OR	
18		Differentiate between Dense index and Sparse index with example.	(6)
		Module V	
19		Explain HBase and Cassandra.	(6)
		OR	
20		Explain XML and its applications.	(6)
		****	

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY KTU Estd 2014

SYLLABUS

Module I:

Relational Databases:- Introduction - Purpose of Database System – Database System Applications - View of data: Data Abstraction, Instances and Schemas, Data Models – Database Architecture - Database Users and Administrators: Database Users and Interfaces, DBA – Introduction to the Relational Model: Structure of Relational Database, database Schema, Keys, Relational Query language – The Relational Algebra: Fundamental Operations, Formal definition of the relational algebra, additional relational algebra operations – The Entity-Relationship model: Entity Set, Relationship Set, Attributes – Constraints: Mapping cardinalities, Key Constraints, Participation Constraints - E-R Diagrams: Basic structure, Complex attributes, Roles, Non binary relationship sets, Weak Entity Set, Relational Database Design using ER- to Relational Mapping – Extended ER Features: Specialization, Generalization, Attribute inheritance, Constraints on generalization, Aggregation.

Module II:

Database Design:- Database Tables and Normalization – The Need for Normalization – The Normalization Process: Inference Rules for Functional Dependencies (proof not needed) - Minimal set of Functional Dependencies - Conversion to First Normal Form, Conversion to Second Normal Form, Conversion to Third Normal Form - Improving the Design - Surrogate Key Considerations - Higher Level Normal Forms: Boyce/Codd Normal Form, Fourth Normal Form, Join dependencies and Fifth Normal Form – Normalization and Database Design.

Module III:

Transaction Management and Concurrency Control:- Transaction: Evaluating Transaction Results, Transaction Properties, Transaction Management with SQL, The Transaction Log – Concurrency Control: Lost Updates, Uncommitted Data, Inconsistent Retrievals, The Scheduler – Concurrency Control with Locking Methods: Lock Granularity, Lock Types, Two Phase Locking to Ensure Serializability, Deadlocks – Concurrency Control with Timestamping Methods: Wait/Die and Wait/Wound Schemes – Concurrency Control with Optimistic Methods - Database Recovery Management: Transaction Recovery.

Module IV:

Data Storage and Querying:- RAID – File Organization – Organization of Records in Files – Indexing and Hashing: Basic concept, Ordered Indices, B+ tree Index Files: Structure of a B+-Tree (structure only, algorithms not needed) - B tree index files – Static Hashing – Dynamic Hashing – Query Processing: Overview - Selection Operation.

Module V:

System Architecture, Object Oriented Databases, XML and NoSQL:- Distributed Databases: Homogeneous and Heterogeneous Databases, Distributed Data Storage,

Distributed Transactions - Object Based Databases: Overview, Complex Data types, Structured types and inheritance in SQL, Table Inheritance, Array and Multiset types in SQL, Object identity and reference types in SQL - XML: DTD and XML Schema, XML presentation, XML Applications - Next Generation Databases: Distributed Relational Databases - Nonrelational Distributed Databases - MongoDB Sharding and Replication - Hbase - Cassandra - CAP Theorem.

Text Books

1. Abraham Silberschatz, Henry F. Korth, S. Sudarshan,” ***Database System Concepts***”, McGraw Hill Education, 6th Edition, 2011. *(for Module 1 Refer Chapter 1 [1.1 to 1.3, 1.9,1.12], Chapter 2 [2.1-2.3,2.5], Chapter 6 [6.1], Chapter 7 [7.2, 7.3, 7.8(7.81. To 7.8.5)], for Module 4 Refer Chapter 10 [10.3, 10.5, 10.6], Chapter 11 [11.1, 11.2, 11.3(11.3.1), 11.4.5 and module 5 Refer Chapter 19 [19.1,19.2, 19.3 - Distributed Databases], Refer Chapter 22 [22.1 to 22.6 - Object Based Databases]).*
2. Ramez Elmasri, Shamkant B.Navathe, “ ***Fundamentals of Database Systems*** “, Pearson Education, 5th Edition, 2007. *(for Module 1 - Refer Chapter 7 [7.1] - 7.1.1 - Relational Database Design using ER- to Relational Mapping]) and for Module 2 - Refer Chapter 10 [10.2.2 and 10.2.4], Refer Chapter 11 [11.4 - Join dependencies and Fifth Normal Form).*
3. Guy Harrison, “***Next Generation Databases: NoSQL, NewSQL, and Big Data***”, Apress, 1st Edition, 14 December 2015. *Refer Chapters 8 and 3 (for Module 5 - Next Generation Databases and CAP Theorem).*

4. Rob, Peter and Carlos Coronel, "***Database Principles: Fundamentals of Design, Implementation and Management***", 9th Edition, 2011. *(for Module 2, refer chapter 6) and (for module 3, refer chapter 10) and (for Module 5, refer Chapter 14 -XML).*

Reference Books

1. Ashutosh Kumar Dubay, "***Database Management Concepts***", S.K. Kataria & Sons, 1st Edition (2012).
2. Raghu Ramakrishnan and Johannes Gehrke, "***Database Management Systems***", McGraw Hill, 3rd Edition (2014).
3. Thomas M Connolly and Carolyn E Begg, "***Database systems- A Practical Approach to Design, Implementation and Management***", Pearson Education, 4th Edition (2014).

Web Resources

1. Introduction to Databases (nptel) https://nptel.ac.in/courses/106/106/106106220/
2. Database Design (nptel) https://nptel.ac.in/courses/106/106/106106093/
3. Introduction to Database Systems and Design https://nptel.ac.in/courses/106/106/106106095/
4. Fundamentals of Database Systems https://nptel.ac.in/courses/106/104/106104135/#
5. Database Management Essentials (Coursera) https://www.coursera.org/learn/database-management
6. Database Systems Concepts & Design https://www.udacity.com/course/database-systems-concepts-design--ud150

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY KTU Estd 2014

Course Contents and Lecture Schedule

No.	Topic	No. of Lectures
1	***Module I: Relational Databases***	***15 hrs***
1.1	Introduction - Purpose of Database System - Database System Applications	1 hr
1.2	View of data: Data Abstraction, Instances and Schemas, Data Models	1 hr
1.3	Database Architecture	1 hr
1.4	Database Users and Administrators: Database Users and Interfaces, DBA	1 hr
1.5	Introduction to the Relational Model: Structure of Relational Database, database Schema, Keys, Relational Query language	1 hr
1.6	The Relational Algebra: Fundamental Operations, Formal definition of the relational algebra, additional relational algebra operations	2 hr
1.7	The Entity-Relationship model: Entity Set, Relationship Set, Attributes	1 hr
1.8	Constraints: Mapping cardinalities, Key Constraints, Participation Constraints	2 hr
1.9	E-R Diagrams: Basic structure, Complex attributes, Roles, Non binary relationship sets, Weak Entity Set	1 hr
1.10	Relational Database Design using ER- to Relational Mapping	2 hr
1.11	Extended ER Features: Specialization, Generalization, Attribute inheritance, Constraints on generalization, Aggregation.	2 hr
2	***Module II: Database Design***	***9 hrs***
2.1	Database Tables and Normalization - The Need for Normalization	1 hr
2.2	The Normalization Process: Inference Rules for Functional Dependencies (proof not needed) - Minimal set of Functional Dependencies - Conversion to First Normal Form, Conversion to Second Normal Form	2 hr
2.3	Conversion to Third Normal Form	1 hr
2.4	Improving the Design - Surrogate Key Considerations	1 hr

No	Topic	No. of Lectures
2.5	Higher Level Normal Forms: Boyce/Codd Normal Form	1 hr
2.6	Fourth Normal Form	1 hr
2.7	Join dependencies and Fifth Normal Form	1 hr
2.8	Normalization and Database Design	1 hr
3	***Module III: Transaction Management and Concurrency Control***	***9 hrs***
3.1	Transaction: Evaluating Transaction Results, Transaction Properties	1 hr
3.2	Transaction Management with SQL, The Transaction Log	1 hr
3.3	Concurrency Control: Lost Updates, Uncommitted Data, Inconsistent Retrievals, The Scheduler	2 hr
3.4	Concurrency Control with Locking Methods: Lock Granularity	1 hr
3.5	Lock Types, Two Phase Locking to Ensure Serializability	1 hr
3.6	Deadlocks	1 hr
3.7	Concurrency Control with Timestamping Methods: Wait/Die and Wait/Wound Schemes, Concurrency Control with Optimistic Methods, Database Recovery Management: Transaction Recovery	2 hr
4	***Module IV: Data Storage and Querying***	***10 hrs***
4.1	RAID	1 hr
4.2	File Organization	1 hr
4.3	Organization of Records in Files	1 hr
4.4	Indexing and Hashing: Basic concept, Ordered Indices	1 hr
4.5	B+ tree Index Files: Structure of a B+-Tree, B tree Index Files	2 hr
4.6	Static Hashing, Dynamic Hashing	2 hr
4.7	Query Processing: Overview, Selection Operation	2 hr

No	Topic	No. of Lectures
5	***Module V: System Architecture, Object Oriented Databases, XML and NoSQL***	***13 hrs***
5.1	Distributed Databases: Homogeneous and Heterogeneous Databases, Distributed Data Storage, Distributed Transactions	2 hr
5.2	Object Based Databases: Overview, Complex Data types	1 hr
5.3	Structured types and inheritance in SQL	1 hr
5.4	Table Inheritance	1 hr
5.5	Array and Multiset types in SQL	1 hr
5.6	Object identity and reference types in SQL	1 hr
5.7	XML: DTD and XML Schema	1 hr
5.8	XML presentation, XML Applications	1 hr
5.9	Next Generation Databases: Distributed Relational Databases - CAP Theorem	1 hr
5.10	Norelational Databases – MongoDB Sharding and Replication	1 hr
5.11	Hbase	1 hr
5.12	Cassandra	1 hr

20MCA104	**ADVANCED COMPUTER NETWORKS**	**CATEGORY**	**L**	**T**	**P**	**CREDIT**
		GENERAL	**3**	**1**	**0**	**4**

Preamble: This course intends to provide insight into Advanced Computer Networks. A software professional should have an understanding of layered network architecture. Various kinds of network architectures, issues in integrating networks to modern application development are to be addressed. It is also intended to expose the student to modern technologies such as IPV6 and software defined networks. More detailed treatment can be done through seminars, assignments and talks by eminent external experts.

Prerequisite: Basic concepts of computer operating systems.

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

CO 1	Comprehend the terminology and concepts of basic communication model, analyse the protocol layers and design application layer protocols.
CO 2	Understand and analyse the various transport layer protocols.
CO 3	Compare and contrast various routing algorithms in the network layer.
CO 4	Understand and analyse the concepts of link layer and physical layer.
CO 5	Understand how modern cellular and wireless networks work

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	3	3	2		2	2	2		3		2	
CO 2	3	3	2		2	2			3		2	
CO 3	3	3			2	2	2		3		2	
CO 4	3	3				2			3		2	
CO 5	3	3				2			3			

Estd. 2014

Assessment Pattern

Bloom's Category	**Continuous Assessment Tests**		**End Semester Examination**
	1	**2**	
Remember	10	10	10
Understand	20	20	20
Apply	20	20	30
Analyze			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance	: 8 marks
Continuous Assessment Test (2 numbers)	: 20 marks
Assignment/Quiz/Course project	: 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Explain HTTP request-response behavior with a neat diagram.
2. Compare and contrast OSI and TCP/IP network reference models.
3. Explain the importance of layering in data communication.

Course Outcome 2 (CO2)

1. Explain the process of three-way handshaking in TCP.
2. Compare and contrast Multiplexing and De-multiplexing process in transport layer.
3. Explain How TCP is controlling congestion during data transmission.

Course Outcome 3(CO3):

1. Explain how multicast routing is used in routing protocols.
2. Compare and contrast IPV4 and IPV6.
3. Differentiate virtual circuits and datagram networks.

Course Outcome 4 (CO4):

1. Explain how parity is used to achieve error detection in data communication.
2. Illustrate IEEE 802.3 frame structure.
3. Write short notes on routers, switches and bridges.

Course Outcome 5 (CO5):

1. List out and explain the various IEEE 802.11 WLAN Components.
2. Explain the architecture of Bluetooth in personal area networks.
3. Explain any six network attacks and their counter measures.

Model Question paper

Part A

1. Differentiate HTTP persistent and non-persistent communication.
2. List out and explain the functionalities of different DNS records.
3. Compare TCP and UDP at transport layer.
4. Demonstrate how stop-and-wait protocol is used for reliable data transfer.
5. Explain how IPv6 solve the problem of IPv4 exhaustion?
6. Explain how ARP is working in data link layer?
7. A series of 8-bit message blocks to be transmitted across a data link using CRC for error detection. A generator polynomial of $x^3+ x^2+1$.is to be used. Message transmitted as 110010. Explain how CRC check is implemented?
8. Classify various wired media used in short and long distance communication.
9. Explain Network Address Translation (NAT).
10. Explain piconet topology of Bluetooth? [3x10=30 Marks]

Part B

Module 1

11. List and explain ISO/OSI layers and their functions. [6 Marks]

OR

12. Describe various service models in Quality of Service (QOS). [6 Marks]

Module 2

13. Write a short note on:
    a. Stop-and-wait [3 Marks]
    b. Go-back-N [3 Marks]

OR

14. Explain the principles of congestion control with its fairness and efficiency. [6 Marks]

Module 3

15. Define routing? Explain the process of link state routing with OSPF protocol.

OR [6 Marks]

16. What is Virtual circuit? Explain the connection management in Virtual circuit with suitable diagrams.

[6 Marks]

Module 4

17. Write a short note on:

a. Collision based multiple access protocol [3 Marks]
b. Token based multiple access protocol [3 Marks]

OR

18. Explain IEEE 802.3 Ethernet frame format with its access protocol.

[6 Marks]

Module 5

19. What is Bluetooth? Explain the various layers of Bluetooth with a neat diagram.
[6 Marks]

OR

20. Write a short note on:
a. Traffic analysis tools [3 Marks]
b. Troubleshooting [3 Marks]

Syllabus

Module	Contents	Hours
I	Overview of Computer Networks and the Internet. History. Protocols, Review of last mile technologies used for internet access. Packet switching. Basic ideas about delay queuing throughput. Concept of Quality of Service, Protocol layering . OSI model and TCP model Application layer protocols - Client-server architecture Network layer 7 application architecture, Web, HTTP, FTP, SMTP, POP3, and DNS, Peer-to-peer file sharing networks	10

Module	Contents	Hours
II	Transport Layer Protocols: Introduction to transport layer, Multiplexing and de-multiplexing, Principles of Reliable data transfer - Stop-and-wait and Go-back- N design and evaluation, Connection oriented transport TCP, Connectionless transport UDP, Principles of congestion control -efficiency and fairness	10
III	Network Layer Protocols: Virtual circuits and datagrams, Principles of routing, internet protocol Ipv4 CIDR Routing algorithms: Link-state and distance vector routing, Routing on the internet RIP OSPF and BGP, Multicast routing. Introduction to IPV6 and software defined networks, Open flow	10
IV	Link layer and Physical Layer: Introduction to link layer - Error detection (parity, checksum, and CRC), Multiple access protocols (collision and token based), IEEE 802.3 Ethernet, Switching and bridging, Media, Signal strength and interference. Data encoding. Ethernet switches , Routers MAC, ARP, FIB	8
V	**I**EEE 802.11 Wi-Fi, Bluetooth, and cellular networks,Threats and attacks, Network Address Translation , Firewalls, VPNs, Introduction to network management, SNMP, Overview of tools and troubleshooting, Traffic analysis tools and Configuration management.	10

Textbooks:

1. Behrouz A Forouzan, Firouz Mosharraf, "***Computer Networks: A top down Approach***", McGraw Hill Education, 1 st Edition (2011).
2. James F Kurose and Keith W Ross, "***Computer Networking: A Top - Down Approach***", Pearson Education; 6 th Edition (2017).

Reference Books:

1. Kevin R. Fall, W. Richard Stevens, "***TCP/IP Illustrated, Volume 1 -The Protocols***", Pearson Education, 2 nd Edition (2014).
2. Larry Peterson, Bruce Davie, "***Computer Networks, A systems Approach***", Morgan Kaufmann Publishers, 5th Edition (2011).
3. Uyless Black, "***Computer Networks: Protocols, Standards and Interface***", Prentice HallIndia Learning Private Limited, 8 th Edition (2015).
4. William Stallings, "***Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud***", Pearson Education, 1 st Edition (2016)
5. ***The Illustrated Network: How TCP/IP Works in a Modern Network*** 2nd edition Walter Goralski Morgan Kaufmann Publications

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1		
1.1	Overview of Computer Networks and the Internet. History. Protocols , Review of last mile technologies used for internet access. Packet switching.	2
1.2	Basic ideas about delay queuing through put. Concept of Quality of Service Protocol layering . OSI model and TCP model	4
1.3	Application layer protocols - Client-server architecture Network application architecture, Web, HTTP, FTP, SMTPPOP3 and DNS, Peer-to-peer file sharing networks	4
2		
2.1	Transport Layer Protocols: Introduction to transport layer	2
2.2	Multiplexing and demultiplexing, Principles of Reliable data transfer - Stop-and-wait and Go-back- N design and evaluation	3
2.3	Connection oriented transport TCP, Connection less transport UDP	3
2.4	Principles of congestion control -efficiency and fairness	2
3		
3.1	Network Layer Protocols: Virtual circuits and datagrams	2
3.2	Principles of routing, internet protocol Ipv4 NAT , Routing algorithms: Link-state and distance vector routing,	3
3.3	Routing on the internet RIP OSPF and BGP, Multicastrouting.	2
3.4	Introduction to IPV6 and software defined networks	2
4		
4.1	Link layer and Physical Layer: Introduction to link layer - Error detection (parity, checksum, and CRC)	2
4.2	Multiple access protocols (collision and token based), IEEE 802.3	2
4.3	Ethernet, Switching and bridging, Media, Signal strength and interference. Data encoding. Ethernet switches , Routers MAC, ARP, FIB	4

No	Topic	No. of Lectures
5		
5.1	**I**EEE 802.11 Wi-Fi, Bluetooth, and cellular networks,	3
5.2	Threats and attacks, Firewalls, NAT,VPNs, Introduction to network management, SNMP,	4
5.3	Overview of tools and troubleshooting, Traffic analysis tools and Configuration management**.**	3

20MCA162	APPLIED STATISTICS	CATEGORY	L	T	P	CREDIT
		ELECTIVE	3	1	0	4

Preamble: This course introduces the concepts and application of probability distribution, Correlation, Regression and testing of hypothesis. The topics treated in this course have applications in Computer Science.

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

CO 1	Apply the concept of discrete probability distributions in determining the parameters of the distribution and hence to solve different problems
CO 2	Apply the concept of continuous probability distribution in solving different problems
CO 3	Apply the principles of correlation and regression in practical problems.
CO 4	Develop confidence intervals for various problems.
CO 5	Test the given hypothesis on the basis of known criteria.

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	3	3										
CO 2	3	3										
CO 3	3	3										
CO 4	3	3										
CO 5	3	3										

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember(K1)	10	10	10
Understand(K2)	20	20	20
Apply(K3)	20	20	30
Analyse(K4)			
Evaluate(K5)			
Create(K6)			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question caries 3 marks. Part B contains 2 questions from each module of which student should answer any one.Each question can have maximum 2 sub-divisions and carry 6 marks

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. With the usual notation find p for the binomial random variable X, if n = 6 and

9 p[x =4] = p[x =2] (K3)

2. Define Poisson distribution. Derive it's Mean. (K1)

3. A die is tossed twice. Getting 'a number greater than 4' is considered as success. Find the mean and variance of the probability distribution of the number of success. (K3)

Course Outcome 2 (CO2)

1. Define distribution function of a continuous random variable. Also state it’s important properties. (K1)
2. Derive the mean and variance of a continuous uniform distribution. (K4)
3. In a normal distribution 31% of the items are under 45 and 8% are over 64. Find the mean and standard deviation of the distribution. (K3)

Course Outcome 3(CO3):

1. State the principle of least squares. (K1)
2. Fit a parabola by the method of least squares, to the following data. (K3)

x:	1	2	3	4	5
y:	5	12	26	60	97

3. Compute the correlation coefficient from the following data. (K3)

x:	77	54	27	52	14	35	90	25	96	60
y:	35	58	60	40	50	40	35	56	34	42

Course Outcome 4 (CO4):

1. Differentiate parameter and statistic. (K1)
2. A random sample of 700 units from a large consignment showed that 200 were damaged. Find i) 95% and ii) 99% confidence limits for proportion of damaged limits in the consignment. (K3)
3. Explain different types of sampling. (K2)

Course Outcome 5 (CO5):

1. State Type I and Type II error. (K1)
2. Explain the different steps in testing of hypothesis. (K2)
3. In a big city 325 men out of 600 men were found to be smokers. Does this information support the conclusion that the majority of men in this city are smokers? (K5)

Syllabus

Module 1

Introduction – Random Experiment, Random Variables, Discrete Random Variables, Probability Distributions and Probability Mass Functions, Mean and Variance of a Discrete Random Variable, Discrete Uniform Distribution - Mean and Variance, Binomial Distribution - Mean and Variance, Geometric Distribution - Mean and Variance, Poisson Distribution - Mean and Variance

Module 2

Continuous Random Variables, Probability Density Functions, Mean and Variance of a Continuous Random Variable, Continuous Uniform Distribution- Mean and Variance, Normal Distribution-Mean and Variance (Proof not required), Standard Normal Distribution, Exponential Distribution.

Module 3

Curve fitting – Principle of least squares – fitting a straight line – fitting a parabola – linear correlation and regression – Karl's Pearson's Coefficient of Correlation.

Joint and marginal probability distribution - Conditional probability distribution - independent random variable (discrete case only).

Module 4

Sampling distribution – Introduction to sampling – random sampling – sampling distribution – standard error – estimation – interval estimates and confidence interval – estimation of population mean and proportions (small and large samples).

Module 5

Testing of hypothesis – introduction – basic concepts – Hypothesis concerning a mean – equality of means – Hypothesis concerning one proportion – difference of two proportions.

Text Books

1. Veerarajan T, "***Probability and Random Process***", 3rd Edition, Tata McGraw-Hill(2002)

2. Gupta S.C and Kapoor V .K, "***Fundamentals of Mathematical Statistics***", Sultan Chand and Sons (2014).

Reference Books

1. David S. Moore and George P. McCabe, "***Introduction to practice of statistics***", W.H. Freeman & Company, 5th Edition (2005).

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY KTU Estd. 2014

2. G. Jay Kerns, "***Introduction to Probability and Statistics Using R***", Chapman & Hall (2010)
3. Douglas C. Montgomery and George C. Runger, "***Applied Statistics and Probability for Engineers***", Wiley India, 5th Edition (2012).

Web Resources

1. Probability and statistics EBook
http://wiki.stat.ucla.edu/socr/index.php/EBook
2. https://www.openintro.org/stat/textbook.php
3. http://www.math.uah.edu/stat/index.html
4. Statistics Online Computational Resource
http://www.socr.ucla.edu/

Course Contents and Lecture Schedule

Topic	No. of lectures
Module 1	**9 hrs**
Introduction – Random Experiment, Random Variables, Discrete Random Variables, Probability Distributions and Probability Mass Functions, Mean and Variance of a Discrete Random Variable	3
Discrete Uniform Distribution - Mean and Variance	1
Binomial Distribution - Mean and Variance	2
Geometric Distribution - Mean and Variance, Poisson Distribution - Mean and Variance	3
Module 2	**9 hrs**
Continuous Random Variables, Probability Density Functions, Mean and Variance of a Continuous Random Variable	3
Continuous Uniform Distribution, Mean and Variance	2
Normal Distribution, Mean and Variance (Proof not required), Standard Normal Distribution	3
Exponential Distribution	1

Topic	No. of lectures
Module 3	**9 hrs**
Curve fitting – Principle of least squares – fitting a straight line – fitting a parabola	3
linear correlation and regression – Karl's Pearson's Coefficient of Correlation	2
Joint and marginal probability distribution	2
Conditional probability distribution - independent random variable (discrete case only)	2
Module 4	**9 hrs**
Sampling distribution – Introduction to sampling – random sampling	3
sampling distribution – standard error	2
estimation – interval estimates and confidence interval – estimation of population mean and proportions (small and large samples)	4
Module 5	**9 hrs**
Testing of hypothesis – introduction – basic concepts	3
Hypothesis concerning a mean – equality of means	3
Hypothesis concerning one proportion – difference of two proportions	3

20MCA164	ORGANIZATIONAL BEHAVIOUR	CATEGORY	L	T	P	CREDIT
		ELECTIVE	3	1	0	4

Preamble: This course is designed primarily for students who are being exposed to Organizational Behaviour for the first time. Primary aim of this course is to help students to understand the organizational culture and its dynamics and to acquire skills to take rational decisions in groups or organizations.

Prerequisite: Nil

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

CO 1	Identify managers' challenges and opportunities in applying OB concepts.
CO 2	Analyse various characteristics of individual behaviour and its impact on organizational performance.
CO 3	Acquire knowledge about the complexities associated with management of individual behaviour in the organization.
CO 4	Understand group behaviour and develop inter-personal skills and group dynamics.
CO 5	Understand organizational structures and analyze the behavioral implications of different organizational designs.

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	2	2	1			2		2	2		1	1
CO 2	2	2				1	2	3	3		3	1
CO 3	2	2				1	2	3	3		3	1
CO 4	2	2				1		3	3		3	
CO 5	2	2	1			2		2	2		1	

APJ ABDUL KALAM
TECHNOLOGICAL
UNIVERSITY
Estd.
2014

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember(K1)	30%	30%	30%
Understand(K2)	30%	30%	30%
Apply(K3)			
Analyse(K4)	30%	30%	30%
Evaluate(K5)	10%	10%	10%
Create(K6)			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Seminar/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question caries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Describe the importance of inter-personal skills in the workplace.
2. Analyse the challenges and opportunities for managers in using OB concepts.
3. "The workplace discriminations undermine organisational performance", Justify.

Course Outcome 2 (CO2):

1. Identify the major job attitude and job satisfaction parameters.
2. How to apply concepts about emotions and moods to specific OB issues.
3. Differentiate between person fit for job and person fit for organisation.

Course Outcome 3(CO3):

1. What is learning and what are the theories of learning?
2. How do individual differences and organisational constraints influence decision making?
3. Identify how employee involvement measures motivate employees.

Course Outcome 4 (CO4):

1. Differentiate group and team.
2. Relate the contemporary theories of leadership to earlier foundational theories.
3. What are three types of conflicts and the three loci of conflict?

Course Outcome 5 (CO5):

1. What are the functional and dysfunctional effects of organisational culture?
2. What are your suggestions to overcome resistance to change in an organization?
3. Identify the potential environmental, organisational and personal sources of stress at work.

Model Question paper

Reg No.:________________			Name:_______________________________________ _
APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY FIRST SEMESTER M.C.A.DEGREE EXAMINATION, MODEL QUESTION PAPER			
20MCA164 – Organisational Behaviour			
Max. Marks: 60			Duration: 3 Hours
PART A			
	Answer all questions, each carries 3 marks.		Marks
1		Define organisational behaviour. What is the objective of learning this subject in this programme?	3

2		What is workforce diversity? How to manage diversity?	3
3		State and explain the foundations of individual behaviour.	3
4		Differentiate ‘Classical conditioning’ and ‘Operant conditioning’ behavioural theories.	3
5		Describe how an understanding of attitudes is useful for the study of organisational behaviour.	3
6		What is job design? Describe different approaches to job design.	3
7		What is a team? Can groups become team? Defend your answer.	3
8		What is the difference between transformational leadership, transactional leadership and charismatic leadership?	3
9		What is creativity? How creativity can be enhanced in organisations?	3
10		What is Organisational Development? Why is it undertaken by organisations?	3
PART B			
Answer any one question from each module. Each question carries 6 marks.			
Module I			
11		State your views on the following statement: “People influence organizations and organizations influence people”.	6
OR			
12		Why have career management and talent management become important these days? Justify your points.	6
Module II			
13		What is personality? What are its determinants? As per your opinion, which of them are more important in shaping personality.	6
OR			
14		From your own experience, provide three examples of perceptual errors. Discuss the outcomes of each instance.	6

Module III			
15		Compare and contrast Maslow's need hierarchy theory with Herzberg's two-factory of motivation.	6
OR			
16		What is the relationship between stress and personality? What aspects of personality might tend to increase or decrease stress?	6
Module IV			
17		Why groupthink is to be avoided? How might a manager attempt to ensure that groupthink does not occur in his / her group?	6
OR			
18		What are the potential problems with upward and downward communications? How can managers alleviate these problems?	6
Module V			
19		What are the obstacles to change organisational culture? How can change be brought about?	6
OR			
20		What are the forces leading to change in organisations? Using Lewin's theory justify why the change is resisted.	6
****			

Syllabus

Module 1

Nature of Organisational Behaviour: What are Organisations? – Why do Organisations Exist? – Nature of Organisational Behaviour – Foundations of OB – Contemporary OB – Scope of Organisational Behaviour – Contextual Perspectives of OB – Evolution of OB – OB Model

Management and Managers: Functions of Management – Manager's Roles – Types of Managers –Evolution of Management Theory – Contemporary Trends in Management Thinking

Challenges in OB: Managing Inclusivity / Diversity – Career Management – Talent Management – Globalisation

Module 2

Foundations of Individual Behaviour: Personal Factors – Environmental Factors – Organisational Systems and Resources – Models of Individual Behaviour

Intelligence: Nature of Intelligence – Types of Intelligence – Model, Theories, Measurement of Intelligence – Factors Influencing Intelligence

Personality: Nature of Personality – The Shaping of Personality – Determinants of Personality – Personality Structure – OB Related Personality Traits

Perception and Attribution: Perception: Meaning and Definition – Factors Influencing Perception – Perceptual Process – Perception and OB

Learning: Explicit and Tacit Knowledge – How Learning Occurs? – Principles of Learning – Learning and OB

Module 3

Attitudes and Values: Nature of Attitudes – Components of Attitudes – Formation of Attitudes – Functions of Attitudes – Changing Attitudes – Work-related Attitudes – Values

Motivation: Nature of Motivation – Importance of Motivation – Motivational Challenges – Theories on Motivation

Applied Motivational Practices: Rewards – Job Design – Behaviour Modification – Empowerment –Problem Employees – Quality of Work Life – Employee Engagement

Work Stress: Work Stress Model – Burnout – Stress Management – Stress and Performance

Module 4

Group Dynamics: Nature of Groups – Types of Groups – Group Development – Usefulness & Pitfalls of Groups – Determinants of Group Behaviour – Group Structuring – Group Decision Making

Team Dynamics: Teams vs. Groups – Benefits from Teams – Types of Teams – Implementing Teams in Organisations – Team Properties – Effective Teamwork

Workplace Behaviour: Nature of Conflict – Changing Views of Conflict – Functional and Dysfunctional Conflict – The Process of Conflict – Levels of Conflict – Conflict Resolution – Conflict Management Styles - Managerial Implications – Negotiation and Conflict Resolution

Leadership: Nature of Leadership – Leadership and Management – Importance of Leadership – Formal and Informal Leadership – Leadership Styles and Their Implications – Theories of Leadership – Contemporary Issues on Leadership – Leadership Development

Communication: Interpersonal Communication – Organisational Communication – Communication Networks – Communication Roles – Informal Communication – Communication Media – Information Technologies – Managerial Implications

Module 5

Organisations: Nature of Organisations – Organisational Structure – Key Factors of Organisational Structure – Types of Organisational Structures – Organisations for Future – Informal Organisations – Managerial Implications

Organisational Culture: Cultural Dimensions – How is Culture Created? – Sustaining Culture – Effects of Culture – Changing Organisational Culture – Creativity in Organisations – Innovation in Organisations

Organisational Change and Development: Nature of Change – Levels of Change – Types of Change – Forces for Change in Organisations – Resistance to Change – Force Field Theory of Change - The Change Process – Organisational Development – Managerial Implications

Text Books

1. K Aswathappa, ***Organizational Behaviour***, Himalaya Publishing House, 2018.
2. Robbins, Stephen, Timothy, A & Sanghi, S. "***Organizational Behavior***",13th Edn, Pearson Education. 2009.

Reference Books

1. Mc Shane & Von Glinow, "***Organizational Behavior***", Mc Graw Hill Publications, New Delhi, 2008
2. ***Understanding Organizational Behaviour*** by Udai Pareek, Oxford University Press (Third Edition)
3. ***Behaviour in Organizations*** by Jerald Greenberg and Robert A. Baron, PHI learning private Ltd, New Delhi (Ninth Edition).
4. Laurie J. Mullins, ***Management and Organisational Behaviour***, Oxford Publishers, New Delhi, 2007.
5. ***ORGB*** by Nelson, Quick and Khandelwal, Cengage Learning New Delhi (second edition).

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY KTU Estd. 2014

Course Contents and Lecture Schedule

Topic	No. of lectures (49 Hrs)
Module 1	**9 Hrs**
Nature of Organisational Behaviour	3
Management and Managers	3
Challenges in OB	3
Module 2	**10 Hrs**
Foundations of Individual Behaviour	2
Intelligence	2
Personality	2
Perception and Attribution	2
Learning	2
Module 3	**9 Hrs**
Attitudes and Values	2
Motivation	2
Applied Motivational Practices	3
Work Stress	2
Module 4	**12 Hrs**
Group Dynamics	2
Team Dynamics	2
Workplace Behaviour	3
Leadership	3
Communication	2
Module 5	**9 Hrs**
Organisations	3
Organisational Culture	3
Organisational Change and Development	3

20MCA166	FUNCTIONAL PROGRAMMING	CATEGORY	L	T	P	CREDIT
		ELECTIVE	3	1	0	4

Preamble: This course introduces a functional programming approach in problem solving. Salient features of functional programming like recursion, pattern matching, higher order functions are discussed.

Lists and their features, new types such as Recursive types, Enumerated types, Composite and Abstract types along with their applications are being discussed with high importance.

Haskell is introduced to give a practical flavour to the course.

Prerequisite: Discrete mathematics.

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

CO 1	Understand the principles of functional programming (Module 1)
CO 2	Write purely functional programs, using recursion, pattern matching, and higher- order functions ((Module 2)
CO 3	Design immutable data structures like lists. (Module 3)
CO 4	Understand generic types for functional programs (Module 4)
CO 5	Write programs using Haskell (Module 5)

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	2	2	2			2			2			
CO 2	2	2	2			2			2			
CO 3	2	2	2			2			2			
CO 4	2	2	2			2			2			
CO 5	2	2	2		2	2	2		2			2

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	10	10	10
Understand	20	20	20
Apply	20	20	30
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance	: 8 marks
Continuous Assessment Test (2 numbers)	: 20 marks
Assignment/Quiz/Course project	: 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Design a recursive function to add two numbers.
2. Design a tail recursive function to find the nth Fibonacci number.
3. Explain the basic differences between imperative style programming and functional style programming.

4. Analyse each of the following lambda expressions to clarify its structure. If the expression is a function, identify the bound variable and the body expression, and then analyse the body expression. If the expression is an application, identify the function and argument expressions, and then analyse the function and argument expressions:

   i) λa.(a λb.(b a))

   ii) λx.λy.λz.((z x) (z y))

   iii) (λf.λg.(λh.(g h) f) λp.λq.p)

   iv) λfee.λfi.λfo.λfum.(fum (fo (fi fee))

   v) (λp.(λq.p λx.(x p)) λi.λj.(j i))

Course Outcome 2 (CO2)

1. Explain with the help of examples the various forms of function definitions.
2. Explain functional composition with the help of examples.
3. Deduce the type of the following expression:

   (.) f g x = f (g x) where . -> Functional Composition.

Course Outcome 3(CO3):

1. Predict the output of the following along with detailed explanation on how did you arrive at the answer:
   A. [(a,b) | a <- [1 . . 8] ; even a; b <- [a + 3. . 4] ; odd b]
   B. ["Party" | k <- [1 .. 5]]
   C. [' * ' | i <- [1 .. 3] ; j <- [1, 2]]

2. Explain any three list operations along with function definitions and examples.

Note: Questions can be asked to solve problems using list comprehensions, to prove properties on list operations and functions on natural numbers using Mathematical Induction.

Course Outcome 4 (CO4):

1. Define Natural numbers as a Recursive Type and explain how this definition enumerates all Natural numbers.

2. Find the equivalent decimal representation of this value:

Succ (Succ (Succ (Succ (Succ (Succ Zero)))))

3. Define Fibonacci numbers using Pattern matching. Natural numbers should be represented as a Recursive type.

Note: Questions can be asked to prove properties on Binary Trees and Binary Search Trees using Structural Induction (Variant of Mathematical Induction.

Course Outcome 5 (CO5):

1. Duplicate only even numbers among the elements of a list using a Haskell function and explain. You need to do this in two ways; 1. Recursion 2. List Comprehension

Example : λ> dupli [1, 2, 3] ANS: [2,2]

Model question Paper

Duration : 3 Hours **PART A** **Total : 60 Marks**

1. Design a recursive function to add two numbers.
2. Can Arrays be used as a data structure in functional programming? Explain with reasons.
3. Explain functional composition with the help of an example.
4. Deduce the type of the following expression:

   (.) f g x = f (g x) where . -> Functional Composition.

5. Predict the output of the following along with detailed explanation on how did you arrive at the answer:
   a. [(a,b) | a <- [1 . . 8] ; even a; b <- [a + 3. . 4] ; odd b]
   b. ["Party" | k <- [1 .. 5]]
   c. [' * ' | i <- [1 .. 3] ; j <- [1, 2]]
6. Define the function “take”. For example, take does the following:

   Prelude> take 2 [1,2,3,4] [1,2].

7. Find the equivalent decimal representation of this value:

   Succ (Succ (Succ (Succ (Succ (Succ Zero)))))

8. Explain composite types with the help of an example.
9. Define a Haskell function to find the factorial of a given number.
10. Define a Haskell function to reverse the elements of a list.

(3 x 10 = 30 Marks)

PART B

11. Explain commonly used data types in functional programming along with their properties.

OR

12. Write Recursive definitions along with an explanation for the below Arithmetic operations. Illustrate the recursive flow with the help of a diagram.

    1. add x y   2. mult x y   3. div x y

13. Explain the following along with suitable examples:

    1. Currying in Functional Programming. 2. Strict and Non strict functions.

OR

14. Explain the following along with suitable examples:

    1. Guards and Pattern matching.2. Inverse functions.

15. Given below the definition of a function funcky

```
funcky:: Int ->Int

funcky n

|n == 0 = 0

|otherwise = 1 + funcky ( n−1)
```

    Predict the output of f for all n >= 0? Prove your answer.

OR

16. Explain any three list operations along with function definitions and examples.

17. Explain Recursive Data Types with the help of an example.

OR

18. Give the type definition of a binary tree along with explanation of two functions on binary trees.

19. Duplicate only even numbers among the elements of a list using a Haskell function and explain. You need to do this in two ways; 1. Recursion 2. List Comprehension

Example : λ> dupli [1, 2, 3] ANS: [2,2]

OR

20. Define a queue data type in Haskell along with any two operations on it as well as examples.

(6 x 5 = 30 Marks)

SYLLABUS

Module I:

Review of recursion -Tail recursion -recursive program design- Functional Programming: Introduction, λ calculus, λ expressions, Identity function, Self application function, Function application function, Notation for naming functions and application reduction, Functions from functions, Argument selection and argument pairing functions,Free and bound variables,Name clashes and α conversion, Simplification through eta reduction, Conditions, Booleans and Integers, Recursion and Arithmetic, Expressions and values, Basic Data Types , Names and values in programming- Data structures in functional languages - Names and values in imperative and functional languages- Execution order in imperative and functional languages- Repetition in imperative and functional languages- Functions as values.

(Note : Recursion is a very important technique in functional programming, hence high importance needs to be given to make students understand the essentials of recursive thinking and program design, Basic Lambda (λ) calculus needs to be taught.)

Module II:

Functions: Functions and definitions, Functional composition, Operators, Inverse functions, Strict and non-strict functions, Type Inference.

(Note : Basic ways of defining functions, how to infer the types of variables and function needs to be taught)

Module III:

Lists: List notation, List comprehensions, Operations on lists, Map and filter, List patterns, Recursion and Induction: Over natural numbers, Over lists. Operations on lists

(Note : Mathematical Induction based Proofs needs to be taught from the reference text book.)

Module IV:

New Types : Enumerated types , Composite types , Recursive types , Abstract types , Trees: Binary trees , Binary search trees

(Note : Various definitions of properties of these new types, their property proofs etc needs to be taught)

Module V:

Programming with Haskell: Introduction to Haskell, Defining functions: guards, pattern matching and recursion, Lists, strings and tuples, Types and polymorphism, Higher order functions on lists: map, filter, list comprehension, User defined data types:lists, queues, trees

(Note : Students need to be taught how to program using Haskell in this module)

Text Books

1. Richard S. Bird, Philip Wadler, “Introduction to Functional Programming”,Prentice Hall , 1988 (Module 1,2,3,4)
2. Greg Michaelson, “An introduction to functional programming through lambda calculus”, Dover Publications, 2011 (Module 1)
3. Miran Lipovača, “Learn You a Haskell for Great Good!: A Beginner's Guide”, No Starch Press, 1st Edition (15 March 2011) (Module 5)

Reference Books and Sites

1. Simon Peyton Jones , “The Implementation of Functional Languages” , Prentice Hall.
2. Benjamin C. Pierce, " Types and Programming Languages", MIT Press, 2002
3. https://www.haskell.org/
4. http://learnyouahaskell.com

Course Contents and Lecture Schedule

No	Topic/Module	No. of Lectures
1	Review of recursion -Tail recursion -recursive program design- Functional Programming: Introduction, λ calculus, λ expressions, Identity function, Self application function, Function application function, Notation for naming functions and application reduction, Functions from functions, Argument selection and argument pairing functions,Free and bound variables,Name clashes and α conversion, Simplification through eta reduction, Conditions, Booleans and Integers, Recursion and Arithmetic, Expressions and values, Basic Data Types , Names and values in programming- Data structures in functional languages - Names and values in imperative and functional languages- Execution order in imperative and functional languages- Repetition in imperative and functional languages- Functions as values.	10
2	Functions: Functions and definitions, Functional composition, Operators, Inverse functions, Strict and non-strict functions, Type Inference.	8
3	Lists: List notation, List comprehensions, Operations on lists, Map and filter, List patterns, Recursion and Induction: Over natural numbers, Over lists. Operations on lists	10
4	New Types : Enumerated types , Composite types , Recursive types , Abstract types , Trees: Binary trees , Binary search trees	10
5	Programming with Haskell: Introduction to Haskell, Defining functions: guards, pattern matching and recursion, Lists, strings and tuples, Types and polymorphism, Higher order functions on lists: map, filter, list comprehension, User defined data types:lists, queues, trees	10

20MCA168	VIRTUALISATION AND CONTAINERS	CATEGORY	L	T	P	CREDIT
		ELECTIVE	3	1	0	4

Preamble:

A graduate course in Computer Applications should give due exposure to the recent developments. Since virtualization and containers are the technologies that drive the majority of the day to day applications and industry, this course is designed to build upon the knowledge acquired at the undergraduate/graduate level on Operating Systems and familiarise students with virtualization and container technologies.

Prerequisite: Operating Systems

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

CO 1	Understand the basics of virtualization technology, architecture, limitations and applications.
CO 2	Apply Networking Principles to setup virtual machines and connect to the network
CO 3	Understand the basics of VM life cycle, VM migrations, VM scheduling and load balancing
CO 4	Understand Container fundamentals including how to configure and set up a container
CO 5	Understand the basics of security, troubleshooting and monitoring aspects in container technology
CO 6	Apply the knowledge in Virtualization and docker to setup VM and dockers.

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	2	1										
CO 2	3	2	1	1			1					
CO 3	2	1					1					
CO 4	2	1					1					
CO 5	2	1					1					
CO 6	3	2	1	1			1					

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	10	10	10
Understand	20	20	20
Apply	20	20	30
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance	: 8 marks
Continuous Assessment Test (2 numbers)	: 20 marks
Assignment/Quiz/Course project	: 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

(a) Explain the need and applications of virtualization.

(b) Describe the hypervisor architecture.

(c) Mention tools and technologies used in virtualization.

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY KTU Estd. 2014

Course Outcome 2 (CO2)

(a) Describe IP addressing

(b) Explain the concept of paging and virtual memory

Course Outcome 3(CO3):

(a) Describe the VM life cycle.

(b) Explain VM provisioning, VM scheduling and load balancing

(c) Write in detail the KVM architecture and commands

Course Outcome 4 (CO4):

(a) Discuss the container fundamentals and different container technologies.

(b) Explain the container orchestration and clustering.

Course Outcome 5 (CO5):

(a) Discuss the concepts of security and isolution in containers.

(b) Explain troubleshooting, monitoring and alerting in containers.

Course Outcome 6 (CO6):

(a) Explain how to configure and set up virtual machines.

(b) Describe the configuring and setting up of containers.

Model Question paper

Reg No.:________________		Name:______________________________________ _	
APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY FIRST SEMESTER M.C.A.DEGREE EXAMINATION, MODEL QUESTION PAPER			
20MCA168 VIRTULAISATION AND CONTAINERS			
Max. Marks: 60			Duration: 3 Hours

PART A			
		Answer all questions, each carries 3 marks.	Marks
1		Explain the different types of hypervisors.	(3)
2		What are the types of virtualization ?	(3)
3		Explain the difference between private IP and Public IP.	(3)
4		What is virtual memory.	(3)
5		Explain any three commands in KVM and their uses.	(3)
6		What do you mean by virtual machine migration?	(3)
7		Write a short note on the container creation process.	(3)
8		Explain virtual networking in containers.	(3)
9		How are the running containers controlled	(3)
10		Explain troubleshooting in container technologies.	(3)
PART B			
Answer any one question from each module. Each question carries 6 marks.			
Module I			
11		Briefly explain the hypervisor architecture.	(6)
OR			
12		Explain difference between virtualization and virtual computing.	(6)
Module II			
13		Briefly explain the different modes in networking can be set up in virtual machines.	(6)
OR			
14		Explain how paging is important in the context of virtualization.	(6)

Module III			
15		Briefly explain the VM life cycle.	(6)
OR			
16		Describe any VM scheduling algorithm.	(6)
Module IV			
17		Explain the Breadth First Search algorithm with a suitable example.	(6)
OR			
18		Compare virtual machines and containers	(6)
Module V			
19		Explain the importance of security and isolation in Docker.	(6)
OR			
20		Explain the business context of containers	(6)

Syllabus

Module 1 *(10 Hours)*: Physical and virtual machines, Traditional and virtual computing, Understanding virtualization, Need and Applications of virtualization, Limitations, Simulations and Emulations, Challenges in Virtualized environment, tools and technologies in virtualized environments. Types of Hypervisors, Hypervisor architecture

Module 2 *(8 Hours)*: IP addressing - Private address, Public address, virtual LAN, Memory addressing, Paging, Memory mapping, virtual memory, complexities and solutions of memory virtualization

Module 3 *(14 Hours)***:** VM lifecycle, Process and system level VMs, VM configurations, VM migrations, Migration types and process, VM provisioning, Scaling,VM scheduling, Load balancing: Significance, Types and Algorithms. Case study : KVM, KVM architecture, KVM commands

Module 4 *(10 Hours)***:** Container fundamentals, Containers versus virtual machines, Different container technologies, Configuring a container engine, Container virtual networking, Container orchestration and clustering, Images and containers. Case study : Docker

Module 5 *(6 Hours)***:** Working with remote repositories, Security and isolation, Troubleshooting, Monitoring and alerting, Controlling running containers, Containers in a business context

References

1. Chris Wolf , Erick M. Halter, ***Virtualization: From the Desktop to the Enterprise***, APress 2005.

2. Kumar Reddy, Victor Moreno, ***Network virtualization***, Cisco Press, July, 2006.

3. James E. Smith, Ravi Nair, ***Virtual Machines: Versatile Platforms for Systems and Processes***, Elsevier/Morgan Kaufmann, 2005

4 Matthew Portnoy, ***Virtualization Essentials***, Wiley; Second edition (2016)

5. Sean P. Kane, Karl Matthias, ***Docker: Up & Running - Shipping Reliable Containers in Production***, Second Edition, O'Reilly

Web References

1. https://www.linux-kvm.org/page/Main_Page
2. https://docs.docker.com/get-started/

Course Contents and Lecture Schedule

No	Topic	No. of Lecture Hours
1	**Introduction**	
1.1	Physical and virtual machines	1
1.2	Traditional and virtual computing	1
1.3	Understanding virtualization	1
1.4	Need, Applications and Limitations of virtualization	2
1.5	Simulations and Emulations	1
1.6	Challenges in Virtualized environment	1
1.7	Tools and technologies in virtualized environments	1
1.8	Types of Hypervisors	1
1.9	Hypervisor architecture	1
2	**Network Virtualization**	
2.1	IP addressing	1
2.2	Private address and Public Addresses	1
2.3	Virtual LAN	1
2.4	Memory addressing	1
2.5	Paging	1
2.6	Memory mapping	1
2.7	Virtual memory	1
2.8	Complexities and solutions of memory virtualization	1

No	Topic	No. of Lecture Hours
3	**Virtual Machine**	
3.1	VM lifecycle	1
3.2	Process and system level VMs	1
3.3	VM configurations	1
3.4	VM migrations	1
3.5	Migration types and process	1
3.6	VM provisioning	1
3.7	Scaling,VM scheduling	2
3.8	Load balancing: Significance	1
3.9	Types and Algorithms	1
3.10	Case study : KVM	1
3.11	KVM architecture	1
3.12	KVM commands	2
4	**Containers**	
4.1	Container fundamentals	1
4.2	Containers versus virtual machines	1
4.3	Different container technologies	1
4.4	Configuring a container engine	1
4.5	Container virtual networking	1
4.6	Container orchestration and clustering	1
4.7	Images and containers	1
4.8	Case study : Docker	3
5	**Security and Management**	
5.1	Working with remote repositories	1
5.2	Security and isolation	1
5.3	Troubleshooting	1
5.4	Monitoring and alerting	1
5.5	Controlling running containers	1
5.6	Containers in a business context	1

20MCA172	ADVANCED OPERATING SYSTEMS	CATEGORY	L	T	P	CREDIT
		ELECTIVE	3	1	0	4

Preamble: This course intends to provide insight into more Advanced Operating Systems. Detailed discussion on various concepts like process synchronization, mutual exclusion, resource sharing, concurrency control and security are discussed at algorithm level. Various kinds of advanced operating systems like Distributed Systems, Multiprocessor systems, and Database Systems are included to the level possible within the scope of a single course. More detailed treatment can be done through seminars , assignments and talks by eminent external experts .

Prerequisite: Basic concepts of desktop computer operating systems.

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

CO 1	Identify synchronization problems in operating systems and issues in distributed systems.
CO 2	Explain classification of mutual exclusion algorithms and security violations.
CO 3	Explain the design of distributed shared memory and issues in load distribution.
CO 4	Explain design issues and synchronization in multiprocessor systems.
CO 5	Explain synchronization and concurrency control in database systems.

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	2	2			2				1			
CO 2	2	1							1			
CO 3	2	1							1			
CO 4	2	1							1			
CO 5	2	2			1		1		1			

Assessment Pattern

Bloom's Category	**Continuous Assessment Tests**		**End Semester Examination**
	1	**2**	
Remember	10		10
Understand	20	20	20
Apply	20	20	20
Analyse		10	10
Evaluate			
Create			

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
Estd.
2014

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance	: 8 marks
Continuous Assessment Test (2 numbers)	: 20 marks
Assignment/Quiz/Course project	: 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Explain synchronization using semaphore.
2. Classify Advanced operating systems.
3. Illustrate limitation of Lamports clocks.

Course Outcome 2 (CO2)

1. Explain some of the algorithms for mutual exclusion.
2. Explain potential security violations.
3. Compare the Lamport's algorithm and Rickart-Agarwala algorithm.

Course Outcome 3(CO3):

1. Explain major design issues and building mechanisms in Distributed file systems.
2. Explain important algorithms for implementing DSM.
3. Explain issues in load distribution.

Course Outcome 4 (CO4):

1. Explain system architecture of Multiprocessor systems.
2. Explain design issues in Database Multiprocessor Systems.
3. Explain how virtualization is implemented.

Course Outcome 5 (CO5):

1. Explain Lock based algorithms for concurrency control in Database Systems.
2. Illustrate Timestamp based algorithms for concurrency control in Database Systems.
3. Explain design issues in Database Systems.

Model Question paper

Part A

1. Categorize various advanced operating systems.
2. Illustrate synchronization using semaphore.
3. Explain potential security violations.
4. Explain requirements of mutual exclusion.
5. What is the difference between load balancing and load sharing?
6. Which are the major components of a load distributing algorithm?
7. Explain the interconnection network in multiprocessors?
8. Explain the structure of Multiprocessor of Operating Systems.
9. Explain what is meant by serializability.
10. What is meant by Log equivalence? (3 x 10 = 30 Marks)

Part B

Module 1

11. Identify any six issues that are common with Distributed systems. [6 Marks]

OR

12. Write a note on the following
    a. mutex [3 marks]
    b. semaphore [3 marks]

Module 2

13. Write short notes on
    a. Rickart-Agarwala Algorithm [3 Marks]
    b. Lamport’s algorithm. [3 Marks]

OR

14. Explain any six Design Principles for Secure Systems. [6 Marks]

Module 3

15. Identify major design issues in Distributed File systems.

[6 Marks]

OR

16. Write any two algorithms for implementing DSM

[6 Marks]

Module 4

17. Explain Multiprocessor System Architectures and Interconnection Networks.

[6 Marks]

OR

18. Discuss the synchronization of processes in Multiprocessors.

[6 Marks]

Module 5

19. Explain the basic Synchronization Primitives for Concurrency Control in Database systems.

[6 Marks]

OR

20. Write and explain an optimistic algorithm for concurrency control in database systems.

[6 Marks]

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY KTU Estd. 2014

Syllabus

Module	Contents	Hours
I	**Overview**: Functions of Operating System –Design Approaches –Types of Advanced Operating Systems. **Synchronization Mechanisms**: Concept of Processes and Threads –The Critical Section Problem – Other Synchronization Problems:– Monitor –Serializer – Path Expressions. **Distributed Operating Systems**:- Issues in Distributed Operating System – Communication Networks And Primitives –Lamport's Logical clocks – Causal Ordering of Messages.	10
II	**Distributed Mutual Exclusion**:- Classification - Requirements – Measuring Performance – Lamport's Algorithm – Rickart-Agarwala Algorithm – Suzuki-Kasami's Broadcast Algorithm. : **Security** Potential Security Violations – Design Principles for Secure Systems –The Access Matrix Model and Implementation- The Access Control list Method.	10
III	**Distributed Resource Management**: Mechanisms for building Distributed File Systems – Design Issues - Distributed Shared Memory – Algorithms for Implementing Distributed Shared memory – Issues in Load Distributing – Components of Load Distributing Algorithm – Sender-Initiated Algorithm – Receiver- Initiated Algorithm.	10
IV	**Multiprocessor Operating Systems**: Basic Multiprocessor System Architectures – Interconnection Networks – Structures – Design Issues – Threads – Process - Synchronization – Processor Scheduling – Memory Management – Virtualization – Types of Hypervisors – Paravirtualization – Memory Virtualization – I/O Virtualization.	8
V	**Database Systems**: Problem of Concurrency Control – Serializability – Basic Synchronization Primitives for Concurrency Control – Lock-Based Algorithms – Time-Stamp Based Algorithms – Optimistic Algorithms.	10

Textbooks:

1. Mukesh Singhal and Niranjan G. Shivaratri, "***Advanced Concepts in Operating Systems*** – Distributed, Database, and Multiprocessor Operating Systems", Tata McGraw-Hill, 2001.
2. Andrew S. Tanenbaum, "***Modern Operating Systems***", 3rd Edition, Prentice Hall, 2012.

Reference Books:

1. Pradeep K Sinha, "***Distributed Operating Systems: Concepts and Design***", Prentice Hall of India, 2007.
2. George Coulouris, Jean Dollimore,Tim Kindberg, Gordon Blair, ***"Distributed Systems,Concepts and Design"***, 5th Edtn,Pearson, 2019
3. https://www.classcentral.com/course/udacity-advanced-operating-systems-1016
4. https://www.my-mooc.com/en/mooc/advanced-operating-systems--ud189/

Course Contents and Lecture Schedule

No	**Topic**	**No. of Lectures**
1	**Introduction**	
1.1	**Overview**: Functions of Operating System –Design Approaches –Types of Advanced Operating Systems.	2
1.2	**Synchronization Mechanisms**: Concept of Processes and Threads – The Critical Section Problem – Other Synchronization Problems:– Monitor –Serializer – Path Expressions.	4
1.3	**Distributed Operating Systems**:- Issues in Distributed Operating System – Communication Networks And Primitives –Lamport's Logical clocks – Causal Ordering of Messages	4
2		
2.1	**Distributed Mutual Exclusion**:- Classification - Requirements – Measuring Performance – Lamport's Algorithm –	2
2.2	. Rickart-Agarwala Algorithm – Suzuki- Kasami's Broadcast Algorithm.	3
2.3	**Security :**Potential Security Violations – Design Principles for Secure Systems –The Access Matrix Model and Implementation- The Access Control list Method	5

No	Topic	No. of Lectures
3		
3.1	**Distributed Resource Management**: Mechanisms for building Distributed File Systems – Design Issues.	3
3.2	**Distributed Shared Memory** – Algorithms for Implementing Distributed Shared memory	3
3.3	**Load Distribution** : Issues in Load Distributing – Components of Load Distributing Algorithm – Sender- Initiated Algorithm – Receiver-Initiated Algorithm.	4
4		
4.1	**Multiprocessor Operating Systems**: Basic Multiprocessor System Architectures – Interconnection Networks – Structures – —.	3
4.2	Design Issues – Threads – Process Synchronization - Processor Scheduling – Memory Management	3
4.3	Virtualization – Types of Hypervisors – Paravirtualization – Memory Virtualization – I/O Virtualization.	2
5		
5.1	**Database Systems**: Problem of Concurrency Control – Serializability – Basic Synchronization Primitives for Concurrency Control – Lock-Based Algorithms-	5
5.2	Time-Stamp Based Algorithms	3
5.3	Optimistic Algorithms	2

20MCA182	BUSINESS MANAGEMENT	CATEGORY		L	T	P	CREDIT
		ELECTIVE		3	1	0	4

Preamble: The primary aim of this course is to understand basic principles of management and accounting. In our day to day life managers will have to manage so many resources in the present day complex business environment. By effective and efficient management the goals of the organisation can be attained. This course is intended to give an idea regarding managing the resources for the effective performance of the organisation and decision making in everyday life. Basic idea regarding book keeping and accounting is also required for managers for taking decisions.

Prerequisite: NIL

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

CO 1	Understand management as a process.
CO 2	Critically analyse and evaluate management theories and practices
CO 3	Perform planning and organising for an organisation
CO 4	Do staffing and related human resource development function
CO 5	Take proper decisions to get competitive advantage
CO 6	Understand basic concepts in book keeping and accounting.

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

	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	3							2				
CO 2		3										
CO 3			3									3
CO 4											3	
CO 5					3	2						
CO 6	3							3				

Assessment Pattern

Bloom's Category	**Continuous Assessment Tests**		**End Semester Examination**
	1	**2**	
Remember	10	10	20
Understand	20	20	20
Apply	20	20	20
Analyse			
Evaluate			

Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance	: 8 marks
Continuous Assessment Test (2 numbers)	: 20 marks
Assignment/Quiz/Course project	: 12 marks

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 any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Sample Questions

Course Outcome CO1:

Describe various functions of management.

Course Outcome CO2 :

Explain different theories of management thought.

Course Outcome CO3:

Illustrate different steps in planning.

Course Outcome CO4:

Describe different types of training methods for employees in an organisation.

Course Outcome CO5:

Explain the decision process in an organisation with case example.

Course Outcome CO6:

Explain the procedure of preparation of balance sheet with a simple example.

Model Question Paper

Total Pages:

Reg No.:________________ Name:____

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
SECOND SEMESTER M.C.A. DEGREE EXAMINATION

Course Code: 20MCA182

Course Name: BUSINESS MANAGEMENT

Max. Marks: 60 Duration: 3 Hours

PART A

Answer all questions, each carries3 marks.

			Marks
1		Define management. What are the levels of management?	(3)
2		Distinguish between efficiency and effectiveness in management.	(3)
3		Explain system approach in management.	(3)
4		Illustrate different types of plans	(3)
5		Explain matrix form of organisation.	(3)
6		What is meant by job analysis?	(3)
7		Explain bench marking.	(3)
8		What is product life cycle?	(3)
9		Explain the rules of debit and credit.	(3)
10		Explain the advantages of accounting softwares.	(3)

PART B

Answer six questions, one full question from each module and carries 6 marks.

Module I

11		What are the different roles that managers play in an organisation?	(6)

OR

12		Explain the major contributions of F W Taylor to scientific management.	(6)

Module II

13		Explain various steps involved in planning with a case example.	(6)

OR

14		Explain any 3 types of organisation structures.	(6)

Module III			
15		Explain various steps involved in selection of employees for an organisation.	(6)
OR			
16		Describe different types of training methods for employees in an organisation.	(6)
Module IV			
17		Illustrate the decision process in an industry by giving different steps involved in it.	(6)
OR			
18		Explain the marketing mix elements with a case example.	(6)
Module V			
19		What is a Journal? Explain the rules of journalising	(6)
OR			
20		What are final accounts? Explain the procedure of preparing balance sheet with a simple example.	(6)

Syllabus

Module I

Introduction to Management: Basic Managerial Concepts, Levels of management, Managerial Skills, Managerial role. Management functions- Planning, Organising, Staffing, leading and Controlling.

Early Contributions in Management: Management thought - Classical approach, scientific management, contributions of Taylor, Gilbreths, Fayol’s 14 principles of management.

Human relation approach - contribution of Elton Mayo Systems approach - organization as an open system and Contingency approach

Module II

Planning: Nature and importance of planning, types of plans - Steps in planning, Levels of planning - The Planning Process - MBO definition and process, SWOT Analysis, importance.

Organising : Nature of organizing,-span of control in management, factors affecting span of control- Authority and responsibility.

Organisation structure - Formal and informal, Types of organization structure line, line and staff, functional, divisional, project, matrix, virtual form of organisations.

Module III

Staffing and related HRD Functions: meaning, nature, staffing process, Job analysis and manpower planning, job description and job specification, Recruitment & selection, selection process, Tests and interviews. Training and development - concept and methods ,Performance appraisal- concept and methods.

Module IV

Managerial Decision Making and controlling : Decision making –types of decisioins, decision making process, Decision Making Tools, Importance of controlling, Techniques of controlling- Break Even Analysis, Budgetary Control - Benchmarking –importance and limitations of benchmarking, Six Sigma importance, limitations and process of six sigma, Total Quality Management- Introduction to marketing management-Marketing mix- product life cycle

Module V

Book- Keeping and Accountancy -Elements of Double Entry -Book- Keeping - rules for journalizing -Ledger accounts –Cash book- – Trial Balance- Method of Balancing accounts- the journal proper (simple problems). Final accounts: Preparation of trading and profit and loss Account- Balance sheet (with simple problems) - Introduction to Accounting packages (Description only)

References

1. L M Prasad, "***Principles of Management***", Sultan Chand & Sons, 8th Edition (2010)
2. Peter F Drucker, "***The Practice of Management***", Butterworth-Heinemann publication, 2nd Edition (2007)
3. Harold Koontz and Heinz Weihrich, "***Essentials of Management***", McGraw Hill Education, 10th Edition (2015).
4. Robbins and Coulter, ***Management***, Pearson Education 13th Edition, 2016,
5. R N Gupta, "***Principles of Management***", S. Chand & Company Ltd., (2010)
6. Tripathi, "***Principles of Management***", McGraw Hill Education, 5th Edition (2012)
7. ***Double Entry book Keeping*** – Batliboi
8. ***A Systematic approach to Accounting***: Dr K.G. Chandrasekharan Nair

Suggested MOOCs

1. Management Functions http://nptel.ac.in/courses/122108038/
2. Leadership http://nptel.ac.in/courses/110105033/33

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	Introduction to Management: Basic Managerial concepts	2
1.1	Levels of management, Managerial Skills	2
1.2	Management roles	1
1.3	Management functions	2
1.4	Early Contributions in Management: Management thought - Classical approach, scientific management, contributions of Taylor, Gilbreths, Fayol's 14 principles of management. Human relation approach - contribution of Elton Mayo Systems	3
2	Planning: Nature and importance of planning, types of plans - Steps in planning, Levels of planning - The Planning Process	2
2.1	MBO definition and process, SWOT Analysis, importance.	2
2.2	Organising : Nature of organizing,-span of control in management, factors affecting span of control- authority and responsibility.	2
2.3	Organisation structure - Formal and informal, Types of organization structure line, line and staff, functional, divisional, project, matrix, virtual form of organisations	2
3	Staffing and related HRD Functions: meaning, nature, staffing process.	2
3.1	Job analysis and manpower planning, job description and job specification	2
3.2	Recruitment & selection, selection process, Tests and interviews. Training and development - concept and methods	3
3.3	Performance appraisal - concept and methods.	2
4	Managerial Decision Making and controlling : Decision making –types of decisions, decision making process, Decision Making Tools.	2
4.1	Importance of controlling, Techniques of controlling- Break Even Analysis, Budgetary Control	2
4.2	Benchmarking –importance and limitations of benchmarking	2
4.3	Six Sigma importance, limitations and process of six sigma,	2
4.4	Total Quality Management-	2
4.5	Introduction to marketing management-Marketing mix- product life cycle	2

No	Topic	No. of Lectures
5	Book- Keeping and Accountancy -Elements of Double Entry -Book-Keeping	2
5.1	Rules for journalizing -Ledger accounts –Cash book-	3
5.2	Trial Balance- Method of Balancing accounts- (simple problems). Final accounts: Preparation of trading and profit and loss Account- Balance sheet (with simple problems)	3
5.3	Introduction to Accounting packages.	2

20MCA184	EMBEDDED SYSTEMS	CATEGORY	L	T	P	CREDIT
		ELECTIVE	3	1	0	4

Preamble: This course introduces students to the basic concepts behind Embedded Systems. It helps the students to understand the various techniques involved in embedded system design and development.

Prerequisite:

- 20MCA103 Digital Fundamentals & Computer Architecture.
- 20MCA107 Advanced Software Engineering.
- Basic knowledge of the subjects Operating Systems and System Software.

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

CO 1	Understand the basic concepts of Embedded Systems and its Applications.
CO 2	Demonstrate the role of individual components involved in a typical embedded system.
CO 3	Learn about the co-design approach for embedded hardware and firmware development.
CO 4	Understand the concepts involved in Embedded System Design and development Process.
CO 5	Learn about techniques used in the Integration and Testing of Embedded Hardware and Firmware.
CO 6	Understand the basic concepts of RTOS based Embedded System Design.

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	2											
CO 2	3	3										
CO 3	3	3			3							
CO 4			3	2								
CO 5		3										
CO 6		3										

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember(K1)	10	10	10
Understand(K2)	20	20	20
Apply(K3)	20	20	30
Analyse(K4)			
Evaluate(K5)			
Create(K6)			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance	: 8 marks
Continuous Assessment Test (2 numbers)	: 20 marks
Assignment/Quiz/Course project	: 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question caries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Define Embedded System.
2. Illustrate the major Applications of Embedded System.
3. List out the classifications of Embedded System.

Course Outcome 2 (CO2):

1. Illustrate the components of an embedded System with the help of relevant diagram.
2. Explain about the processor Embedded into a System.

Course Outcome 3 (CO3):

1. Describe the Fundamental Issues in Hardware Software Co-Design.
2. Explain UML with the help of an example.

Course Outcome 4 (CO4):

1. Describe any three Digital Electronic Components used in the embedded Hardware development.
2. Explain about Embedded Firmware Design Approaches.

Course Outcome 5 (CO5):

1. Explain any one technique used for the Integration of Hardware and Firmware.
2. List out the techniques used for the Testing of Embedded Systems.

Course Outcome 6(CO6):

1. Define RTOS.
2. Describe How you will Choose an RTOS.

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

Model Question Paper

Course Code: 20MCA184

Course Name: Embedded Systems

Max. Marks: 60		Duration: 3 Hours

PART A

		Answer all questions, each carries 3 marks.	Marks
1		Define embedded computing system? Write two functionalities of an embedded system.	(3)
2		What are the building blocks and devices of hardware in an embedded system?	(3)
3		Describe the Design Process in Embedded System.	(3)

4		Explain about Formalism of System Design.	(3)
5		Illustrate Data Flow Graph Model with the help of relevant diagram.	(3)
6		Define State Machine Model with the help of suitable example.	(3)
7		Describe the Analog Electronic Components in Embedded Hardware Design.	(3)
8		Illustrate the Super Loop Based firmware development approach.	(3)
9		How will you Test Embedded Systems?	(3)
10		Define Real Time Operating System.	(3)
		PART B	
		Answer any one question from each module. Each question carries 6 marks.	
		Module I	
11		Differentiate Embedded system and General Computing System.	(6)
		OR	
12		Explain about the Classification of Embedded systems.	(6)
		Module II	
13		Explain in detail about Design Challenges in Embedded System Design.	(6)
		OR	
14		Explain about the Hardware-Software Co-Design in an Embedded System.	(6)
		Module III	
15		With the help of suitable diagrams explain about UML Building Blocks.	(6)
		OR	
16		Describe the Fundamental Issues in Hardware Software Co-Design.	(6)
		Module IV	
17		Explain about any three Digital Electronic Components in Embedded systems with the help of suitable diagrams.	(6)
		OR	
18		Explain about Embedded Firmware Development Languages.	(6)
		Module V	

19		Explain in detail about the commonly used techniques for the Integration of Hardware and Firmware.	(6)
		OR	
20		A lot of factors needs to be analysed carefully before making a decision of choosing an RTOS. Justify.	(6)
		****	

Syllabus

Module 1

Introduction to Embedded Systems: Embedded system, Embedded system Vs General Computing System, Processor Embedded into a System, Embedded Hardware units and devices in a system, Embedded Software in a System, Introduction to embedded system design, Classification of Embedded systems, Skills Required for an embedded system Designer, Examples of the Embedded Systems. Major Application Areas of Embedded Systems, Purpose of Embedded Systems.

Module 2

Embedded System Design and development Process: Embedded System-On-Chip (SoC) and Use of VLSI Circuit Design Technology, Build Process for embedded systems, Design Process in Embedded System, Design Challenges in Embedded System Design, Hardware-Software Co-Design in an Embedded System, Formalism of System Design.

Module 3

Hardware Software Co-Design and Program Modelling: – Fundamental Issues in Hardware Software Co-Design, Computational Models in Embedded Design - Data Flow Graph Model, Control Data Flow Graph, State Machine Model, Sequential Program Model, Concurrent / Communicating Process Model, Object oriented model, UML.

Module 4

Design and Development of Embedded Product:

Embedded Hardware Design and Development: - Analog Electronic Components, Digital Electronic Components, VLSI and Integrated Circuit Design.

Embedded Firmware Design and Development: - Embedded Firmware Design Approaches, Embedded Firmware Development Languages.

Module 5

Integration and Testing of Embedded Hardware and Firmware: - Integration of Hardware and Firmware, Testing Embedded Systems.

RTOS based Embedded System Design: - Basic operating system services, Introduction to Real Time Operating System(RTOS), RTOS Task-Scheduling models, How to Choose an RTOS.

Text Books

1. Raj Kamal, Embedded Systems: Architecture, Programming and Design, Third Edition, McGraw Hill Education (India), 2014.
2. Shibu K.V., Introduction to Embedded Systems, McGraw Hill Education (India), 2009.

Reference Books

1. J Staunstrup and Wayne Wolf, Hardware / Software Co-Design: Principles and Practice, Prentice Hall.
2. Jean J. Labrose, Micro C/OS II: The Real Time Kernel, 2e, CRC Press, 2002.
3. Steave Heath, Embedded System Design, Second Edition, Elsevier.
4. Wayne Wolf , Computers as Components-Principles of Embedded Computer System Design, Morgan Kaufmann publishers, Third edition, 2012.

Web Resources

1. https://nptel.ac.in/courses/108/102/108102045/
2. https://www.coursera.org/learn/embedded-software-hardware.
3. https://www.edx.org/course/embedded-systems-shape-the-world-multi-threaded-in.

Course Contents and Lecture Schedule

Topic	**No. of lectures**
Module 1	**9 hrs.**
Introduction to Embedded Systems: Embedded system, Embedded system Vs General Computing System, Processor Embedded into a System.	2
Embedded Hardware units and devices in a system, Embedded Software in a System, Introduction to embedded system design, classification of Embedded	

systems, Skills Required for an embedded system Designer, Examples of the Embedded Systems.	5
Major Application Areas of Embedded Systems, Purpose of Embedded Systems.	2
Module 2	**10 hrs**
Embedded System Design and development Process: Embedded System-On-Chip (SoC) and Use of VLSI Circuit Design Technology, Build Process for embedded systems.	4
Design Process in Embedded System, Design Challenges in Embedded System Design.	3
Hardware-Software Co-Design in an Embedded System, Formalism of System Design.	3
Module 3	**9 hrs**
Hardware Software Co-Design and Program Modelling: – Fundamental Issues in Hardware Software Co-Design.	2
Computational Models in Embedded Design - Data Flow Graph Model, Control Data Flow Graph, State Machine Model, Sequential Program Model, Concurrent / communicating Process Model, Object oriented model, UML.	7
Module 4	**10 hrs**
Design and Development of Embedded Product: **Embedded Hardware Design and Development:** - Analog Electronic Components, Digital Electronic Components, VLSI and Integrated Circuit Design.	5
Embedded Firmware Design and Development: - Embedded Firmware Design Approaches, Embedded Firmware Development Languages.	5
Module 5	**10 hrs**
Integration and Testing of Embedded Hardware and Firmware: - Integration of Hardware and Firmware, Testing Embedded Systems.	6
RTOS based Embedded System Design: - Basic operating system services, Introduction to Real Time Operating System(RTOS), RTOS Task-Scheduling models, How to Choose an RTOS.	4

20MCA186	COMPUTER GRAPHICS	CATEGORY	L	T	P	CREDIT
		ELECTIVE	3	1	0	4

Preamble: This subject intends to provide an overview of the foundations of Computer Graphics rendering. Special emphasis is laid on modern concepts like Ray Tracing that have already become industry standards for graphics rendering with modern GPUs. Other fundamentals such as colorimetry and radiometry are also introduced in the subject. Although the course is expected to be treated theoretically for evaluation purposes, practical sessions and talks by external experts from the graphics processing industry may be desirable.

Prerequisite:

Fundamentals of computer hardware, Linear Algebra

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

CO 1	Apply foundational knowledge in computer graphics to work with Graphics APIs
CO 2	Explain various shape drawing algorithms and transformations.
CO 3	Explain viewing concepts and follow the workflow in computer graphics pipeline.
CO 4	Explain different shading, texture mapping and data structures used in computer graphics.
CO 5	Apply concepts in Raytracing to better understand and design computer graphics models.
CO 6	Apply concepts in colorimetry and radiometry to work with images.

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	3											
CO 2	3	2	2		3							
CO 3	3											
CO 4	3											
CO 5	3				1		1					
CO 6	3											

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	10	10	10
Understand	20	20	20
Apply	20	20	30
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 8 marks
Continuous Assessment Test (2 numbers) : 20 marks
Assignment/Quiz/Course project : 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question caries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks

Course Level Assessment Questions

Course Outcome 1 (CO1):

4. Explain the workflow in the computer graphics pipeline.
5. How does alpha composition affect image appearance?
6. Explain Eigen vectors and Eigen values?

Course Outcome 2 (CO2)

4. Explain some of the pitfalls of Bresenham line drawing algorithm.
5. Explain Affine transformation and its purpose.
6. Compare the mid-point and Bresenham circle drawing algorithms.

Course Outcome 3(CO3):

4. Explain how anti-aliasing affects the image quality.
5. Explain projective transformation.
6. Explain Field of View.

Course Outcome 4 (CO4):

4. Explain some of the benefits of using Triangle meshes.
5. How is Phong shading different from Artistic Shading?
6. How is texture mapping for rasterized image performed.

Course Outcome 5 (CO5):

4. Explain what makes Ray Tracing a highly system intensive rendering process.
5. How can transparency be achieved using ray tracing.
6. Explain the techniques used in Ray Tracing for shadows.

Course Outcome 6 (CO6):

1. Explain Tonal Reproduction.
2. Write a short note on particle tracing for Lambertian scenes.
3. How can rough and smooth surfaces be modeled?

Model Question paper

Part A

21. Explain the concept and idea behind pixels.
22. Write three major applications of computer graphics.
23. Explain the term antialiasing?
24. Demonstrate how the DDA Line drawing algorithm works with a simple example of your own.
25. What is the purpose of tiling multidimensional arrays.
26. Explain the meaning and purpose of graphics APIs?
27. Explain the concept of ray tracing?
28. Explain the basic concept behind ray-object intersection and how it is established.
29. Explain what is meant by color space and give examples.
30. What is meant by a High Dynamic Range image? [3x10 =30 Marks]

Part B

Module 1

31. Explain the process of converting an analog image to a digital image. [6 Marks]

OR

32. Write a note on the following
    - a. RGB Color Space [3 marks]
    - b. Matrix Diagonalisation [3 marks]

Module 2

33. Write short notes on
    - c. Projective Transformation [3 Marks]
    - d. Perspective Projection [3 Marks]

OR

34. Explain a typical Graphics Pipeline. [6 Marks]

Module 3

35. Compare the various graphics APIs citing their advantages and disadvantages.

OR [6 Marks]

36. Explain the various specialized data structures used in computer graphics

[6 Marks]

Module 4

37. Explain how shading and shadowing are achieved using Ray Tracing.

OR [6 Marks]

38. Explain the geometry for graphics hardware.

[6 Marks]

Module 5

39. Explain the methods for Nighttone mapping.

[6 Marks]

OR

40. Explain how accurate direct lighting can be achieved.

[6 Marks]

Syllabus

Module	Contents	Hours
I	**Introduction to computer graphics**: Major Areas and Major Applications, Preliminary discussion on Graphics Pipeline, Numerical Issues, Efficiency and Coding Graphics Programs. **Raster Images**: Raster Devices, Images, Pixels, RGB Color and Alpha Composition. **Fundamentals of Signal Processing** for Images and Sampling Theory(Theoretical understanding only). **Mathematical Foundations of Computer Graphics**: Review of Trigonometry and Geometry, Theoretical foundations of Linear Algebra – Vectors and Matrices, Eigen Values and Eigen Vectors, Matrix Diagonalization(Theoretical understanding only).	9
II	**Fundamentals of shape drawing**:- Line drawing - DDA and Bresenham Algorithms, Circle drawing: Mid Point and Bresenham. **Transformations(2D, 3D)**:, Translation and Affine Transformations, Inverse of Transformation Matrices, Coordinate transformations. **Viewing**: Viewing Transformations, Projective Transformation, Perspective Projection, Field of View **Graphics Pipeline**: Rasterization, Operations, Antialiasing, Culling primitives for efficiency.	11
III	**Surface shading**: Diffuse Shading, Phong Shading, Artistic Shading. **Texture Mapping**: 2D and 3D Mapping, Texture Mapping for Rasterized Triangles, Bump Textures, Displacement Mapping, Shadow Maps. **Data Structures for Graphics**: Triangle Meshes, Scene Graphs, Spatial Data Structures, BSP Tree for visibility, Tiling Multidimensional Arrays. **Graphics APIs**: Intuitive understanding of role of Graphics APIs such as OpenGL, Direct3D(DirectX), Vulkan etc. – No programming required	10
IV	**Ray Tracing**: Basic Ray Tracing Algorithms, Perspective, Computing Viewing Rays, Ray-Object Intersection, Shading, Shadows, Ideal Specular Reflection, Ray Tracing Program, Transparency and Refraction, Instancing, Constructive Solid Geometry, Distribution Ray Tracing. **Using Graphics Hardware**: Introduction, Geometry for Hardware, Processing Geometry using Pixels.	8

Module	Contents	Hours
V	**Light**: Radiometry, Transport Equation, Photometry; **Colors**: Colorimetry, Color Spaces, Chromatic Adaption, Color Appearance; **Tonal Reproduction**: Classification, Dynamic Range, Image Formation, Frequency based Operators, Gradient Domain Operators, Gradient Domain Operators, Spatial Operators, Division, Sigmoids, Night Tonemapping. **Global Illumination**: Particle tracing for Lambertian scenes, Path Tracing, Accurate Direct Lighting. Reflection Models: Real World Materials, Implementing Reflection Models, Specular Reflection models, Smooth layered Model, Rough Layered Model.	11

Textbooks:

1. Peter Shirley, Steve Marschner: ***"Fundamentals of Computer Graphics"***, 4th Edtn. AK Peters, 2015. – All Modules.
2. Donald Hearn and M. Pauline Baker, ***"Computer Graphics"***, 2nd Edtn. PHI, 1996. – Module 2(Fundamentals of Shape Drawing).

Reference Books:

1. Matt Pharr and Greg Humphreys, ***"Physically Based Rendering: From Theory to Implementation"***, 2nd Edtn, Morgan Kaufmann,2010;
2. Gilbert Strang, ***"Introduction to Linear Algebra"***, 4th Edtn, Wellesley-Cambridge Press, 2009
3. William Stallings, ***"Data and Computer Communications"***, 10th Edtn, Pearson, 2013.
4. Vulkan Documentation, https://www.khronos.org/vulkan/
5. OpenGL Documentation, https://www.khronos.org/opengl/
6. Nvidia Developer, ***"Nvidia Ray Tracing Documentation"***, Nvidia Documentation, https://raytracing-docs.nvidia.com/. – Module 3 and 4, Topics on Ray Tracing.

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Introduction**	
1.1	**Introduction to computer graphics**: Major Areas and Major Applications, Preliminary discussion on Graphics Pipeline, Numerical Issues, Efficiency and Coding Graphics Programs.	2

No	Topic	No. of Lectures
1.2	**Raster Images**: Raster Devices, Images, Pixels, RGB Color and Alpha Composition.	3
1.3	**Fundamentals of Signal Processing** for Images and Sampling Theory(Theoretical understanding only). **Mathematical Foundations of Computer Graphics**: Review of Trigonometry and Geometry, Theoretical foundations of Linear Algebra – Vectors and Matrices, Eigen Values and Eigen Vectors, Matrix Diagonalization(Theoretical understanding only).	4
2	**Shape Drawing, transformations and Viewing**	
2.1	**Fundamentals of shape drawing**:- Line drawing - DDA and Bresenham Algorithms, Circle drawing: Mid Point and Bresenham.	2
2.2	**Transformations(2D, 3D)**:, Translation and Affine Transformations, Inverse of Transformation Matrices, Coordinate transformations.	2
2.3	**Viewing**: Viewing Transformations, Projective Transformation, Perspective Projection, Field of View	3
2.4	**Graphics Pipeline**: Rasterization, Operations, Antialiasing, Culling primitives for efficiency.	4
3	**Shading**	
3.1	**Surface shading**: Diffuse Shading, Phong Shading, Artistic Shading.	3
3.2	**Texture Mapping**: 2D and 3D Mapping, Texture Mapping for Rasterized Triangles, Bump Textures, Displacement Mapping, Shadow Maps.	3
3.3	**Data Structures for Graphics**: Triangle Meshes, Scene Graphs, Spatial Data Structures, BSP Tree for visibility, Tiling Multidimensional Arrays.	4

No	Topic	No. of Lectures
4	**Ray Tracing, Graphics Hardware and APIs**	
4.1	**Ray Tracing**: Basic Ray Tracing Algorithms, Perspective, Computing Viewing Rays, Ray-Object Intersection, Shading, Shadows, Ideal Specular Reflection, Ray Tracing Program, Transparency and Refraction, Instancing, Constructive Solid Geometry, Distribution Ray Tracing.	3
4.2	**Using Graphics Hardware**: Introduction, Geometry for Hardware, Processing Geometry using Pixels.	3
4.3	**Graphics APIs**: Intuitive understanding of role of Graphics APIs such as OpenGL, Direct3D(DirectX), Vulkan etc. – No programming required	2
5	**Radiometry, colorimetry and tones**	
5.1	**Light**: Radiometry, Transport Equation, Photometry; **Global Illumination**: Particle tracing for Lambertian scenes, Path Tracing, Accurate Direct Lighting Reflection Models: Real World Materials, Implementing Reflection Models, Specular Reflection models, Smooth layered Model, Rough Layered Model.	5
5.2	**Colors**: Colorimetry, Color Spaces, Chromatic Adaption, Color Appearance;	3
5.3	**Tonal Reproduction**: Classification, Dynamic Range, Image Formation, Frequency based Operators, Gradient Domain Operators, Gradient Domain Operators, Spatial Operators, Division, Sigmoids, Night Tonemapping.	3

20MCA188	**ARTIFICIAL INTELLIGENCE**	**CATEGORY**	**L**	**T**	**P**	**CREDIT**
		ELECTIVE	**3**	**1**	**0**	**4**

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
Estd. 2014

Preamble: This course introduces the techniques of Artificial Intelligence and analyzes various methods of solving problems using it. The concept of expert system architecture & fuzzy operations are introduced. This course serves as a prerequisite for many advanced courses in Data Science areas.

Prerequisite: Mathematical Foundations for Computing, Advanced Data structures

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

CO 1	Apply the steps needed to provide a formal specification for solving the problem.
CO 2	Apply and analyze the different types of control and heuristic search methods to solve problems
CO 3	Understand various Game theory problems& Knowledge structures
CO 4	Formulate knowledge representation and examine resolution in predicate and propositional logic
CO 5	Apply feasible planning and learning techniques to solve non-trial problems
CO 6	Analyze expert systems & fuzzy operations to solve real life problems.

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	3	3							2			
CO 2	3	3							2			
CO 3	3	3							2			
CO 4	3	3							2			
CO 5	3	3							2			
CO 6	3	3	3				3		2		2	2

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember(K1)	10	10	10
Understand(K2)	20	20	20
Apply(K3)	20	20	30
Analyse(K4)			
Evaluate(K5)			
Create(K6)			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance	: 8 marks
Continuous Assessment Test (2 numbers)	: 20 marks
Assignment/Quiz/Course project	: 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question caries 3 marks. Part B contains 2 questions from each module of which student should answer any one.Each question can have maximum 2 sub-divisions and carry 6 marks

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Describe the areas of Artificial intelligence. (K1)

2. List the problem formulations & production characteristics. (K1 & K2)

3. Solve the various problems such as 8 puzzle, Crypt arithmeticetc(K3)

Course Outcome 2 (CO2):

1. Describe search strategies in solving problems. (K1 & K2)

2. List the disadvantages of hill climbing algorithm (K1& K2)

3. Illustrate A* algorithm for the graph (K3)

Course Outcome 3 (CO3):

1. Demonstrate two player Zero sum game (K3)

2. List and explain the knowledge representation methods in AI. (K1&K2)

3. Explain how alpha-beta algorithm works in pruning of branches with an example.(K3)

Course Outcome 4 (CO4):

1. Translate the following sentence to predicate logic (K3)

a) 'All pompeians were Roman'

b)'All Romans were either loyal to Caesar or hated him'.

2. Explain the algorithm to convert WFF to clause.(K1 & K2)

3.Describe about resolution graph in predicate and propositional logic.(K1 & K2)

Course Outcome 5 (CO5):

1. Differentiate between Goal stack and Hierarchical planning in AI. (K1 & K2)
2. Discuss about neural net learning(K1 & K2)

3. List out the steps in genetic learning. (K1 & K2)

Course Outcome 6 (CO6):

1. Specify the components in expert system. (K1 & K2)

2. Solve various fuzzy operations (K3)

3. List out & explain various tools and languages in AI. (K1 & K2)

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY KTU Estd 2014

Model question paper

Part A

1. List the applications areas in AI
2. Solve the following cryptarithmetic problem

    SEND +
    MORE
    MONEY

3. Explain iterative deepening search
4. List the disadvantages of hill climbing
5. Solve a simple two player Zero sum game
6. Explain about conceptual dependency
7. Explain inference rules in FOPL
8. List components of a planning system
9. Give a short note on role of an expert system
10. List various fuzzy operations

(10X3=30 marks)

Part B

11. Consider a water jug problem .You are given two jugs, a 4 gallon and 3 gallons. Neither has any measuring markers on it. There is a pump that can be used to fill the jugs with water. How can you get exactly 2 gallons of water into 4-gallon jug.State the production rule for waterjug problem (6)

OR

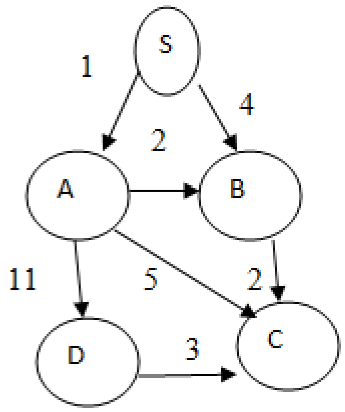
12. Solve missionaries and cannibals problem (6)

13. Explain blind search strategies in detail (6)

OR

14. Explain A* Algorithm for the given graph (6)

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
KTU
Estd.
2014



Heuristic Value:

S	6
A	2
B	3
C	0
D	5

15. List and explain the knowledge representation methods in AI. (6)

OR

16. Explain how alpha-beta algorithm works in pruning of branches with an example. (6)

17. Explain the algorithm to convert WFF to clause with an example. (6)

OR

18. Explain Neural net and Genetic learning methods in AI (6)

19. Illustrate architecture of an expert system and mention its features. (6)

OR

20. Solve the following using various Fuzzy set operations (6)

A= {0.1/1,0.3/2,0.45/3}

B={0.15/1,0.34/2}

(5X6=30 Marks)

SYLLABUS

Module 1

Introduction to AI and Production Systems:- AI-Problem formulation, Problem Definition - Production systems, Problem characteristics, Production system characteristics , Example AI Problems (8 Puzzle problem, Missionary Cannibals Problem, Crypt arithmetic Problems, block world Problem)

Module 2

Search Strategies : - Blind search strategies -Depth First Search, Breadth First Search, Best First Search, Iterative Deepening Search, Heuristic Search strategies- Admissible Heuristics and examples - Simple Hill Climbing and Steepest Ascending Hill Climbing, Simulated Annealing , A* algorithm.

Module 3

Game playing : Two Player Zero Sum Games, Modelling Two Player Zero Sum Games as search problems, Min-Max Algorithm, Optimizing Min Max Algorithm using α – β cut off, *Knowledge Representation Structures :* Frames, Sematic Networks and Conceptual Dependencies.

Module 4

Knowledge representation using Logic : - First Order Predicate Logic (FOPL), Well Formed Formula(WFF) in FOPL, Inference rules for FOPL, The Clause Form and conversion of WFFs to Clause Form, Resolution- Refutation .*Planning :*- Overview, components of a planning system, Goal stack planning, Hierarchical planning, *Learning :*-Forms of learning, neural net learning & genetic learning

Module 5

Expert systems:–Architecture of expert systems, Roles of expertsystems,Languages and tools – Typical expert system examples.*Fuzzy Logic:* - Fuzzy Variables ,Fuzzy Sets and Fuzzy Set Operations, Typical Examples using FuzzySets.

Text Books

1. Kevin Night and Elaine Rich, "***Artificial Intelligence (SIE)***", McGrawHill-2008.
2. StuartRussel and Peter Norvig "***AI – A Modern Approach***", 2nd Edition, Pearson Education2007.

Reference Books

1.Peter Jackson, “***Introduction to Expert Systems***”, 3rd Edition, Pearson Education,2007.
2.Dan W. Patterson, “***Introduction to AI and ES***”, Pearson Education,2007.

Course Contents and Lecture Schedule

No	**Topic**	**No. of Lectures**
1	***Module I: Introduction to AI***	***9 hrs***
1.1	AI-Problem formulation, Problem Definition -Production systems	
1.2	Production system characteristics	
1.3	AI Problems	
2	***Module II: Search Strategies***	***9 hrs***
2.1	Blind search strategies	
2.2	Heuristics search strategies	
2.3	Simple Hill Climbing and Steepest Ascending Hill Climbing,	
2.4	Simulated anneling	
2.5	A* algorithm	
3	***Module III: Game playing***	***9 hrs***
3.1	Zero sum game	
3.2	Minimax algorithm	
3.3	Alpha beta pruning	
3.4	Knowledge representation structure	
4	***Module IV: Knowledge representation using Logic***	***12 hrs***
4.1	First Order Predicate Logic (FOPL)	
4.2	Well Formed Formula(WFF) in FOPL, Inference rules for FOPL	
4.3	The Clause Form and conversion of WFFs to Clause Form	
4.4	Resolution	
4.5	Planning	
4.6	Learning	
5	***Module V: APPLICATIONS***	***6 hrs***
5.1	Expert system Architecture	
5.2	Fuzzy logic operations	
5.3	Languages and tools	

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
Estd
2014

20MCA192	IPR AND CYBER LAWS	CATEGORY	L	T	P	CREDIT
		ELECTIVE	3	1	0	4

Preamble: This course intends to provide insight into Intellectual Property Rights and Cyber Laws. It includes detailed discussion on various intellectual property rights, procedures to apply for copyrights &patents, legalities of intellectual property to avoid plagiarism and other IPR related crimes. Effectiveness of cyber-laws and other countermeasures against cybercrime and cyber warfare are discussed in detail. Various kinds of Intellectual Property issues in cyberspace and the growth and development of the law in this regard are included to the level possible within the scope of a single course. More detailed treatment can be done through seminars, assignments and talks by eminent external experts including industry.

Prerequisite: General awareness on internet essentials, web technologies, e-commerce.

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

CO 1	Explain the fundamentals of IPR and patents.
CO 2	Apply intellectual property related tools such as trademark and copyright to real problems.
CO 3	Discuss Industrial designs, trade secret and geographic Indications.
CO 4	Describe laws governing cyberspace and analyze the role of Internet Governance in framing policies for Internet security.
CO 5	Discuss different types of cybercrimes and penalties under IT Act.

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	3	2	2	1		1						
CO 2	3	3	2	1		1						
CO 3	3	2	1	1								
CO 4	2	2	1			1						
CO 5	2	2	1	1		1						

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	10	10	10
Understand	20	30	25
Apply	20	10	25
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	40	60	3 hours

Continuous Internal Evaluation Pattern:

Attendance	: 8 marks
Continuous Assessment Test (2 numbers)	: 20 marks
Assignment/Quiz/Course project	: 12 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 compulsory short answer questions, 2 from each module. Each question carries 3 marks. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 6 marks

Course Level Assessment Questions

Course Outcome 1 (CO1):

4. Discuss the need for protection of intellectual property.
5. Explain TRIPS Agreement.
6. Illustrate types of patent applications.

Course Outcome 2 (CO2)

4. Explain Trademark Infringement and Protection of trademarks.
5. Explain the rights conferred by copyright, registration and ownerships of copyrights.
6. Discuss about software copyright.

Course Outcome 3(CO3):

4. Discuss the need for protection of design and explain Design Act, 2000.
5. Explain basic concepts of Geographic Indications such as filing, granting and Protection of geographic indications.
6. Describe the procedure of discovering and protecting of trade secret.

Course Outcome 4 (CO4):

4. Explain the need for cyber laws.
5. Discuss protection of copyright on cyberspace.
6. Explain ISP in cyberspace.

Course Outcome 5 (CO5):

4. Explain different amendments on IT Act 2000.
5. Discuss Terrorism on cyberspace.
6. Explain offences of misrepresentation.

Model Question paper

Part A

21. Categorize various patent applications.
22. Explain the criteria for categorizing an invention as patentable or non-patentable.
23. What are the requirements for filing trademarks?
24. Explain copyright and the rights conferred by copyrights.
25. Explain the term geographical indications by giving suitable examples.
26. What is meant by design under the Design Act,2000?
27. Describe the risks associated with cyber space.
28. What is meant by the term cyber laws.
29. Explain cyber stalking and phishing.
30. Define the term hacking and explain its essentials. [3 x 10 =30 Marks]

Part B

Module 1

31. Describe the procedure for registration of patents. [6 Marks]

OR

32. Write short notes on
    - c. Intellectual property and the need for its protection. [3 marks]
    - d. Importance and features of WIPO. [3 marks]

Module 2

33. Explain the methods for transferring copyrights. [6 Marks]

OR

34. Describe software copyright and how can software be classified according to copyrights. [6 Marks]

Module 3

35. What is industrial design? Describe the salient features of Design act, 2000. [6 Marks]

OR

36. How are the trade secrets dealt with under the Indian law? Discuss.

[6 Marks]

Module 4

37. Explain the essential requirements of cyber squatting.

[6 Marks]

OR

38. Discuss about cyber space and the protection of copyrights on cyberspace.

[6 Marks]

Module 5

39. Explain the objectives and features of Information Technology Act 2000.

[6 Marks]

OR

40. What do you mean by cyber crimes? Discuss the nature and types of cyber crimes.

[6 Marks]

Syllabus

Module	**Contents**	**Hours**
I	Fundamentals of IPR- Introduction – Intellectual property – Need for protection of intellectual property – WIPO – Intellectual property rights and development – Rationale of protection – TRIPS Agreement - **Patents : –** Introduction **–** Patentable and Non-patentable Invention – Types of patent applications – Guidelines for registration of patent – patent filing – grant of patent – types of patent documents.	10

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY KTU Estd. 2014

Module	Contents	Hours
II	**Trademarks**– Introduction – Guidelines for registration- Requirements for filing trademarks – Trademark Infringement – Protection of trademarks – **Copyright –** Introduction – Rights conferred by copyright – registration – ownerships – terms – transfer of copyrights – copyright infringement – databases and copyright- **Software Copyright**–Introduction – Need of software copyright – classification of software according to copyright – software auditing –copyright notice – transfer of copyright.	10
III	**Industrial Designs**– Introduction – Need for protection of design – requirements for registration of designs – Design Act,2000 – Duration of registration of design – application procedure – **GeographicIndications –**Introduction – Filing – **Granting** – Protection of geographic indications**.** **Trade Secret –** definition – discovering and protecting of trade secret.	10
IV	**Cyber law** - Need for cyber laws - Historical perspective - cyberspace - deception by squatting in cyberspace - protection of copyright on cyberspace - infringement of copyright on cyberspace - linking,hyperlinking and framing - ISP in cyberspace - cyberspace and protection of patents in India.	8
V	**Information Technology Act and Punishments-** Introduction to IT Act 2000- Amendments on IT Act - Violation of the right of privacy in cyberspace/internet-punishment for violation of privacy, breach of confidentiality and privacy under IT act-Terrorism on cyberspace Overview of cybercrimes-offences by intermediaries- offences related to protected system- offences of misrepresentation-punishment for Abetment and Attempt to commit offences under the IT act.	10

Textbooks:

1.Dr. R. Radhakrishnan and Dr. S. Balasubramanian, "**Intellectual Property Rights: Text and Cases**", Excel Books
2.Harish Chander, "**Cyber Law and IT Protection**", PHI Learning Pvt.Ltd.

Reference Books:

5. D.Bainbridge, “**Introduction to Computer Law**”, Pearson Education
6. RohasNagpal, “**Cyber Crime & Corporate Liability**”, CCH, 2008
7. https://www.udemy.com/course/cyber-security-law/
8. https://www.coursera.org/specializations/introduction-intellectual-property

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Introduction**	
1.1	**Fundamentals of IPR-** Introduction – Intellectual property – Need for protection of intellectual property	2
1.2	WIPO – Intellectual property rights and development – Rationale of protection – TRIPS Agreement	3
1.3	**Patents :** – Introduction – Patentable and Non-patentable Invention – Types of patent applications – Guidelines for registration of patent – patent filing – grant of patent – types of patent documents	5
2		
2.1	**Trademarks**– Introduction – Guidelines for registration – Requirements for filing trademarks – Trademark Infringement – Protection of trademarks	3
2.2	**Copyright –** Introduction – Rights conferred by copyright – registration – ownerships – terms – transfer of copyrights – copyright infringement – databases and copyright	3
2.3	**Software Copyright**– Introduction – Need of software copyright – classification of software according to copyright – software auditing – copyright notice – transfer of copyright.	4
3		
3.1	**Industrial Designs**– Introduction – Need for protection of design – requirements for registration of designs – Design Act,2000 – Duration of registration of design – application procedure	4

No.	**Topic**	**No. of Lectures**
3.2	**Geographic Indications** – Introduction – Filing **Granting** – Protection of geographic indications**.**	4
3.3	**Trade Secret** – definition – discovering and protecting of trade secret.	2
4		
4.1	**Cyber law** - Need for cyber laws - Historical perspective - cyberspace - deception by squatting in cyberspace.	3
4.2	Protection of copyright on cyberspace - infringement of copyright on cyberspace - linking, hyper linking and framing -	3
4.3	ISP in cyberspace - cyberspace and protection of patents in India.	2
5		
5.1	**Information Technology Act and Punishments**- Introduction to IT Act2000- Amendments on IT Act	2
5.2	Violation of the right of privacy in cyberspace/internet-punishment for violation of privacy, breach of confidentiality and privacy under IT act-Terrorism on cyberspace overview of cybercrimes	4
5.3	Offences by intermediaries- offences related to protected system- offences of misrepresentation-punishment for Abetment and Attempt to commit offences under the IT act.	4

20MCA132	**OBJECT ORIENTED PROGRAMMING LAB**	**CATEGORY**	**L**	**T**	**P**	**CREDIT**
		PRACTICAL	**0**	**1**	**3**	**2**

Preamble: This course enables the students to understand the concepts of object-oriented programming and to develop skills using these paradigms using Java.

Prerequisite: Knowledge of any programming language preferred.

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

CO 1	Understand object-oriented concepts and design classes and objects to solve problems
CO 2	Implement arrays and strings.
CO 3	Implement object-oriented concepts like inheritance, overloading and interfaces
CO 4	Implement packages, exception handling, multithreading and generic programming. Use java.util package and Collection framework
CO 5	Develop applications to handle events using applets
CO 6	Develop applications using files and networking concepts

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	2	2	2	2	3							
CO 2	**3**	2	2		3							
CO 3	**3**	2	2		3							
CO 4	**3**	2	2		3							
CO 5	**3**	3	3		3	2			3		3	
CO 6	**3**	3	3		3	2			3		3	

Assessment Pattern

Bloom's Category	**Continuous Assessment Tests**		**End Semester Examination**
	1	**2**	
Remember(K1)			
Understand(K2)			
Apply(K3)	10	10	10
Analyse(K4)	10	10	10
Evaluate(K5)	10	10	10
Create(K6)	20	20	20

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 hours

Continuous Internal Evaluation Pattern:

Maximum Marks: 50	
Attendance	15%
Maintenance of daily lab record and GitHub management	20%
Regular class viva	15%
Timely completion of day to day tasks	20%
Tests/Evaluation	30%

End Semester Examination Pattern:

Maximum Marks: 50			
Verification of Daily program record and Git Repository			5 marks
Viva			10 marks
Problem solving (Based on difficulty level, one or more questions may be given)	Flowchart / Algorithm / Structured description of problem to explain how the problem can be solved / Interface Design	15%	35 marks
	Program correctness	50%	
	Code efficiency	15%	
	Formatted output and Pushing to remote Git repository	20%	
Total Marks			50 marks

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Define a class ‘product’ with data members pcode, pname and price. Create 3 objects of the class and find the product having the lowest price.
2. Read 2 matrices from the console and perform matrix addition.

3. Add complex numbers
4. Read a matrix from the console and check whether it is symmetric or not.
5. Create CPU with attribute price. Create inner class Processor (no. of cores, manufacturer) and static nested class RAM (memory, manufacturer). Create an object of CPU and print information of Processor and RAM.

Course Outcome 2 (CO2)

1. Program to Sort strings
2. Search an element in an array.
3. Perform string manipulations
4. Program to create a class for Employee having attributes eNo, eName eSalary. Read n employ information and Search for an employee given eNo, using the concept of Array of Objects.

Course Outcome 3(CO3):

1. Area of different shapes using overloaded functions
2. Create a class ‘Employee’ with data members Empid, Name, Salary, Address and constructors to initialize the data members. Create another class ‘Teacher’ that inherit the properties of class employee and contain its own data members department, Subjects taught and constructors to initialize these data members and also include display function to display all the data members. Use array of objects to display details of N teachers.
3. Create a class ‘Person’ with data members Name, Gender, Address, Age and a constructor to initialize the data members and another class ‘Employee’ that inherits the properties of class Person and also contains its own data members like Empid, Company_name, Qualification, Salary and its own constructor. Create another class ‘Teacher’ that inherits the properties of class Employee and contains its own data members like Subject, Department, Teacherid and also contain constructors and methods to display the data members. Use array of objects to display details of N teachers.
4. Write a program has class Publisher, Book, Literature and Fiction. Read the information and print the details of books from either the category, using inheritance.
5. Create classes Student and Sports. Create another class Result inherited from Student and Sports. Display the academic and sports score of a student.

6. Create an interface having prototypes of functions area() and perimeter(). Create two classes Circle and Rectangle which implements the above interface. Create a menu driven program to find area and perimeter of objects.

7. Prepare bill with the given format using calculate method from interface.

Order No.

Date :

Product Id	Name	Quantity	unit price	Total
101	A	2	25	50
102	B	1	100	100
			Net. Amount	150

Course Outcome 4 (CO4):

1. Create a Graphics package that has classes and interfaces for figures Rectangle, Triangle, Square and Circle. Test the package by finding the area of these figures.
2. Create an Arithmetic package that has classes and interfaces for the 4 basic arithmetic operations. Test the package by implementing all operations on two given numbers
3. Write a user defined exception class to authenticate the user name and password.
4. Find the average of N positive integers, raising a user defined exception for each negative input.
5. Define 2 classes; one for generating multiplication table of 5 and other for displaying first N prime numbers. Implement using threads. (Thread class)
6. Define 2 classes; one for generating Fibonacci numbers and other for displaying even numbers in a given range. Implement using threads. (Runnable Interface)
7. Producer/Consumer using ITC
8. Program to create a generic stack and do the Push and Pop operations.
9. Using generic method perform Bubble sort.
10. Maintain a list of Strings using ArrayList from collection framework, perform built-in operations.
11. Program to remove all the elements from a linked list
12. Program to remove an object from the Stack when the position is passed as parameter
13. Program to demonstrate the creation of queue object using the PriorityQueue class
14. Program to demonstrate the addition and deletion of elements in deque
15. Program to demonstrate the creation of Set object using the LinkedHashset class
16. Write a Java program to compare two hash set

17. Program to demonstrate the working of Map interface by adding, changing and removing elements.
18. Program to Convert HashMap to TreeMap

Course Outcome 5 (CO5):

1. Program to draw Circle, Rectangle, Line in Applet.
2. Program to find maximum of three numbers using AWT.
3. Find the percentage of marks obtained by a student in 5 subjects. Display a happy face if he secures above 50% or a sad face if otherwise.
4. Using 2D graphics commands in an Applet, construct a house. On mouse click event, change the color of the door from blue to red.
5. Implement a simple calculator using AWT components.
6. Develop a program that has a Choice component which contains the names of shapes such as rectangle, triangle, square and circle. Draw the corresponding shapes for given parameters as per user's choice.
7. Develop a program to handle all mouse events and window events
8. Develop a program to handle Key events.

Course Outcome 6 (CO6):

1. Program to list the sub directories and files in a given directory and also search for a file name.
2. Write a program to write to a file, then read from the file and display the contents on the console.
3. Write a program to copy one file to another.
4. Write a program that reads from a file having integers. Copy even numbers and odd numbers to separate files.
5. Client server communication using Socket – TCP/IP
6. Client Server communication using DatagramSocket - UDP

Syllabus:

Classes and Objects, Constructors, Method Overloading, Access Modifiers, Arrays and Strings, Inheritance, Interfaces, Abstract classes, Dynamic Method Dispatch, String, Packages, Introduction to java.util, Collection framework, User defined packages, Exceptions, Multithreading, Applets, Graphics, File, Generic programming, Socket Programming

Reference Books

1. Herbert Schildt, "***Java The Complete Reference***", Seventh Edition, Tata McGraw-Hill Edition

2. C. Thomas Wu, "***An introduction to Object-oriented programming with Java***", Fourth Edition, Tata McGraw-Hill Publishing company Ltd.

3. Cay S. Horstmann and Gary Cornell, "***Core Java: Volume I – Fundamentals***", Eighth Edition, Sun Microsystems Press.

4. K. Arnold and J. Gosling, "***The JAVA programming language***", Third edition, Pearson Education.

5. Paul Deitel and Harvey Deitel, "***Java, How to Program***", Tenth Edition, Pearson Education

6. Rohit Khurana, "***Programming with Java***", Vikas Publishing, 2014.

7. Timothy Budd, "***Understanding Object-oriented programming with Java***", Updated Edition, Pearson Education.

8. Y. Daniel Liang, "***Introduction to Java programming***", Seventh Edition, Pearson Education.

Web Reference

- https://www.hackerrank.com/domains/java
- https://www.geeksforgeeks.org/java-tutorial/
- https://www.w3resource.com/java-tutorial/
- https://www.w3resource.com/java-exercises/
- https://nptel.ac.in/courses/106/105/106105191/
- https://nptel.ac.in/noc/courses/noc20/SEM1/noc20-cs08/
- https://www.coursera.org/learn/object-oriented-java
- https://www.edx.org/course/object-oriented-programming-in-java-2

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
KTU
Estd
2014

Course Contents and Lab Schedule

Topic	No. of hours
1. Classes and Objects.	3
2. Constructors, Method Overloading, Access Modifiers	2
3. Arrays and Strings.	4
4. Inner class – static and non-static	2
5. Inheritance, Multiple inheritance - implementation using interfaces	3
6. Method overriding, Abstract classes, Dynamic Method Dispatch	3
7. Interfaces and Packages, StringBuffer class	3
8. Introduction to java.util package – Vector, Scanner, StringTokenizer	2
9. Collection framework – ArrayList, LinkedList, Stack, Queue, Set, Map	3
10. User defined packages	2
11. Exceptions – User defines exceptions	2
12. Multithreading – Thread class	2
13. Inter Thread Communication	2
14. Generic programming	2
15. Applets, Graphics – 2D	3
16. Event handling in Applet	3
17. File	3
18. Socket Programming	3

20MCA134	ADVANCED DBMS LAB	CATEGORY	L	T	P	CREDIT
		PRACTICAL	0	1	3	2

Preamble: This course is to provide understanding on relational and non-relational database systems and its design. The course covers SQL, PL/SQL and NoSQL programs which are essential for the development and deployment of web based applications. Also this course serves as a prerequisite for many advanced courses in Data Science areas.

Prerequisite: Database Management Systems

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

CO 1	Design and build a simple relational database system and demonstrate competence with the fundamentals tasks involved with modelling, designing and implementing a database.
CO 2	Apply PL/SQL for processing databases.
CO 3	Comparison between relational and non-relational (NoSQL) databases and the configuration of NoSQL Databases.
CO 4	Apply CRUD operations and retrieve data in a NoSQL environment.
CO 5	Understand the basic storage architecture of distributed file systems.
CO 6	Design and deployment of NoSQL databases with real time requirements.

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	3	2	3	2	2					1	1	
CO 2	2	2	2		1							
CO 3	2	2	2	2						1	1	
CO 4	2	2	3	1	2		1			1	1	1
CO 5	3	2	2				1				1	1
CO 6	2	2	3	1	1			1		1	1	2

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
Estd. 2014

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember(K1)			
Understand(K2)			
Apply(K3)	10	10	10
Analyse(K4)	10	10	10
Evaluate(K5)	10	10	10
Create(K6)	20	20	20

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 hours

Continuous Internal Evaluation Pattern:

Maximum Marks: 50	
Attendance	15%
Maintenance of daily lab record and GitHub management	20%
Regular class viva	15%
Timely completion of day to day tasks	20%
Tests/Evaluation	30%

End Semester Examination Pattern:

Maximum Marks: 50			
Verification of Daily program record and Git Repository			5 marks
Viva			10 marks
Problem solving (Based on difficulty level, one or more questions may be given)	Flowchart / Algorithm / Structured description of problem to explain how the problem can be solved / Interface Design	15%	35 marks
	Program correctness	50%	
	Code efficiency	15%	
	Formatted output and Pushing to remote Git repository	20%	
Total Marks			50 marks

Course Level Assessment

Questions Course Outcome 1 (CO1):

1. Creation of a database using DDL commands including integrity constraints. (K6)
2. Create an application to apply Data Manipulation Language (DML) commands to modify the database. (K6)
3. Apply DCL and TCL commands to impose restrictions on databases. (K3)
4. Create an application to retrieve data from databases using select, views. (K6)
5. Create an application to use joins for query optimization. (K6)

Course Outcome 2 (CO2):

1. Construct PL/SQL code for sample databases. (K6)

Course Outcome 3(CO3):

1. Compare relational and non-relational databases. (K5)
2. Understand the installation and configuration of NoSQL Databases. (K2)

Course Outcome 4 (CO4):

1. Build sample collections/documents to perform query operations. (K6)

Course Outcome 5 (CO5):

1. Build sample collections/documents to perform the shell commands like replica set, indexing etc. (K6)

Course Outcome 6 (CO6):

1. Develop sample applications using any of the front end tools and NoSQL. (K6)
2. Usage of concerned Online/Cloud Storage Management Systems like MongoDB Atlas, Cassandra DataStax etc. (K6)
3. Deployment of NoSQL in Cloud: Google Bigtable/ Amazon DynamoDB/ Azure Cosmos DB. (K6)

Syllabus

1. An overview of relational database design using MySQL/ MariaDB/ PostgreSQL etc. (Apply the following basic queries on an Employee/ Student database etc.)
    a. DDL Commands
    b. DML Commands
    c. Imposing restrictions on database (DCL & TCL Commands)
    d. Accessing database (SELECT, Filtering using WHERE, HAVING, GROUP BY, ORDER BY Clauses, Subquery and View)
    e. Optimizing databases (Join, Aggregate & Set operations, Other operators like arithmetic, logical, special etc.)
2. PL/SQL Programs (Trigger, Cursor, Stored Procedures and Functions)
3. Introduction to NoSQL Databases.
    a. Installation and configuration of any one of the NoSQL databases - MongoDB/ Cassandra/ HBase/ CouchDB/ Amazon DynamoDB/ Redis/ Neo4j etc.
4. Designing Databases using NoSQL
5. Query Processing
    a. Performing CRUD operations
    b. Retrieving Data from a NoSQL database
    c. Usage of aggregate functions, regular expressions etc.
6. NoSQL Administration
    a. Security, Monitoring & Backup
    b. Create Users and Roles
7. NoSQL shell commands
    a. Perform Sharding, Replication (Master-Slave/ Master-Less/ Peer-to-Peer Architectures), Clustering, Partitioning, Indexing (Corresponding to the selected NoSQL Database)

8. Deployment
    a. Local Deployment
        i. NoSQL and Front-End: PHP/Java/Python (MongoDB/ Cassandra etc.)
    b. Cloud Deployment
        i. NoSQL and Cloud: Amazon DynamoDB/ Google Bigtable/ Azure Cosmos DB
        ii. Familiarization of Atlas/ DataStax corresponding to the selected NoSQL Database
9. ***Micro project:*** *Students can be given a group micro project, so that they learn to work in a team environment.*

Text Books

1. Abraham Silberschatz, Henry F. Korth, S. Sudarshan," ***Database System Concepts***", McGraw Hill Education, 6th Edition (2011)
2. Guy Harrison, "***Next Generation Databases: NoSQL, NewSQL, and Big Data***", Apress, 1st Edition (14 December 2015)

Reference Books

1. Raghu Ramakrishnan and Johannes Gehrke, "***Database Management Systems***", McGraw Hill, 3rd Edition (2014).
2. HBase: The Definitive Guide. Lars George O'Reilly Media; August 2011, ISBN: 9781449315771
3. Shashank Tiwari. Professional NoSQL. John Wiley and Sons. ISBN: 978-0-470-94224-6.
4. MongoDB Administrator's Guide, Cyrus Dasadia, October 2017, Packet Publishing ISBN: 9781787126480
5. Cassandra: The Definitive Guide Distributed Data at Web Scale, 1st Edition, Eben Hewitt, Jeff Carpenter, O'Reilly Media; November 2010

Web Resources

1. Database Management System https://nptel.ac.in/courses/106/105/106105175/
2. Databases: SQL https://www.edx.org/course/databases-5-sql
3. Introduction to MongoDB https://www.coursera.org/learn/introduction-mongodb
4. Apache Cassandra https://www.edureka.co/cassandra
5. NoSQL systems https://www.coursera.org/learn/nosql-databases
6. https://hbase.apache.org/
7. https://couchdb.apache.org/ https://aws.amazon.com/dynamodb/
8. https://aws.amazon.com/dynamodb/

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	*An overview of relational database design using MySQL/ MariaDB/ PostgreSQL etc. (Apply the following basic queries on an Employee/ Student database etc.)*	*6 hrs*
1.1	• DDL Commands • DML Commands • Imposing restrictions on database (DCL & TCL Commands)	3
1.2	• Accessing database (SELECT, Filtering using WHERE, HAVING, GROUP BY, ORDER BY Clauses, Subquery and View) • Optimizing databases (Join, Aggregate & Set operations, Other operators like arithmetic, logical, special etc.)	3

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
KTU
Estd
2014

No	*Topic*	*No. of Lectures*
2	*PL/SQL Programs*	*4 hrs*
2.1	• Trigger, Cursor, Stored Procedures and Functions	4
3	*Introduction to NoSQL Databases*	*2 hrs*
3.1	• Installation and configuration of any one of the NoSQL databases - MongoDB/ Cassandra/ HBase/ CouchDB/ Amazon DynamoDB/ Redis/ Neo4j etc.	2
4	*Designing Databases using NoSQL*	*2 hrs*
5	*Query Processing*	*8 hrs*
5.1	• Performing CRUD operations • Retrieving Data from a NoSQL database • Usage of aggregate functions, regular expressions etc.	8
6	*NoSQL Administration*	*2 hrs*
6.1	• Security, Monitoring & Backup • Create Users and Roles	2
7	*NoSQL shell commands*	*6 hrs*
7.1	• Perform Sharding, Replication (Master-Slave/ Master-Less/ Peer-to-Peer Architectures), Clustering, Partitioning, Indexing (Corresponding to the selected NoSQL Database)	6
8	*Deployment*	*16 hrs*
8.1	• Local Deployment NoSQL and Front-End: PHP/Java/Python (MongoDB/ Cassandra etc.)	4
8.2	• Cloud Deployment NoSQL and Cloud: Amazon DynamoDB/ Google Bigtable/ Azure Cosmos DB	8
8.3	• Familiarization of Atlas/ DataStax corresponding to the selected NoSQL Database	4
9	*Micro project*	*10 hrs*

20MCA136	**NETWORKING & SYSTEM ADMINISTRATION LAB**	**CATEGORY**	**L**	**T**	**P**	**CREDIT**
		PRACTICAL	**0**	**1**	**3**	**2**

Preamble: This laboratory course is intended to provide the background knowledge required for a software professional in the fields of networking and system administration. Students will acquire necessary knowledge to deploy and administer systems.

Prerequisite: Basic understanding of computer programming, Internet and operating systems

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

CO 1	Install and configure common operating systems.
CO 2	Perform system administration tasks.
CO 3	Install and manage servers for web applications.
CO 4	Write shell scripts required for system administration.
CO 5	Acquire skill sets required for a DevOps.

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	1		2		2							
CO 2	1		2									
CO 3			2		2							
CO 4					2							
CO 5	2				2							

Assessment Pattern

Bloom's Category	**Continuous Assessment Tests**		**End Semester Examination**
	1	**2**	
Remember(K1)			
Understand(K2)			
Apply(K3)	10	10	10
Analyse(K4)	10	10	10
Evaluate(K5)	10	10	10
Create(K6)	20	20	20

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 hours

Continuous Internal Evaluation Pattern:

Maximum Marks: 50	
Attendance	15%
Maintenance of daily lab record and GitHub management	20%
Regular class viva	15%
Timely completion of day to day tasks	20%
Tests/Evaluation	30%

End Semester Examination Pattern:

Maximum Marks: 50			
Verification of Daily program record and Git Repository			5 marks
Viva			10 marks
Problem solving (Based on difficulty level, one or more questions may be given)	Flowchart / Algorithm / Structured description of problem to explain how the problem can be solved / Interface Design	15%	35 marks
	Program correctness	50%	
	Code efficiency	15%	
	Formatted output and Pushing to remote Git repository	20%	
Total Marks			50 marks

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Install latest version of Ubuntu on a virtual box, set up a static ip address to it and install drupal environment.
2. You are given a computer with very low hardware resources. It is to be used as a kiosk. Identify and install a suitable Linux distribution. You can simulate it in a virtual environment.

Course Outcome 2 (CO2)

1. You are given a system which is connected to internet. However, users logging on to the system are unable to access internet from their browser. Trouble shoot the issue,clearly documenting the steps you have taken (Possible issues to look for are browser configuration, network connectivity, routing, ip address configuration,DNS resolution)
2. You are given a system which boots to a non graphical environment. You are also given a shell script which is designed for a specific task. Your task is to make sure that the script runs every time the system boots up. Write/modify necessary scripts for this.
3. You are required to add 100 users to a Linux system. Details of the users to be added were collected from a web form to a csv file. The csv may contain errors such as wrong case or missing fields. Write a script to add users using the data provided in the csv file with proper error checking.

Course Outcome 3(CO3):

1. You are given a bare bone installation of latest version Ubuntu. Assume that the system is accessible from internet. Your task is to successfully install word press (or any other web application) on this server. Clearly indicate the steps taken and software installed for this task.

2. Assume that you have an installation of old version Ubuntu. However, it does not have the latest version of virtual box (or some other application). The new version is available as a binary on a website. Upgrade to this version.

Course Outcome 4 (CO4):

1.Look at the system log files. Write a shell script to extract the last login details of a particular user and list out all failed logins. Store the results to a file. The user name should be given as a command line argument.

2.Write a shell script to display the details of a particular process currently running. Assume that you have necessary permissions. The process name/id is to be given as a command line argument

Course Outcome 5 (CO5):

1.Capture network traffic on your system. Using wireshark find out all http and https traffic to a specific host.

2.Write an Ansible playbook to deploy a new Linux VM on a remote server.

Syllabus:

Introduction to Computer hardware. Study of various peripherals. Study of common operating systems. File system organization in common operating systems.

Study of command line environment in common operating systems. Study of command line tools for system administration.

Shell scripting: bash shell, shell scripts for system management.

Study of startup scripts.

Study of server software for common applications such as http, ftp, dns, dhcp.

Practical study of Ipv4 and Ipv6 networking protocols. Setting up firewalls.

Virtual machines and containers. Configuration and deployment.

List of Lab Experiments/Exercises

To gain proficiency in command line tools and operations, it is highly recommended to use a terminal window instead of GUI tools. This will later help the student with latest approaches in maintaining cloud based infrastructure. virtualbox/quemu. may be used for this.

1. Introduction to Computer hardware: Physical identification of major components of a computer system such as mother board, RAM modules, daughter cards, bus slots, SMPS, internal storage devices,i nterfacing ports. Specifications of desktop and server class computers. Installation of common operating systems for desktop and server use. (Students may be asked to formulate specification for computer to be used as Desktop, Web server)

2. Study of a terminal based text editor such as Vim or Emacs. (By the end of the course, students are expected to acquire following skills in using the editor: cursor operations, manipulate text, search for patterns, global search and replace)
   Basic Linux commands, familiarity with following commands/operations expected
    1. man
    2. ls, echo, read
    3. more, less, cat,
    4. cd, mkdir, pwd, find
    5. mv, cp, rm ,tar
    6. wc, cut, paste
    7. head, tail, grep, expr
    8 chmod, chown
    9. Redirections & Piping
    10. useradd, usermod, userdel, passwd
    11. df,top, ps
    12 ssh, scp, ssh-keygen, ssh-copy-id

   .

3. File system hierarchy in a common Linux distribution, file and device permissions, study of system configuration files in /etc, familiarizing log files for system events, user activity, network events.

4. Shell scripting: study bash syntax, environment variables, variables, control constructs such as if, for and while, aliases and functions, accessing command line arguments passed to shell

scripts. Study of startup scripts, login and logout scripts, familiarity with systemd and system 5 init scripts is expected.

5. Installation and configuration of LAMP stack. Deploy an open source application such as phpmyadmin and Wordpress.

6. Installation and configuration of common software frame works such as Laravel. (Student should acquire the capability to install and configure a modern framework)

7. Build and install software from source code, familiarity with make and cmake utilities expected.

8. Introduction to command line tools for networking
   IPv4 networking, network commands: ping route traceroute, nslookup, ip. Setting up static and dynamic IP addresses. Concept of Subnets, CIDR address schemes, Subnet masks, iptables, setting up a firewall for LAN, Application layer (L7) proxies.

9. Analyzing network packet stream using tcpdump and wireshark. Perform basic network service tests using nc.

10. Introduction to Hypervisors and VMs, Xen or KVM , Introduction to Containers: Docker, installation and deployment.

11. Automation using Ansible: Spin up a new Linux VM using Ansible playbook

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY KTU Estd. 2014