PAAVAI ENGINEERING COLLEGE, NAMAKKAL – 637 018
(AUTONOMOUS)

DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS

REGULATIONS 2019

(CBCS)

CURRICULUM AND SYLLABUS

I – II SEMESTER

M.C.A.Programme

(2 Years Duration)

PAAVAI ENGINEERING COLLEGE

(Autonomous)

Vision

To strive to be a globally model Institution all set for taking ‘lead-role’ in grooming the younger generation socially responsible and professionally competent to face the challenges ahead.

Mission

- To provide goal- oriented, quality – based and value – added education through state – of – the – art technology on a par with international standards.
- To promote nation – building activities in science, technology, humanities and management through research
- To create and sustain a community of learning that sticks on to social, ethical, ecological, cultural and economic upliftment.

PAAVAI ENGINEERING COLLEGE

(Autonomous)

DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS

Vision

To strive to be globally model institutions all set for taking lead role in grooming the younger generation socially responsible and professionally competent to face the challenges ahead

Mission

- To upgrade the academic activities by continuous improvement in the teaching - learning process with value based education.
- To enhance social responsibilities of the students necessary for successful practice of the profession.
- To facilitate research and industrial interaction.
- To mould the students into competent and creative technocrats to meet the growing global changes and challenges.
- To encourage the students as entrepreneurs and leaders of the society for the betterment of the Country.

Programme Educational Objectives (PEOs)		
Master of Computer Applications Graduates will be able to :		
PEO1	Global Reputation	To create value added, disciplined high profile Master of Computer Applications professionals for successful careers in their related industry that makes them globally reputed.
PEO2	Fundamental Knowledge	To develop the students with a sound foundation in mathematical, scientific and engineering fundamentals necessary to synthesize the technical core concepts focusing on skill development and knowledge up gradation which will lead to technical innovations.
PEO3	Continuous Learning	To practice and demonstrate the ability to use the domain knowledge and expertise through periodic assignments, performances and projects to continuously prove the functionality of Master of computer applications learning in social environmental aspects and to make allowances for further improvements.

Programme Outcomes (POs)		
Master of Computer Applications Graduates will be able to :		
PO1	Engineering knowledge	Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
PO2	Problem Analysis	Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
PO3	Design/Development of Solutions	Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
PO4	Conduct investigations of complex problems	Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
PO5	Modern Tool Usage	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.
PO6	The engineer and society	Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice

PO7	Environment and sustainability	Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
PO8	Ethics	Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
PO9	Individual and team work	Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
PO10	Communications	Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
PO11	Project management and finance	Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments
PO12	Life-long learning	Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Programme Specific Outcomes (PSOs)		
PSO1	Core Competencies:	Foundation of Computer System and Software development: Ability to understand the principles and working of computer systems for the development of software solutions
PSO2	Creativity and Design:	Applications of Computing and Research Ability: Ability to use knowledge in various domains to identify research gaps and hence to provide solution with new ideas and innovations.

PAAVAI ENGINEERING COLLEGE, NAMAKKAL – 637 018

(AUTONOMOUS)

MASTER OF COMPUTER APPLICATIONS

REGULATIONS 2019

(CHOICE BASED CREDIT SYSTEM)

CURRICULUM& SYLLABUS

SEMESTER I

S.No.	Category	Course	Course Title	L	T	P	C
Theory							
1	FC	PMA20101	Resource Management Techniques	3	1	0	4
2	PC	CA20101	Computer Organization	3	0	0	3
3	PC	CA20102	Operating System with Unix	3	0	0	3
4	PC	CA20103	Data Structures and Algorithms	3	0	0	3
5	PC	CA20104	Programming with Java	3	0	0	3
Practical							
6	PC	CA20105	Data Structures and Algorithms Laboratory	0	0	4	2
7	PC	CA20106	Programming with Java Laboratory	0	0	4	2
8	HS	PEN20101	Communication and Soft Skills Laboratory	0	0	2	1
			TOTAL	**15**	**1**	**10**	**21**

SEMESTER II

S.No.	Category	Course	Course Title	L	T	P	C
Theory							
1	PC	CA20201	Database Management Systems	3	0	0	3
2	PC	CA20202	Web Technology	3	0	0	3
3	PC	CA20203	Python and R Programming	3	0	0	3
4	PC	CA20204	Computer Communication Networks	3	0	0	3
5	PE	CA2015*	Professional Elective I	3	0	0	3
Practical							
6	PC	CA20205	Web Technology Laboratory	0	0	4	2
7	PC	CA20206	Python and R Programming Laboratory	0	0	4	2
8	PC	CA20207	DBMS Laboratory	0	0	4	2
			TOTAL	**15**	**0**	**10**	**21**

SEMESTER III

S.No	Category	Course	Course Title	L	T	P	C
Theory							
1	PC	CA20301	Cloud Computing	3	0	0	3
2	PC	CA20302	Machine Learning and Deep Learning	3	0	0	3
3	PC	CA20303	Internet of Things	3	0	0	3
4	PE	CA2025*	Professional Elective II	3	0	0	3
5	PE	CA2035*	Professional Elective III	3	0	0	3
Practical							
6	PC	CA20304	Mobile App Development Laboratory	0	0	4	2
7	EE	CA20305	Mini Project	0	0	4	2
8	EE	CA20306	Career Development Laboratory	0	0	2	1
			TOTAL	**15**	**0**	**10**	**20**

SEMESTER IV

S.No	Category	Course	Course Title	L	T	P	C
1	PC	CA20401	Cyber Security	3	0	0	3
2	PC	CA20402	Big Data Analytics	3	0	0	3
3	EE	CA20403	Project Work	0	0	24	12
			TOTAL	**0**	**0**	**24**	**18**

PROFESSIONAL ELECTIVES (PE)

ELECTIVE I

S.No	Category	Course	Course Title	L	T	P	C
1	PE	CA20151	Linux Internals	3	0	0	3
2	PE	CA20152	Information Security	3	0	0	3
3	PE	CA20153	Object Oriented Software Engineering	3	0	0	3
4	PE	CA20154	Advanced Database	3	0	0	3

ELECTIVE II

S.No	Category	Course	Course Title	L	T	P	C
1	PE	CA20251	PHP with MySQL	3	0	0	3
2	PE	CA20252	High Performance Computing	3	0	0	3
3	PE	CA20253	Software Project Management	3	0	0	3
4	PE	CA20254	Organizational Behavior	3	0	0	3

ELECTIVE III

S.No	Category	Course	Course Title	L	T	P	C
1	PE	CA20351	Artificial Intelligence	3	0	0	3
2	PE	CA20352	Block Chain Technology	3	0	0	3
3	PE	CA20353	Agile Software Engineering	3	0	0	3
4	PE	CA20354	Enterprise Resource Planning	3	0	0	3

FOUNDATION COURSE (FC)

S.No	Category	Course	Course Title	L	T	P	C
1	FC	PMA20101	Resource Management Techniques	3	1	0	4
			Total	**3**	**1**	**0**	**4**

PROFESSIONAL CORE (PC)

S.No.	Category	Course	Course Title	L	T	P	C
1	PC	CA20101	Computer Organization	3	0	0	3
2	PC	CA20102	Operating System with Unix	3	0	0	3
3	PC	CA20103	Data Structures and Algorithms	3	0	0	3
4	PC	CA20104	Programming with Java	3	0	0	3
5	PC	CA20105	Data Structures and Algorithms Laboratory	0	0	4	2
6	PC	CA20106	Programming with Java Laboratory	0	0	4	2
7	PC	CA20201	Database Management Systems	3	0	0	3
8	PC	CA20202	Web Technology	3	0	0	3
9	PC	CA20203	Python and R Programming	3	0	0	3
10	PC	CA20204	Computer Communication Networks	3	0	0	3
11	PC	CA20205	Web Technology Laboratory	0	0	4	2
12	PC	CA20206	Python and R Programming Laboratory	0	0	4	2
13	PC	CA20207	DBMS Laboratory	0	0	4	2
14	PC	CA20301	Cloud Computing	3	0	0	3
15	PC	CA20302	Machine Learning and Deep Learning	3	0	0	3
16	PC	CA20303	Internet of Things	3	0	0	3
17	PC	CA20304	Mobile App Development Laboratory	0	0	4	2
18	PC	CA20401	Cyber Security	3	0	0	3
19	PC	CA20402	Big Data Analytics	3	0	0	3
Total				**39**	**0**	**24**	**51**

EMPLOYABILITY ENHANCEMENT COURSES (EE)

S.No.	Category	Course	Course Title	L	T	P	C
1	EE	CA20305	Mini Project	0	0	4	2
2	EE	CA20206	Career Development Laboratory	0	0	2	1
3	EE	CA20403	Project Work	0	0	24	12
			Total	**0**	**0**	**30**	**15**

HUMANITIES AND SOCIAL SCIENCES (HS)

S.No.	Category	Course	Course Title	L	T	P	C
1	HS	PEN20101	Communication and Soft Skills Laboratory	0	0	2	1
			Total	**0**	**0**	**2**	**1**

Curriculum Structure

S.No	Category Name	Actual Credit Break Up	No. of Subjects
1	Professional Core (PC)	51	19
2	Professional Elective (PE)	09	03
3	Humanities and Social Sciences (HS)	01	01
4	Foundation Course (FC)	04	12
5	Employability Enhancement Courses (EE)	15	03
Total		80	38

SUMMARY							
S.No	**SUBJECT AREA**	**CREDITS AS PER SEMESTER**				**CREDITS TOTAL**	**CREDITS IN %**
		I	**II**	**III**	**IV**		
1	HS	1	-	-	-	**1**	**1.23 %**
2	FC	4	-	-	-	**4**	**4.93 %**
3	PC	16	18	11	6	**51**	**62.96 %**
4	PE	-	3	6	-	**9**	**11.11 %**
5	EE	-	-	3	12	**15**	**19.75 %**
TOTAL		**21**	**21**	**20**	**18**	**80**	**100 %**
Non Credit / Mandatory		-	-	-	-	-	-

PMA20101 RESOURCE MANAGEMENT TECHNIQUES 3 1 0 4

COURSE OBJECTIVES

To enable the students to

- provide the concept and an understanding of basic concepts in Operations Research
- study the Techniques for Analysis and Modeling in Computer Applications
- understand the mathematical model of linear programming problems
- develop and solve mathematical model of Transport and assignment problems
- understand network modeling for planning and scheduling the project activities

UNIT I LINEAR PROGRAMMING MODELS 12

Mathematical Formulation; Graphical Solution of linear programming models; Simplex method – Artificial Variable Techniques; Variants of Simplex method

UNIT II TRANSPORTATION AND ASSIGNMENT MODELS 12

Mathematical formulation of transportation problem; Methods for finding initial basic feasible solution; optimum solution- degeneracy; Mathematical formulation of assignment models; Hungarian Algorithm –Variants of the Assignment problem

UNIT III INTEGER PROGRAMMING MODELS 12

Formulation – Gomory's IPP method; Gomory's mixed integer method; Branch and Bound technique.

UNIT IV SCHEDULING BY PERT AND CPM 12

Network Construction – Critical Path Method; Project Evaluation and Review Technique; Resource Analysis in Network Scheduling

UNIT V QUEUEING MODELS 12

Characteristics of Queuing Models – Poisson Queues - (M / M / 1) : (FIFO / ∞ /∞), (M / M / 1) : (FIFO/ N / ∞), (M / M / C) : (FIFO / ∞ / ∞), (M / M / C) : (FIFO / N / ∞) models.

TOTAL PERIODS 60

COURSE OUTCOMES

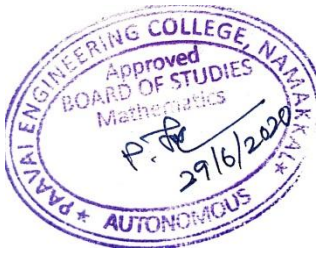
At the end of the course the students would be able to

- understand and apply linear, integer programming to solve operational problem with constraints
- apply transportation and assignment models to find optimal solution in warehousing
- prepare project scheduling using PERT and CPM
- identify and analyze appropriate queuing model to reduce the waiting time in queue
- use optimization concepts in real world problems

REFERENCES

1. Taha H.A., "Operations Research: An Introduction"8th Edition, Pearson Education, 2017
2. A.M.Natarajan, P.Balasubramani, A.Tamilarasi, "Operations Research", Pearson Education, Asia, 2018.
3. Prem Kumar Gupta, D.S. Hira, "Operations Research", S.Chand& Company Ltd, New Delhi, 3rd Edition, 2018.
4. John W. Chinneck "Feasibility and Infeasibility in Optimization Algorithms and Computational Methods Springer, 2018
5. Ravindran, Phillips, Solberg, "Operations Research: Principles And Practice", 2ndEdition, JohnWiley& Sons, 01-Jul-2017

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	2	2	3	2	-	-	-	-	-	-	-	1	2	2
CO2	1	2	2	2	3	-	-	-	-	-	-	2	1	2
CO3	2	2	3	2	3	-	-	-	-		2	1	1	3
CO4	2	2	2	2	-	-	-	2	-	-	-	-	2	2
CO5	1	2	3	1	-	-	-	1	-	-	-	3	2	3



CA20101 COMPUTER ORGANIZATION 3 0 0 3

COURSE OBJECTIVES

To enable the students to

- impart the knowledge in the field of digital fundamentals
- understand the knowledge about the various components of a computer and its internals
- know the functionality of the computer hardware with basic gate
- design other components using combinational and sequential logic
- study the importance of the hardware-software interface

UNIT I DIGITAL FUNDAMENTALS 9

Number Systems and Conversions; Boolean Algebra and Simplification – Minimization of Boolean Functions – Karnaugh Map; Logic Gates – NAND – NOR Implementation.

UNIT II COMBINATIONAL AND SEQUENTIAL CIRCUITS 9

Design of Combinational Circuits – Adder / Subtracter – Encoder – Decoder – MUX / DEMUX; Comparators - Flip Flops – Triggering **–** Master – Slave Flip Flop; State Diagram and Minimization –Counters – Registers.

UNIT III BASIC STRUCTURE OF COMPUTERS 9

Functional units – Basic operational concepts – Bus structures – Performance and Metrics; Instruction and instruction sequencing; Addressing modes – ALU design; Fixed point and Floating point operation.

UNIT IV PROCESSOR DESIGN 9

Processor basics – CPU Organization; Data path design – Control design; Basic concepts – Hard wired control– Micro programmed control – Pipeline control; Hazards – Super scalar operation.

UNIT V MEMORY, I/O SYSTEM AND PARALLEL PROCESSING 9

Memory technology – Memory systems – Virtual memory – Caches – Design methods – Associative memories; Input/output system – Programmed I/O – DMA and Interrupts – I/O Devices and Interfaces; - Multiprocessor Organization – Symmetric multiprocessors – Cache Coherence; Clusters - Non Uniform Memory Access - Vector Computation.

TOTAL PERIODS 45

COURSE OUTCOMES

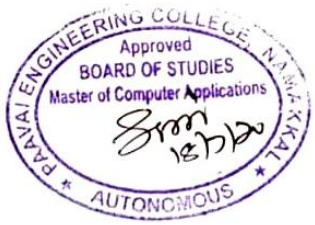
At the end of the course the students would be able to

- design digital circuits by simplifying the boolean functions
- understand the organization and working principle of computer hardware components
- understand mapping between virtual and physical memory
- acquire knowledge about multiprocessor organization and parallel processing
- trace the execution sequence of an instruction through the processor

REFERENCES

1. Morris Mano, "Digital Design", Prentice Hall of India, Fourth Edition 2017.
2. William Stallings, "Computer Organization & Architecture – Designing for Performance" 9th Edition 2017.
3. Charles H. Roth, Jr., "Fundamentals of Logic Design", Jaico Publishing House, Mumbai, Fourth Edition, 2018.
4. David A. Patterson and John L. Hennessy, "Computer Organization and Design: The hardware /Software Interface", Fourth Edition, Morgan Kaufmann / Elsevier, 2017.

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	1	2	3	1	1	-	-	-	-	-	-	1	1	2
CO2	1	2	3	2	3	-	-	-	-	-	3	2	1	2
CO3	1	1	3	3	2	3	-	3	-		2	1	1	3
CO4	1	1	1	1	-	3	3	3	2	3	3	3	2	2
CO5	2	2	3	3	3	-	-	-	2	-	-	2	2	2



CA20102 OPERATING SYSTEM WITH UNIX 3 0 0 3

COURSE OBJECTIVES

To enable the students to

- impart the knowledge in the field of operating systems
- impart knowledge about the process management and scheduling
- design and realize the functionality of the deadlocks
- learn knowledge about the unix file systems
- study the importance of the process management

UNIT I INTRODUCTION 9

Introduction to Operating Systems - Operating System Operations; Computing Environments; Operating System Services - System Calls - Types of System Calls - System Programs; Operating System Structure - Virtual Machines - System boot; Overview of Process Process Concept - Process Scheduling - Operations on Processes; Inter Process Communication; Multithreaded Programming - Multithreading Models

UNIT II PROCESS MANAGEMENT PROCESS SCHEDULING 9

Basic Concepts - Scheduling Criteria - Scheduling Algorithms - Multiple Processor Scheduling - Process Synchronization – Synchronization; The Critical Section Problem - Peterson's Solution; Semaphores - Classical Problems of Synchronization

UNIT III DEADLOCKS 9

System model - Deadlock Characterization - Methods for handling deadlocks - Deadlock Prevention - Deadlock Avoidance - Deadlock Detection and Recovery from deadlock; Memory Management - Memory Management Strategies – Background – Swapping; Contiguous Memory Allocation – Paging – Segmentation; Virtual Memory Management

UNIT IV UNIX FILE SYSTEM 9

File System - The Parent Child Relationship - The HOME Variable - The Home Directory – pwd – cd – mkdir – rmdir - Absolute Pathnames - Relative Pathnames; The Unix File System - Input Mode and Command Mode - File Ownership - File Permissions – chmod - Directory Permissions; Changing the File Ownership - More File Attributes; File Systems and Inodes - Hard Links - Symbolic Links - The Directory – umask - Modification and Access Times - find.

UNIT V THE PROCESS 9

Process Basics - Process Status - System Processes - Mechanism of Process Creation; Internal and External Commands; Running Jobs in Background; Killing Processes with Signals - Job Control

TOTAL PERIODS 45

COURSE OUTCOMES

At the end of the course the students would be able to

- apply the fundamentals of operating systems
- understand the model operating system with given valid systems
- design algorithms for the given problems
- demonstrate the unix environments including file processing
- analyze the process management

REFERENCES

1. Sumitabha Das: UNIX Concepts and Applications, 4th Edition, Tata McGraw Hill, 2006.
2. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating Systems Principles, 8th Edition, Wiley – India.
3. D M Dhamdhere: Operating Systems – A Concept Based Approach, 2nd Edition, Tata McGraw – Hill, 2002.
4. Harvey M Deital: Operating Systems, 3rdEdition, Addison Wesley, 1990
5. P C P Bhatt: Operating Systems, 2ndEdition, PHI, 2006

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	1	2	1	2	-	-	-	-	-	-	-	1	1	2
CO2	1	3	2	1	3	-	-	-	-	-	-	2	2	2
CO3	2	2	2	2	3	-	-	-	-		2	2	1	2
CO4	2	2	2	2	-	-	-	3	-	-	-	-	2	2
CO5	1	2	1	3	-	-	-	2	-	-	-	2	2	3

CA20103 DATA STRUCTURES AND ALGORITHMS 3 0 0 3

COURSE OBJECTIVES

To enable the students to

- understand the linear and non linear data structures available in solving problems
- know about the sorting and searching techniques and its efficiencies
- apply the various algorithm design techniques
- import the data structures and algorithms in real time applications
- analyze the efficiency of algorithm

UNIT I LINEAR DATA STRUCTURES 9

Introduction - Abstract Data Types (ADT) – Arrays and its representation –Structures – Stack – Queue – Circular Queue; Applications of stack; Infix to postfix conversion – evaluation of expression; Applications of Queue; Linked Lists – Doubly Linked lists – Applications of linked list; Polynomial Addition

UNIT II TREE STRUCTURES 9

Need for non-linear structures – Trees and its representation; Binary Tree – expression trees – Binary tree Traversals; left child right sibling data structures for general trees; applications of trees – Huffman Algorithm - Binary search tree.

UNIT III BALANCED SEARCH TREES, SORTING AND INDEXING 9

AVL trees–B-Trees; Sorting–Bubble sort - Quick Sort - Insertion Sort – Heap sort; Hashing - Hashing functions; Collision Resolution Techniques - Separate chaining - Open addressing - Multiple hashing.

UNIT IV GRAPHS 9

Definitions – Representation of graph; Graph Traversals - Depth-first traversal – breadth-first traversal; applications of graphs; Topological sort; shortest-path algorithms – minimum spanning tree – Prim's and Kruskal's algorithms; biconnectivity – Euler circuits.

UNIT V ALGORITHM DESIGN AND ANALYSIS 9

Algorithm Analysis – Asymptotic Notations; Divide and Conquer – Merge Sort – Binary Search; Greedy Algorithms – Knapsack Problem; Dynamic Programming – Warshall’s Algorithm for Finding Transitive Closure; Backtracking – Sum of Subset Problem; Branch and Bound – Travelling Salesman Problem.

TOTAL PERIODS 45

COURSE OUTCOMES

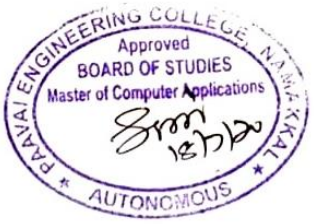
At the end of the course the students would be able to

- select and apply the data structure to suit any given problem
- design their own data structure according to the applications
- apply the algorithm design techniques to any of the real world problem
- develop any new application with the help of data structures and algorithms
- write efficient algorithm for a given problem and able to analyze its time complexity

REFERENCES

1. M. A. Weiss, “Data Structures and Algorithm Analysis in C++”, Pearson Education Asia, 2018.
2. Tanaenbaum A.S.,Langram Y. Augestein M.J “Data Structures using C” Pearson Education , 2017
3. AnanyLevitin “Introduction to the Design and Analysis of Algorithms” Pearson Education 2018.
4. E. Horowitz, S.Sahni and Dinesh Mehta, “Fundamentals of Data structures in C++”, University Press, 2017.
5. ReemaThareja, “Data Structures using C”, Oxford Press, 2018.

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	2	2	3	2	-	-	-	-	-	-	-	1	2	2
CO2	1	2	2	2	3	-	-	-	-	-	-	2	1	2
CO3	2	2	1	2	2	-	-	-	-		2	2	2	2
CO4	2	1	2	2	-	-	-	1	-	-	-	-	2	2
CO5	1	1	3	2	-	-	-	2	-	-	-	1	1	3



CA20104 PROGRAMMINGWITH JAVA 3 0 0 3

COURSE OBJECTIVES

To enable the students to

- understand an overview of oops concepts
- know the basics of constructor and inheritance concepts
- learn the knowledge in packages and threads
- develop programs in JDBC and RMI concepts
- practice the web development techniques on client-side

UNIT I INTRODUCTION TO JAVA 9

Introduction to java - Features of Java; Object Oriented Concepts – Lexical issues; Data Types – Variables - Arrays – Operators – Control Statements.

UNIT II CLASSES, OBJECTS AND METHODS 9

Classes ; Objects ; Constructors ; Overloading Methods – Access Control – Static an Fixed Methods; Inner Class – String Class; Inheritance – Overriding Methods; using Super – Abstract Class

UNIT III PACKAGES 9

Packages – Access Protection – Importing Packing; Interfaces – Exception Handling – Throw and Throws; Thread – Synchronization – Messaging; Runnable Interface – Inter Thread Communication.

UNIT IV JDBC AND RMI 9

JDBC–Introduction to JDBC and its components – Implementing JDBC in Applet; Introduction to RMI-Structure of RMI – Implementing RMI.

UNIT V APPLET AND AWT 9

AWT package – Layouts – Containers; Event Package – Event Model; Garbage Collection; Multithreading; Language Packages.

TOTAL PERIODS 45

COURSE OUTCOMES

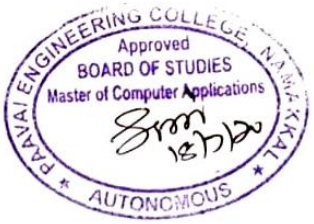
At the end of the course the students would be able to

- acquire knowledge about the overview on oops concepts
- explore the basics of constructor and inheritance in of java
- learn in java packages and threads and able to develop programs using it
- acquire knowledge in JDBC and RMI
- design web development techniques on client-side

REFERENCES

1. RajkumarBuyya, S.ThamaraiSelvi, Xingchen Chu, "Object–Oriented Programming with JAVA: Essentialsand Applications", Fourth Edition, Tata McGraw Hill Education Private Limited, 2018
2. Herbert Schildt , " Java: The Complete Reference ", Ninth Edition, McGraw-Hill, 2017.
3. Felipe Gutierrez, "Introducing Spring Framework: A Primer" Apress, 2018.
4. P.Naughton and H.Schildt, "Java 2(Complete Reference)" by fourth Edition.
5. Kathy Sierra & Bert Bates, "Head First Servlets &Jsp" by O'Reilly publications.

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	1	2	1	2	-	-	-	-	-	-	-	1	1	2
CO2	1	2	2	2	3	-	-	-	-	-	-	2	1	2
CO3	2	2	2	2	2	-	-	-	-		2	1	1	3
CO4	2	1	2	2	-	-	-	2	-	-	-	-	3	2
CO5	1	2	1	2	-	-	-	2	-	-	-	2	2	2



CA20105 DATA STRUCTURES AND ALGORITHMS LABORATORY 0 0 4 2

COURSE OBJECTIVES

To enable the students to

- using the data structures and algorithms in real time applications
- analyze the efficiency of algorithm
- apply the various algorithm design techniques
- analyze the efficiency of algorithm

LIST OF EXPERIMENTS

1. Polynomial Addition using array
2. Array implementation of stack
3. Array implementation of Queue
4. Infix to postfix conversion
5. Singly Linked List operations
6. Binary tree traversals
7. Quick sort
8. Dictionary application using any of the data structure
9. Find the Shortest Path using Dijkstra's Algorithm – Greedy method
10. Warshall's Algorithm for finding transitive closure using Dynamic programming
11. Sum of subset problem using backtracking

TOTAL PERIODS 60

COURSE OUTCOMES

At the end of the course the students would be able to

- develop any new application with the help of data structures and algorithms
- write efficient algorithm for a given problem
- analyze its time complexity of algorithms
- write backtracking algorithm for a given problem

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	2	2	2	2	-	-	-	-	-	-	-	2	2	2
CO2	1	2	3	3	2	-	-	-	-	-	-	2	2	1

CA20106 PROGRAMMING WITH JAVA LABORATORY 0 0 4 2

COURSE OBJECTIVES

To enable the students to

- practice basics of java and String handling
- analyze the exception and multithreading
- analyze the AWT concept
- practice the Layout Manager

LIST OF EXPERIMENTS

1. Writing Java programs by making use of class, interface, package etc for the following
    a. Different types of inheritance study
    b. Uses of 'this' keyword
    c. Polymorphism
    d. Creation of user specific packages
    e. User specific exception handling
2. Write programs in Java for event handling Mouse and Keyboard events.
3. Write programs in Java to manipulate Text Area, Canvas, Scroll Bars, Frames and Menus using AWT.
4. Write programs in Java using Layout Manager create different applications
5. Create an application using Java Applets.

TOTAL PERIODS 60

COURSE OUTCOMES

At the end of the course the students would be able to

- develop the static and dynamic java programs
- acquire knowledge about the exception and thread programs
- explore the features and create interactive web pages using applet
- design the Layout Manager in AWT

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	2	2	2	2	-	-	-	2	-	-	-	-	2	2
CO2	1	2	3	1	-	-	-	1	-	-	-	3	2	3

PEN20102 COMMUNICATION AND SOFT SKILLS LABORATORY 0 0 2 1

COURSE OBJECTIVES

To enable the students to

- develop the ability to communicate effectively in spoken English
- draft flawless resume in English
- enhance their soft skills and interpersonal skills
- equip the learners to make effective presentations on topics in engineering and technology
- participate successfully in group discussions

LIST OF EXPERIMENTS

1. Me chart
2. Role Play
3. Narrating Stories
4. Online Test
5. Job Application with Resume
6. Presentation Skills
   - i) Technical Presentation
   - ii) Non - Technical Presentation
7. Group Discussion
8. Interview Skills

TOTAL PERIODS 30

COURSE OUTCOMES

At the end of the course the students would be able to

- speak effectively and confidently in English
- attend job interviews with confidence
- write effective job applications with resume
- participate in GD with involvement and confidence
- communicate appropriately at different context

REFERENCES

1. Rizvi, Ashraf. M. Effective Technical Communication. Tata McGraw-Hill, New Delhi.2005.
2. Anderson, P.V. "Technical Communication", Thomson Edition, New Delhi, 2007.
3. Rizvi, Ashraf. M. Effective Technical Communication. Tata McGraw-Hill, New Delhi, 2012
4. Kumar Sanjay, PushpLata, "Communication Skills (With CD)", Oxford University Press, New Delhi. 2011
5. Dutt, Kiranmai P and GeethaRajeevan, "Basic Communication Skills", Foundation Books, New Delhi. 2007.

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	-	-	-	-	-	2	-	-	3	3	-	-	-	-
CO2	-	-	-	3	3	-	2	2	3	-	-	-	3	-
CO3	-	-	2	1	3	-	3	3	3	3	2	3	2	3
CO4	-	-	-	2	-	2	-	3	3	3	3	-	2	2
CO5	-	-	-	-	-	2	-	3	3	3	1	-	1	-

SEMESTER II

CA20201 DATABASE MANAGEMENT SYSTEMS 3 0 0 3

COURSE OBJECTIVES

To enable the students to

- understand the fundamentals of data models and conceptualize and depict a database system
- import the fundamentals of ER diagram
- make a study of SQL and relational database design
- know about data storage techniques an query processing
- impart knowledge in transaction processing, concurrency

UNIT I INTRODUCTION 9

File systems versus Database systems; Data Models; DBMS Architecture; Data Independence – Data Modeling using Entity; Relationship Model – Enhanced E-R Modeling.

UNIT II RELATIONAL MODEL AND QUERY EVALUATION 9

Relational Model Concepts – Relational Algebra; SQL – Basic Queries – Complex SQL Queries; Views – Constraints; Relational Calculus – Tuple Relational Calculus – Domain Relational Calculus; overview of commercial RDBMSs – Database Design; Functional Dependencies – Normal Forms – 1NF – 2NF-3NF-BCNF – 4NF-5NF; Algorithms for Executing Query Operations — Cost Estimation.

UNIT III TRANSACTION PROCESSING 9

Transaction Processing – Properties of Transactions - Serializability – Transaction support in SQL; Locking Techniques – Time Stamp ordering – Validation Techniques; Granularity of Data Items – Recovery concepts – Shadow paging–Log Based Recovery; Database Security Issues– Access control – Statistical Database Security .

UNIT IV FILES AND INDEXING 9

File operations–Hashing Techniques; Indexing – Single level and Multi-level Indexes – B+ tree – Static Hashing - Indexes on Multiple Keys.

UNIT V SPECIAL PURPOSE DATABASES 9

OODBMS - Object-Based Databases - OO Data Model - OO Languages – Persistence; Object Relational Databases - Temporal Databases – Mobile Databases – Spatial Databases; Case Study for Design and Manage the Database for any Project .

TOTAL PERIODS 45

COURSE OUTCOMES

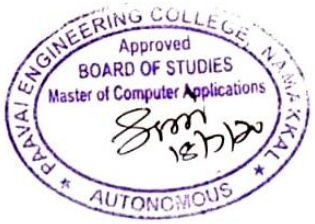
At the end of the course the students would be able to

- understand the basic concepts of the database and data models
- design a database using ER diagrams and map ER into Relations and normalize the relations
- acquire the knowledge of query evaluation to monitor the performance of the DBMS
- develop a simple database applications using normalization
- acquire the knowledge about different special purpose databases

REFERENCES

1. Abraham Silberschatz, Henry F.Korth and S.Sundarshan "Database System Concepts", Sixth Edition, McGraw Hill, 2017.
2. C.J. Date, "An Introduction to Database Systems", Eight Editions, Pearson Education Delhi, 2018.
3. Ramez Elamassri and Shankant B-Navathe, "Fundamentals of Database Systems", Sixth Edition, Pearson Education Delhi, 2017.
4. Raghu Ramakrishnan, Johannes Gehrke, "Database management systems" McGraw Hill, 2018.
5. Peter Rob, Carlos Coronel, "Database System Concepts", Cengage Learning, 2018.

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	1	2	1	2	-	-	-	-	-	-	-	1	1	2
CO2	1	2	2	2	3	-	-	-	-	-	-	2	1	2
CO3	2	2	2	2	2	-	-	-	-		2	1	1	3
CO4	2	1	2	2	-	-	-	2	-	-	-	-	3	2
CO5	1	2	1	2	-	-	-	2	-	-	-	2	2	2



CA20202 **WEB TECHNOLOGY** **3 0 0 3**

COURSE OBJECTIVES

To enable the students to

- understand the HTML programming
- create and execute advanced dynamic web pages with scripts
- create client side scripting language java script
- study the basics of XML, JSP programming
- learn the logic behind advanced web applications

UNIT I INTRODUCTION TO HTML 9

Introduction to HTML and HTML5; Formatting and Fonts-Commenting Code; Anchors; Backgrounds – Images – Hyperlinks;– Lists – Tables; Frames - HTML Forms.

UNIT II STYLE SHEET 9

The need for CSS, Introduction to CSS – Basic syntax and structure; Inline Styles – Embedding Style Sheets - Linking External Style Sheets; Backgrounds - Manipulating text; Margins and Padding; Positioning using CSS.

UNIT III JAVA SCRIPT 9

Core Java Script–Variables–Constant; Expression Conditions; Relational Operators; Data Types; Flow Control–Functions; Objects–Data type Conversation & Equality; Windows and Frames–Forms and Data

UNIT IV SERVLET 9

Servlet Features – 3 Tier Applications; Servlet API – Explaining Servlet Life Cycle; Creating sample Servlet - Working with Servlet Config, Servlet Context; HttpServletRequest and HttpServletResponse – Request Delegation and Request scope

UNIT V JSP 9

Basic JSP Architecture – Life Cycle of JSP; JSP Tags and Expressions – Role of JSP; Scripting elements – Implementation of JSP

TOTAL PERIODS 45

COURSE OUTCOMES

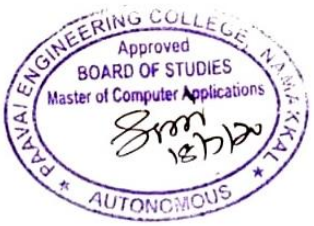
At the end of the course the students would be able to

- understand the HTML programming
- create and execute advanced dynamic web pages with scripts
- create client side scripting language java script
- understand the basics of XML, JSP programming
- understand the logic behind advanced web applications

REFERENCES

1. Bayross, "Teach yourself Web Technology part 1 & 2" BPB 2017
2. J.Niederst, "Web Design in a Nuttshell", SPD, 2017
3. A.Subramanyam, "Java Server Programming", SPD, J2EE edition, 2016
4. M.Young, "Step by Step XML", PHI Second Edition, 2016

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	1	2	1	2	-	-	-	-	-	-	-	1	1	2
CO2	1	2	2	2	3	-	-	-	-	-	-	2	1	2
CO3	2	2	2	2	2	-	-	-	-		2	1	1	3
CO4	2	1	2	2	-	-	-	2	-	-	-	-	3	2
CO5	1	2	1	2	-	-	-	2	-	-	-	2	2	2



CA20203 PYTHON AND R PROGRAMMING 3 0 0 3

COURSE OBJECTIVES

To enable the students to

- acquire knowledge in Python and R programming
- develop Python programs with conditionals and loops and data structures
- learn how to design and programs Python applications
- import to build and package Python modules for reusability
- understand and practice web development techniques on client-side and server side

UNIT I INTRODUCTION TO PYTHON PROGRAMMING 9

Python interpreter and interactive mode; values and types variables, expressions, statements, tuple assignment, Order of operations, comments, debugging; modules and functions: function Calls, adding new functions, Definitions and Uses, flow of execution, parameters and arguments. Conditionals: Boolean values and operators, conditional (if), alternative (if-else), chained conditional (if-elif-else); Iteration: state, while, for, range, break, continue, pass; recursion; Strings: string slices, immutability, string and methods, string module; Lists as arrays.

UNIT II LISTS, TUPLES, DICTIONARIES 9

Lists: Traversing a List, list operations, list slices, list methods, Map, Filter and Reduce, list loop, mutability, aliasing, cloning lists, list parameters; Dictionaries: operations and methods; advanced list processing - list comprehension; Tuples: tuple assignment, tuple as return value.

UNIT III FILES, MODULES, PACKAGES: 9

Files and exception: text files, reading and writing files, format operator; command line arguments, errors and Exceptions, handling exceptions, modules, packages: PANDAS, NUMPY, SCIKIT-LEARN;

UNIT IV INTRODUCTION TO R PROGRAMMING: 9

Introduction and Preliminaries, numbers and vectors, Objects, their modes and attributes, Ordered and unordered factors, Arrays and matrices, Lists and data frames, Grouping, loops and conditional execution, functions.

UNIT V STATISTICAL MODELS, GRAPHICAL PROCEDURES, PACKAGES 9

Statistical models: Defining statistical models; formulae, Linear models, Generic functions for extracting model information, Analysis of variance and model comparison, Updating fitted models, Generalized linear models, Nonlinear least squares and maximum likelihood models; Graphical procedures: High-level and low-level plotting commands, graphics parameters and list, Dynamic graphics. Packages: Standard packages, Contributed packages and CRAN, Namespaces.

TOTAL PERIODS 45

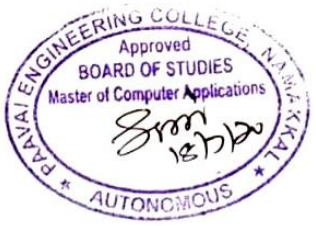
At the end of the course the students would be able to

- problem solving and programming capability
- construct and execute basic programs in Python
- use external libraries and packages with Python
- construct and execute basic programs in R using programming techniques
- use external R-packages in statistical and graphics

REFERENCES

1. Allen B. Downey, "Think Python: How to Think like a Computer Scientist‘‘, 2nd edition, Updated for Python Shroff/O‘Reilly Publishers, 2017.
2. Guido van Rossum and Fred L. Drake Jr, ―An Introduction to Python – Revised and updated for Python 3.2, Network Theory Ltd., 2017.
3. William N. Venables, David M. Smith, An Introduction to R: A Programming Environment for Data Analysis and Graphics, 2nd edition, Network Theory Ltd, 2018.
4. John V Guttag, ―Introduction to Computation and Programming Using Python‘‘, Revised and expanded Edition, MIT Press, 2018.
5. Robert Sedgewick, Kevin Wayne, Robert Dondero, Introduction to Programming in Python: Interdisciplinary Approach, Pearson India Education Services Pvt. Ltd., 2017

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	1	2	1	2	-	-	-	-	-	-	-	1	1	2
CO2	1	2	2	2	3	-	-	-	-	-	-	2	1	2
CO3	2	2	2	2	2	-	-	-	-		2	1	1	3
CO4	2	1	2	2	-	-	-	2	-	-	-	-	3	2
CO5	1	2	1	2	-	-	-	2	-	-	-	2	2	2



CA20204 COMPUTER COMMUNICATION NETWORKS 3 0 0 3

COURSE OBJECTIVES

To enable the students to

- understand networking concepts and basic communication model
- study the network architectures and components required for data communication
- analyze the function and design strategy of physical, data link, network layer and transport layer
- acquire knowledge of various application protocol standard developed for internet
- learn the flow of information from one node to another node in the network

UNIT I INTRODUCTION 9

Data Communications – Networks – Protocols and Standards; Network Models – OSI Model – Layers – TCP/IP Protocol Suite–Addressing; Digital Transmission – Digital-to-Digital Conversion – Analog-to-Digital Conversion; Transmission Modes - Analog Transmission – Digital-to-Analog Conversion – Analog-to-analog Conversion; Transmission Media – Guided and Unguided Media.

UNIT II DATA LINK LAYER 9

Error – detection and correction – Parity – LRC – CRC – Hamming code – Flow Control and Error control; stop and wait – go back-N ARQ – selective repeat ARQ; sliding window – HDLC; LAN - IEEE 802.4 - IEEE 802.5 IEEE 802.11; FDDI - SONET – Bridges.

UNIT III NETWORK LAYER 9

Internetworks; Circuit Switching- Packet Switching and Datagram approach; IP addressing methods– Subnetting; Routing – Distance Vector Routing – Link State Routing – BGP - Routers.

UNIT IV TRANSPORT LAYER 9

Duties of transport layer; Multiplexing – Demultiplexing; Sockets – User Datagram Protocol (UDP) – Transmission Control Protocol (TCP); Congestion Control – Quality of services (QOS) – Integrated Services.

UNIT V APPLICATIONS 9

Domain Name Space (DNS) – SMTP – FTP – HTTP; WWW – Security – Cryptography.

TOTAL PERIODS 45

COURSE OUTCOMES

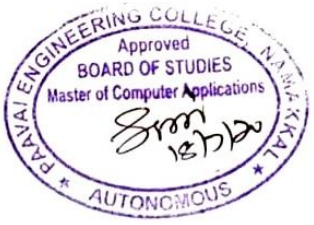
At the end of the course the students would be able to

- identify the components required to build different types of networks
- understand the functionalities needed for data communication into layers
- choose the required functionality at each layer for given application
- understand the working principles of various application protocols
- acquire knowledge about security issues and services available

REFERENCES

1. Larry L. Peterson & Bruce S. Davie, "Computer Networks – A systems Approach", Fourth Edition, Harcourt Asia / Morgan Kaufmann, 2018.
2. William Stallings, "Data and Computer Communications", Ninth Edition, Prentice Hall, 2017.
3. Forouzan, " Data Communication and Networking", Fifth Edition , TMH 2018
4. Andrew S.Tannenbaum David J. Wetherall, "Computer Networks" Fifth Edition , Pearson Education 2017
5. James F. Kurose, Keith W. Ross, "Computer Networking: A Top-down Approach, Pearson Education, Limited, sixth edition, 2018

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	2	2	3	2	-	-	-	-	-	-	-	1	2	2
CO2	1	2	2	2	3	-	-	-	-	-	-	2	1	2
CO3	2	2	3	2	3	-	-	-	-		2	1	1	3
CO4	2	2	2	2	-	-	-	2	-	-	-	-	2	2
CO5	1	2	3	1	-	-	-	1	-	-	-	3	2	3



CA20205 WEB TECHNOLOGY LABORATORY 0 0 4 2

COURSE OBJECTIVES

To enable the students to

- understand and practice the role of language like HTML, XML, Servlet, JSP
- understand the concept of style sheets
- develop and construct the web applications with high reliability
- develop the application using java script

LIST OF EXPERIMENTS

1. Prepare a bio data using HTML formatting Tags
2. Prepare Student Mark sheet using Table tags including colspan and rowspan
3. Develop a webpage for Shopping mall using frames and links using HTML
4. Create a web page with all types of Cascading style sheets.
5. Write a program in JavaScript for a simple calculator
6. Linking the image without clicking using mouseover event in javascript
7. Write a program to change the background of the webpage using javascript
8. Create a program for client and server side program
9. Create a webpage for server side program to find the number of hits.
10. Write a program for html to JSP to generate username and password

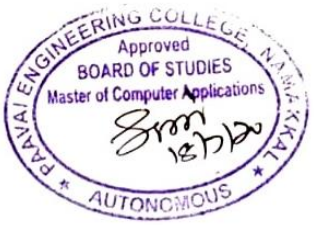
TOTAL PERIOD 60

COURSE OUTCOMES

At the end of the course the students would be able to

- model software projects into high level design using HTML, CSS
- develop the application using style sheets
- demonstrate the application in java script
- design the web page

Mapping of Courses Outcomes with Programme Outcomes: (1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	2	1	2	3	-	-	-	2	-	-	-	-	3	2
CO2	2	2	2	2	-	-	-	1	-	-	-	2	2	3



CA20206 PYTHON AND R PROGRAMMING LABORATORY 0 0 4 2

COURSE OBJECTIVES

To enable the students to

- understand the scripting Programming
- practice the dynamic scripting Programming
- analyze and practice of Python Programming
- analyze and practice of R Programming

LIST OF EXPERIMENTS

1. Explain and use basic concepts in programming
2. Use of conditional statements in Python
3. Use of looping statements in Python
4. Creating different types of arrays in Python
5. Usage of array functions in Python
6. Creating user defined functions in Python
7. Explain and use basic concepts in R programming
8. Assign and manipulate data structures using R programming
9. Create user-defined functions using R programming
10. Use of looping statements using R programming
11. Condition statements and debugging using R programming
12. import/export of data from file using R programming

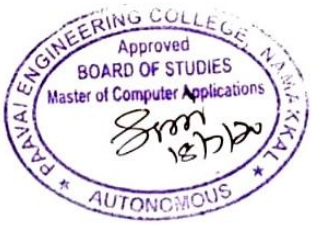
TOTAL PERIODS 60

COURSE OUTCOMES

At the end of the course the students would be able to

- explore the mark-up languages features and create interactive web pages using them
- acquire knowledge about Open source Python libraries
- acquire knowledge about Open source R libraries
- demonstrate the applications

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	2	1	2	3	-	-	-	2	-	-	-	-	3	2
CO2	2	2	2	2	-	-	-	1	-	-	-	2	2	3



CA20207 **DBMS LABORATORY** **0 0 4 2**

COURSE OBJECTIVES

To enable the students to

- understand the fundamentals of data models
- understand the fundamentals of database system using ER diagram
- make a study of SQL and relational database design
- make a study of PL/SQL Triggers

LIST OF EXPERIMENTS

1. Creation of base tables and views.
2. Data Manipulation INSERT, DELETE and UPDATE in Tables. SELECT, Sub Queries and JOIN
3. Data Control Commands
4. High level language extensions – PL/SQL. Or Transact SQL – Packages 5. Use of Cursors, Procedures and Functions
5. Embedded SQL or Database Connectivity.
6. Oracle or SQL Server Triggers – Block Level – Form Level Triggers
7. Working with Forms, Menus and Report Writers for an application project in any domain 9. Front-end tools –Visual Basic/Developer 2000.

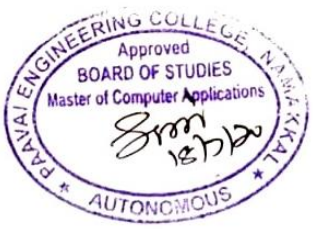
TOTAL PERIODS 60

COURSE OUTCOMES

At the end of the course the students would be able to

- understand the concepts of the database and data models
- design a database using ER diagrams
- design a database using Relations and normalize the relations
- develop application using visual basic

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	2	2	2	2	-	-	-	-	-	-	-	2	2	2
CO2	1	2	3	3	2	-	-	-	-	-	-	2	2	1



PROFESSIONAL ELECTIVE – I

CA20151 LINUX INTERNALS 3 0 0 3

COURSE OBJECTIVES

To enable the students to

- understand an overview of Linux concepts
- know the basics analysis of buffer cache concepts
- learn the knowledge in system calls
- develop the system process concepts
- study and practice of memory management concept

UNIT I INTRODUCTION 9

General Review of the System-History-System structure; User Perspective-Operating System Services; Assumptions About Hardware. Introduction to the Kernel-Architecture; System Concepts; Data Structures-System Administration.

UNIT II ANALYSIS OF BUFFER CACHE 9

The Buffer Cache-Headers-Buffer Pool-Buffer Retrieval; Reading and Writing Disk Blocks - Advantages and Disadvantages Internal Representation of Files; inodes - Structure-Directories-Path Name to inode - Super Block - inode Assignment; Allocation of Disk Blocks -Other File Types

UNIT III SYSTEM CALLS 9

System Calls for the File System-open-read-write-lseek-Close-Create-Special files Creation; Change Directory and Change Root-Change Owner and Change Mode-Stat-fstat; Pipes-Dup-Mount-unmount- Link-Unlink; File System Abstraction-Maintenance.

UNIT IV SYSTEM PROCESSES 9

The System Representation of Processes-States-Transitions-System Memory; Context of a Process-Saving the Context; Manipulation of a Process Address Space-Sleep Process Control; signals-Process Termination-Awaiting -Invoking other Programs; The Shell-System Boot and the INIT Process.

UNIT V MEMORY MANAGEMENT 9

Memory Management Policies; Swapping-Demand Paging-a Hybrid System-I/O; Subsystem-Driver Interfaces-Disk Drivers-Terminal Drivers.

TOTAL PERIODS 45

COURSE OUTCOMES

At the end of the course the students would be able to

- acquire knowledge about the overview of linux concepts
- explore the basics of buffer cache concept
- learn and understand the system calls concept
- acquire knowledge in memory management control
- understand the concept of costing

REFERENCES

1. Maurice J. Bach, "The Design of the Unix Operating System", Pearson Education, 2017
2. Uresh Vahalia, "UNIX Internals: The New Frontiers", Prentice Hall, 2017.
3. John Lion, "Lion's Commentary on UNIX", 6th edition, Peer-to-Peer Communications, 2018.
4. Daniel P. Bovet & Marco Cesati, “Understanding the Linux Kernel”, O’REILLY, Shroff Publishers & Distributors Pvt. Ltd, 2018
5. M. Beck ET AL, “Linux Kernel Programming”, Pearson Education Asia, 2016

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	1	2	1	2	-	-	-	-	-	-	-	1	1	2
CO2	1	2	2	2	3	-	-	-	-	-	-	2	1	2
CO3	2	2	2	2	2	-	-	-	-		2	1	1	3
CO4	2	1	2	2	-	-	-	2	-	-	-	-	3	2
CO5	1	2	1	2	-	-	-	2	-	-	-	2	2	2

CA20152 INFORMATION SECURITY 3 0 0 3

COURSE OBJECTIVES

To enable the students to

- understand the basics of cryptography algorithms
- learn to find the vulnerabilities in programs and to overcome them,
- know the different kinds of security threats in networks and its solution
- exposed the different kinds of security threats in databases and solutions available
- study about the models and standards for security

UNIT I ELEMENTARY CRYPTOGRAPHY **9**

Terminology and Background – Substitution Ciphers – Transpositions; Making Good Encryption Algorithms- Data Encryption Standard- AES Encryption Algorithm – Public Key Encryption; Cryptographic Hash Functions – Key Exchange – Digital Signatures – Certificates.

UNIT II PROGRAM SECURITY **9**

Secure programs – Non-malicious Program Errors – Viruses – Targeted Malicious code – Controls Against Program Threat; Control of Access to General Objects – User Authentication – Good Coding Practices; Open Web Application Security Project Flaws – Common Weakness Enumeration Most Dangerous Software Errors.

UNIT III SECURITY IN NETWORKS **9**

Threats in networks – Encryption – Virtual Private Networks – PKI – SSH – SSL – IPSec – Content Integrity; Access Controls – Wireless Security – Honeypots – Traffic Flow Security – Firewalls; Intrusion Detection Systems – Secure e-mail.

UNIT IV SECURITY IN DATABASES **9**

Security requirements of database systems – Reliability and Integrity in databases; Redundancy – Recovery – Concurrency/ Consistency – Monitors – Sensitive Data; Types of disclosures – Inference-finding and confirming SQL injection.

UNIT V SECURITY MODELS AND STANDARDS **9**

Secure SDLC – Secure Application Testing – Security architecture models; Trusted Computing Base – Bell-LaPadula Confidentiality Model; Biba Integrity Model – Graham-Denning Access Control Model – Harrison - Ruzzo-Ulman Model; Secure Frameworks – COSO – COBIT – Compliances – PCI DSS; Security Standards - ISO 27000 family of standards – NIST.

TOTAL PERIODS 45

COURSE OUTCOMES

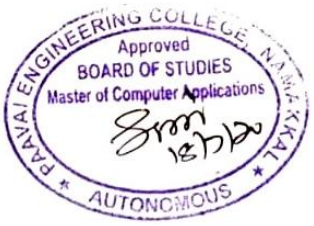
At the end of the course the students would be able to

- apply cryptographic algorithms for encrypting and decryption for secure data transmission
- understand the importance of digital signature for secure e-documents exchange
- understand the program threats and apply good programming practice
- get the knowledge about the security services available for internet and web applications
- understand data vulnerability and trusted computing

REFERENCES

1. Amerding, T (2012) 'The 15 worst data security breaches of the 21st Century', *CSO*, 15.
2. Itami, H. and Roehl, T. (1987) Mobilizing Invisible Assets, Harvard, Harvard University Press.
3. Pew Research Center (2012) the State of the News Media 2012.
4. Protalinski, E. (2013) 'Belgian rail firm SNCB Europe sees 1.5m customer details leaked, but fails to take responsibility'.

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	1	2	1	2	-	-	-	-	-	-	-	1	1	2
CO2	1	3	2	1	3	-	-	-	-	-	-	2	2	2
CO3	2	2	2	2	3	-	-	-	-		2	2	1	2
CO4	2	2	2	2	-	-	-	3	-	-	-	-	2	2
CO5	1	2	1	3	-	-	-	2	-	-	-	2	2	3



CA20153 OBJECT ORIENTED SOFTWARE ENGINEERING 3 0 0 3

COURSE OBJECTIVES

To enable the students to

- understand the phases in a software project and activities in project management
- comprehend the purpose of different UML diagrams
- understand the major considerations in collecting, documenting and analyzing project requirements
- cognize the activities in the crucial phase of system design
- introduce the basics and necessary of software testing

UNIT I INTRODUCTION 9

Introduction to Software Engineering-Concepts; Software engineering development activities; Managing software development

UNIT II REQUIREMENTS AND ANALYSIS 9

Requirements Elicitation - Concepts - Activities & Managing Requirements Elicitation; Analysis: Concepts - Analysis Activities - Analysis Model

UNIT III MODELING WITH UML 9

UML Diagrams: Use Case Diagrams - Class Diagrams – Interaction Diagrams - State Machine Diagrams - Activity Diagrams; Modeling Concepts - Diagram Organization - Diagram Extension

UNIT IV SYSTEM DESIGN 9

Decomposing the System; Addressing Design Goals; Reusing Patterns; Specifying Interfaces; Mapping Models to Code

UNIT V TESTING PRINCIIPLES AND AXIOMS 9

Testing as a Process – Testing Axioms –Software Testing Principles; Origins and Cost of Defects; Defect Classes and Examples; Developer/Tester Support of Developing a Defect Repository; Defect Prevention Strategies.

TOTAL PERIODS 45

COURSE OUTCOMES

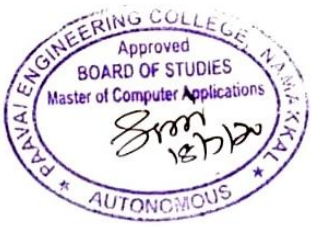
At the end of the course the students would be able to

- conceive the basics concepts of object oriented software engineering
- use knowledge documentation for object oriented software engineering using UML
- analyze problem and alternative solutions using object oriented software engineering approach
- manage software process and build software engineering teams based on engineering approach
- identify and analyze error, bug and defect for programs

REFERENCES

1. Bernd Bruegge, Alan H Dutoit, "Object-Oriented Software Engineering Using UML, Patterns, and Java", 3rd Edition, 2017
2. Philippe Kruchten, "The Rational Unified Process: An Introduction", 3rd Edition, 2017
3. Mike Cohn, "Succeeding with Agile: Software Development Using Scrum", 1st Edition, 2018.
4. Grady Booch, James Rumbaugh and Ivar Jacobson, "The Unified Modeling Language User Guide", Addison -Wesley Longman, USA, 2nd Edition, 2017.
5. Paul C. Jorgensen, "Software Testing: A Craftsman's Approach", Fourth Edition, CRC Press, 2013

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	1	2	1	2	-	-	-	-	-	-	-	1	1	2
CO2	1	2	2	2	3	-	-	-	-	-	-	2	1	2
CO3	2	2	2	2	2	-	-	-	-		2	1	1	3
CO4	2	1	2	2	-	-	-	2	-	-	-	-	3	2
CO5	1	2	1	2	-	-	-	2	-	-	-	2	2	2



CA20154 ADVANCED DATABASE 3 0 0 3

COURSE OBJECTIVES

To enable the students to

- learn the modeling and design of databases
- acquire knowledge on parallel and distributed databases and its applications
- study the usage and applications of object oriented database
- import the principles of intelligent databases
- understand the usage of advanced data models

UNIT I PARALLEL AND DISTRIBUTED DATABASES 9

Database System Architectures - Centralized and Client-Server Architectures – Server System Architectures; Parallel Systems- Distributed Systems – Parallel Databases: I/O Parallelism – Inter and Intra Query Parallelism – Inter and Intra operation Parallelism – Design of Parallel Systems; Distributed Database Concepts – Distributed Data Storage–Distributed Transactions–Commit Protocols – Concurrency Control – Distributed Query Processing; Case Studies.

UNIT II OBJECT AND OBJECT RELATIONAL DATABASES 9

Concepts for Object Databases: Object Identity – Object structure; Type Constructors – Encapsulation of Operations; Methods – Persistence; Type and Class Hierarchies – Inheritance; Complex Objects – Object Database Standards, Languages and Design; ODMG Model – ODL –OQL; Object Relational and Extended – Relational Systems: Object Relational features in SQL/Oracle; Case Studies.

UNIT III INTELLIGENT DATABASES 9

Active Databases - Syntax and Semantics (Starburst, Oracle, DB2); Taxonomy- Applications- Design Principles for Active Rules; Temporal Databases- Overview of Temporal Databases- TSQL2- Deductive Databases; Logic of Query Languages; Datalog - Recursive Rules-Syntax and Semantics of Datalog Languages; Implementation of Rules and Recursion- Recursive Queries in SQL; Spatial Databases- Spatial Data Types- Spatial Relationships - Spatial Data Structures- Spatial Access Methods- Spatial DB Implementation.

UNIT IV ADVANCED DATA MODELS 9

Mobile Databases - Location and Handoff Management - Effect of Mobility on Data Management – Location Dependent Data Distribution; Mobile Transaction Models; Concurrency Control - Transaction Commit Protocols; Multimedia Databases- Information Retrieval; Data warehousing- Data Mining- Text Mining.

UNIT V EMERGING TECHNOLOGIES 9

XML Databases- XML-Related Technologies-XML Schema- XML Query Languages-Storing XML in Databases -XML and SQL- Native XML Databases; Web Databases- Geographic Information Systems; Biological Data Management- Cloud Based Databases- Data Storage Systems on the Cloud; Cloud Storage Architectures- Cloud Data Models - Query Languages; Introduction to Big Data –Storage - Analysis.

TOTAL PERIODS 45

COURSE OUTCOMES

At the end of the course the students would be able to

- select the appropriate high performance database like parallel and distributed database
- model and represent the real world data using object oriented database
- design a semantic based database to meaningful data access
- embed the rule set in the database to implement intelligent databases
- represent the data using XML database for better interoperability

REFERENCES

1. R. Elmasri, S.B. Navathe, "Fundamentals of Database Systems", Fifth Edition, Pearson Education/Addison Wesley, 2017.
2. Thomas Cannolly and Carolyn Begg, "Database Systems, A Practical Approach to Design, Implementation and Management", Third Edition, Pearson Education, 2017.
3. Henry F Korth, Abraham Silberschatz, S. Sudharshan, "Database SystemConcepts", Fifth Edition, McGraw Hill, 2017.
4. C.J.Date, A.Kannan and S.Swamynathan,"An Introduction to Database Systems", Eighth Edition, Pearson Education, 2016.
5. Raghu Ramakrishnan, Johannes Gehrke, "Database Management Systems", McGraw Hill, 3rd Edition 2016.

Mapping of Courses Outcomes with Programme Outcomes: **(1/2/3 indicates strength of correlation) 3-Strong, 2-Medium, 1-Week**														
COs	**Programme Outcomes (POs)**													
CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	2	2	3	2	-	-	-	-	-	-	-	1	2	2
CO2	1	2	2	2	3	-	-	-	-	-	-	2	1	2
CO3	2	2	1	2	2	-	-	-	-		2	2	2	2
CO4	2	1	2	2	-	-	-	1	-	-	-	-	2	2
CO5	1	1	3	2	-	-	-	2	-	-	-	1	1	3