



Faculty of Engineering & Technology

Syllabus

for

Master of Computer Applications (MCA)

(2 Years PG Program)

(w.e.f. 2023-24)

List of Contents

S. No	Contents	Page No.
1.	Introduction	3
2.	Program's Mission and Objectives	3
3.	Relevance of Program with Jagan Nath University, Jaipur Mission and Goals	5
4.	Instructional Design	6
5.	Expected Program Outcomes	7
6.	Curriculum design 6.1 Program structure and Detailed syllabus	8

MCA Program

1. Introduction

Master of Computer Applications (MCA) is a postgraduate program with industry-oriented curriculum containing project based & experiential learning. The MCA program is offered in semester mode as per CBCS framework. The program is designed towards current and emerging issues in IT and business. This programme aims to provide a sound theoretical background and excellent practical exposure to the students to meet the demands of the information technology industry. The course structure for the program includes Core Courses, Core Elective Courses, and Skill Enhancement Courses (SEC).

2. Program Mission and Objectives

Jagan Nath University, Jaipur has been established by an Act of the Rajasthan State Legislature (Act No. 19 of 2008), approved under section 2(f) by the UGC Act, 1956, accredited by **NAAC (2nd Cycle) with "A" grade** in 2022 and is member of the Association of Indian Universities (AIU).

The University is offering **UG and PG** programs in the areas of Engineering & Technology, Architecture, Management, Law, Agriculture, Pharmacy, Physiotherapy, Paramedical Sciences, Education, Arts and Vocational Studies. The **Ph.D.** programs are offered in selected disciplines. The University has approval from various Statutory Regulatory Authorities (CoA, BCI, NCTE and PCI) for programs in respective disciplines.

The University has evved a **Comprehensive Academic System** which incorporates the latest guidelines of NEP-2020 including CBCS, outcome based assessments, skill development, industry partnerships, histic eco-system, research, extension activities with focus on promoting quality higher education and employability among students.

The Master of Computer Applications (MCA), a postgraduate program is having industry-oriented curriculum containing project based & experiential learning. Program stresses the application of theory and computing principles through case studies, presentations and practical assignments. Last semester is devoted to Project. The Program is designed towards current and emerging issues in CS & IT.

The objectives of the Master of Computer Applications (MCA) program are:

- To prepare graduates who will create systems through software development to sve problems in Industry domain areas.

- To prepare graduates who will perform both as an individual and in a team through good analytical, design and implementation skills.
- To prepare graduates who will be lifelong learners through continuous professional development.
- To apply current technologies and research to create systems for solving industry oriented problems.
- To develop the abilities to face the changing trends and career opportunities in computer application.
- To embed strong human values and professional ethics for becoming social responsibilities.

3. Relevance of Program with Jagannath University, Jaipur Mission and Goals

The University Vision and Mission Statements read as under:

Vision

To develop the University as a Centre of Excellence for higher education and research committed towards quality education, skill development, industry integration and holistic eco-system for global competencies among youth and sustainable development of the Nation.

Mission

The University aspires to achieve its vision by:

- I. Innovative, job-oriented and professional academic programs for capacity building in view of the emerging trends of the economy.
- II. Enable students equipped with knowledge and competences to perform successfully in modern organizations in India and abroad.
- III. Training students in generic and life skills in addition to core discipline subjects to enhance employability in job market and for entrepreneurship.
- IV. Engaging students and faculty in research, extension services, consultancy, community development projects, curricular and extra-curricular activities for holistic education.
- V. Promoting use of digital technologies and self-learning resources like MOOCs, Coursera, Virtual Labs, Online Resources, Self-Learning, etc. for enriching information and knowledge.
- VI. Inculcating a culture of excellence among students and faculty.

- VII. Developing a sense of ownership and pride among employees to achieve organizational targets as well as their personal goals.
- VIII. Developing curriculum, training and internship programs to enhance global competencies of absvents.
- IX. Blending skill, entrepreneurship and capacity building for sustainable development coherent with environmental and economic sustainability.

One of the missions of the Jagan Nath University is to provide quality education to the youth, particularly from rural sectors, which face socio-economic barriers to higher education, and enable them to achieve their career aspirations and to encourage student enrlnment from among the learners and provide an accessible and affordable mode of education. There is need to offer a comprehensive education model that integrates theoretical knowledge with practical skills to enhance learners' employability in the job market and entrepreneurship. The Master of Applications (MCA) program aids learners in enhancing their IT skills, enabling them to acquire employment opportunities. Overall, the mission of the MCA program is to provide learners with a comprehensive education in computer applications and information technogy, enabling them to enhance their IT skills and acquire employment opportunities, while also fostering their intellectual, personal, and professional growth.

4. Instructional Design

The Curriculum & Program structure has been designed as per AICTE guidelines for MCA Program. Also the general guidelines as per UGC and Choice Based Credit System (CBCS) have been flowed in framing curriculum design and the program structure for the program. The curriculum and syllabus are duly approved by the Board of Studies, and the University Academic Council, consisting of professionals from academia and industry.

5. Expected Program Outcomes

The following are the expected program outcomes:

- PLO1.** Apply knowledge, skills, and current tools, recent computing technologies of Computer Science innovatively to different applications.
- PLO2.** Enhance skills to design and conduct experiments, as well as to analyze and interpret data and address the research gaps to produce solutions with the help of tools, technology and products.
- PLO3.** Enhance critical thinking by acquiring the skills in modern techniques, methodologies and tools to be innovative and creative. Enhance critical thinking by acquiring the skills in modern techniques, methodologies and tools to be innovative and creative. Analytical reasoning refers to the ability to look at information, be it qualitative or quantitative in nature, and discern patterns within the information.
- PLO4.** Understand the contemporary research, security issues in the different areas of computer science and solve the real world problems.
- PLO5.** An ability to identify, analyze, design, develop, implement and integrate software and hardware based computer systems.
- PLO6.** An ability to communicate effectively, express /present ideas in an impressive and professional manner, both in written and verbal forms.
- PLO7.** An ability to understand leadership and entrepreneurship qualities.
- PLO8.** An ability to work in multidisciplinary and multicultural environment, become entrepreneur.
- PLO9.** An ability to understand health, ethical, legal, financial, and professional responsibilities.
- PLO10.** To recognize the need for self-motivation and ability to engage in lifelong learning through continuing education, research and professional development.

6. Curriculums Design: Programs Offered

The proposed framework for Masters of Computer Applications (MCA) is as under:

**Program Structure for 2 Years Master of Computer Applications (MCA) Program as per
Choice Based Credit System (CBCS)**

FIRST SEMESTER						
THEORY PAPERS			Marks Allocation			
Code	Subject/Paper	Type	IA	EA	Total	Credits
MCA 101	Programming with C	Core	30	70	100	4
MCA 102	Data Structure & Algorithms	Core	30	70	100	4
MCA 103	Computer Organization & Architecture	Core	30	70	100	4
MCA 104	Data Communications and Networking	Core	30	70	100	4
MCA 105	Optimization Techniques	Core	30	70	100	4
PRACTICALS/VIVA-VOCE			Sessional	Practical	Total	Credits
MCA 106	Programming in C Lab	Practical	25	25	50	2
MCA 107	Data Structure & Algorithms Lab	Practical	25	25	50	2
MCA 108	Network Programming Lab	Practical	25	25	50	2
TOTAL			225	425	650	26
SECOND SEMESTER						
THEORY PAPERS			Marks Allocation			
Code	Subject/Paper	Type	IA	EA	Total	Credits
MCA 201	Object Oriented Programming using C++	Core	30	70	100	4
MCA 202	Database Management System	Core	30	70	100	4
MCA 203	Operating Systems	Core	30	70	100	4
MCA 204	Software Engineering	Core	30	70	100	4
MCA 205	Management Information System	Core	30	70	100	4
PRACTICALS/VIVA-VOCE				Practical	Total	Credits
MCA 206	Object Oriented Programming using C++ Lab	Practical	25	25	50	2
MCA 207	Database Management System Lab using SQL	Practical	25	25	50	2

MCA 208	Software Engineering Lab	Practical	25	25	50	2
TOTAL			225	425	650	26
THIRD SEMESTER						
THEORY PAPERS			Marks Allocation			
Code	Subject/Paper	Type	IA	EA	Total	Credits
MCA 301	Python Programming	Core	30	70	100	4
MCA 302	Cryptography and Network Security	Core	30	70	100	4
MCA 303	Artificial Intelligence	Core	30	70	100	4
MCA 304	PHP	Core	30	70	100	4
	Elective(Select any one)					
MCA 305A	E-Commerce	Elective	30	70	100	4
MCA 305B	Big Data Analytics	Elective	30	70	100	4
MCA 305C	Internet of Things	Elective	30	70	100	4
PRACTICALS/VIVA-VOCE			Major	Practical	Total	Credits
MCA 306	Python Programming Lab	Practical	25	25	50	2
MCA 307	PHP Lab	Practical	25	25	50	2
MCA 308	Seminar	Practical	25	25	50	2
TOTAL			225	425	650	26
FOURTH SEMESTER						
THEORY PAPERS			Marks Allocation			
Code	Subject/Paper	Type	IA	EA	Total	Credits
MCA 401	Programming in Java	Core	30	70	100	4
MCA 402	Machine Learning	Core	30	70	100	4
	Elective(Select any one)					
MCA 403.A	Cloud Computing	Elective	30	70	100	4
MCA 403.B	Software Project Management	Elective	30	70	100	4
MCA 403.C	Data mining and Were house	Elective	30	70	100	4
MCA 404	Research and Publication Ethics	Skill	30	70	100	4
PRACTICALS/VIVA-VOCE			Minor	Practical	Total	Credits
MCA 405	Java Programming Lab	Practical	25	25	50	2

MCA 406	Machine Learning Lab	Practical	25	25	50	2
MCA 407	Major Project	Practical	75	75	150	6
TOTAL			245	405	650	26

IA=INTERNAL ASSESSMENT, EA=EXTERNAL ASSESSMENT

The total number of the credits of the MCA Program is 104

6.1 Program structure and detailed syllabus

FIRST SEMESTER

THEORY PAPERS			Marks Allocation			
Code	Subject/Paper	Type	IA	EA	Total	Credits
MCA 101	Programming with C	Core	30	70	100	4
MCA 102	Data Structure & Algorithms	Core	30	70	100	4
MCA 103	Computer Organization & Architecture	Core	30	70	100	4
MCA 104	Data Communications and Networking	Core	30	70	100	4
MCA 105	Optimization Techniques	Core	30	70	100	4
<i>PRACTICALS/VIVA-VOCE</i>			Sessional	Practical	Total	Credits
MCA 106	Programming in C Lab	Practical	25	25	50	2
MCA 107	Data Structure & Algorithms Lab	Practical	25	25	50	2
MCA 108	Network Programming Lab	Practical	25	25	50	2
TOTAL			225	425	650	26

MCA 101 Programming with C

Course Objective:

- To Gain a thorough understanding of the fundamentals of C programming
- To Implement the Ideas of algorithm and computational procedure, editing and executing 'C'- programs in Linux and Windows.
- To Speculating algorithmic solutions to problems and implementing algorithms in C
- To Enlightening Library functions, branching and decision making statements.
- To understand modular programming and recursive solution formulation.

Syllabus

UNIT: 1

Introduction: GCC, Using MAKE Utility, GDB, C Basics: History of C, Characteristics of C, C Program Structure, Variables, Defining Global Variables, Printing Out and Inputting Variables, Constants

UNIT: 2

Arithmetic Operations, Comparison Operators, Logical Operators, Order of Precedence, Conditionals (The if statement, The ? operator, The switch statement) Looping and Iteration (The for statement, The while statement, The do-while statement, break and continue)

UNIT: 3

Arrays and Strings (Single and Multi-dimensional Arrays, Strings) Functions (Function Prototyping, passing parameters, returning values, recursion) Storage classes (auto, extern, static, register)

UNIT: 4

Further Data Types: Defining New Data Types, Structures, Unions, Type-Casting, Enumerated Types, Low Level Operators and Bit Fields (Bitwise Operators, Bit Fields)

UNIT: 5

Pointers: Pointers arithmetic and Arrays, const pointers, void pointers, near, far and huge pointers Dynamic Memory Allocation and Dynamic Structures: (malloc, calloc and realloc; size of free

UNIT: 6

introduction to Linked Lists and dynamic 2- dimensional arrays), Advanced Pointer Topics: (Pointers to Pointers, Pointer to array, Array of pointers, Command line input, Pointers to a Function, Implementing Callbacks)

UNIT: 7

The C Preprocessor: #define, #undef, #include, #if -- Conditional inclusion, Other Preprocessor Commands) C,

UNIT: 8

Linux and Standard Libraries: (Advantages of using Linux with C, Using Linux System Calls and Library Functions) Integer Functions, Random Number

UNIT: 9

String Conversion, Searching and Sorting: <stdlib.h> Mathematics: <math.h> (Math Functions, Math Constants).

UNIT: 10

Input and Output (I/O): stdio.h, Reporting Errors (perror(), errno, exit()) Streams (Predefined Streams, Redirection) Basic I/O (Formatted I/O, printf, scanf),

UNIT: 11

String Handling: <string.h> (Basic String Handling Functions and safety issues, String Searching), Character conversions and testing: ctype.h, Files Character and Line Based I/O, Formatted I/O, Block I/O,

UNIT: 12

File Positioning, Status Functions, Deletion and Renaming, Temporary Files

UNIT: 13

File Accessibility and Directories (access, stat, chmod, chown..., chdir, chroot...)

UNIT: 14

Process Contr: (Running Linux Commands from C, fork (), the exec family, wait(), exit())

UNIT: 15

Thread creation-a simple implementation.

TEXT BOOKS:

1. Yashwant Kanetkar, "Let us C", BPB Publications, 2002.
2. Programming in ANSI C by M. Balaguruswami, Tata McGraw Hill, 2008
3. B. Kernighan and D. Ritchie, "The ANSI C Programming Language", PHI., 2000

REFERENCES:

1. Yashwant Kanetkar, "Pointers in C", BPB Publications, 2002.
2. Paul Deitel and Harvey Dietel, "How to Program", PHI, 6th Ed., 2010.
3. Rama N. Reddy and Car A. Ziegler, "C Programming for Scientist and Engineers with Applications", Jones and Bartlet, 2010.

COURSE OUTCOMES

At the end of the course, a student will be able to

CO:1. Describe the concepts of problem solving through C Programming.

CO:2. Apply the programming concepts in C language

CO:3. Understand the diversified situations using C language.

CO:4. Understand the concept of Input- Output and string handling in C.

CO:5. Experiment File Directories and Process-thread Contr

MCA 102: Data Structure & Algorithms

Course Objective:

- To introduce the basic concepts of data structures and algorithms.
- To understand concepts about searching and sorting techniques
- To understand basic concepts about stacks, queues, list, trees and graphs.
- To enable them to write algorithms for solving problems with the help of fundamental data structures

Syllabus

UNIT: 1

Introduction: Data Structures, data structure operations, complexity, Asymptotic Notation, Time/Space trade-off. Linear Lists: Arrays, address calculation in single and multidimensional arrays, operations on array

UNIT: 2

Sequential search, Binary Search and their complexity analysis.

UNIT: 3

Stacks: Primitive operations, implementation of stacks using Arrays, applications of stacks arithmetic expression conversion and evaluation.

UNIT: 4

Queues: Primitive operations; Implementation of queues using Array, applications of linear queue, circular queue and double ended queue (DEQUE).

UNIT: 5

Linked lists and its operations: linked list, representation of link list in memory

UNIT: 6

Traversing a link list, insertion into a link list, deletion from a link list, header link list, two way link lists.

UNIT: 7

Trees: Definition of tree, Binary tree and related terms

UNIT: 8

Application of binary tree, Tree Traversals

UNIT: 9

Threaded tree, Binary Search Tree, heap, heap sort, General trees.

UNIT: 10

Graph: introduction, sequential representation of Graphs,

UNIT: 11

Adjacency matrix, path matrix, operations on graphs

UNIT: 12

Traversing a Graph, Warshall's algorithm.

UNIT: 13

Sorting Techniques: Selection, Insertion,

UNIT: 14

Bubble, Merge, Quick

UNIT: 15

Radix sort, searching and hashing.

TEXTBOOKS:

1. Schaum Series, “Introduction to Data Structures”, TMH.
2. R.B. Patel, “Expert Data Structures with C”, Second Edition, Khanna Book publishing Co (P) Ltd.

REFERENCE BOOKS:

1. Tenenbaum, “Data Structure using C++”, PHI.
2. Chattopadhyay S., Dastidar d G.and Chattopadhyay Matangini., “Data Structure through C language”, BPB publications.

COURSE OUTCOMES

At the end of the course, a student will be able to

CO1: Assess how the choice of data structures and algorithm design methods impacts the performance of programs .and apply Search problem in Data Structure.

CO2: Understand basic data structures such as arrays, linked lists, stacks and queues.

CO3: Implement different types of trees and apply them to problem sutions.

CO4: Implement Graph search and traversal algorithms and determine the time and computation complexity.

CO5: Apply the different sorting techniques to sort the data.

MCA 103: Computer Organization & Architecture

Course Objective:

- To understand the structure, function and characteristics of computer systems.
- To understand the design of the various functional units and components of computers.
- To identify the elements of modern instructions sets and their impact on processor design.

UNIT: 1

Register Transfer and Micro operation: Register transfer language, register transfer

UNIT: 2

Bus and memory transfer, arithmetic micro operations

UNIT: 3

Logic micro operations, shift micro operations,

UNIT: 4

Instruction codes, computer registers, computer instructions

UNIT: 5

Timing & contr, instruction cycle, Register-Memory reference instructions

UNIT: 6

Design of basic computer, design of accumulator logic.

UNIT: 7

Introduction to CPU, general registers organization

UNIT: 8

Stack organization, instruction formats, and addressing modes,

UNIT: 9

Pipeline: Parallel Processing, instruction pipeline

UNIT: 10

Contr memory, address sequencing.

UNIT: 11

Input-Output Organization: Peripheral devices, input-output interface

UNIT: 12

Asynchronous data transfer, modes of data transfer, priority interrupt, direct memory access, input-output processor.

UNIT: 13

Memory hierarchy, main memory, RAM & ROM chips

UNIT: 14

Auxiliary memory, associative memory

UNIT: 15

Cache memory, Virtual memory.

TEXT BOOKS:

1. Mano M, “Computer System and Architecture”, Pearson, 3rd Ed., 2009
2. Stallings W, “Computer Organization & Architecture”, PHI, 8th Ed., 2010.

REFERENCES:

1. Malvino, “Digital Computer Electronics: An Introduction to Microcomputers”, McGraw Hill, 1993.
2. Andrew S. Tanenbaum, “Structured Computer Organization”, PHI, 5th Ed., 2006.
3. P. V. S Rao, “Computer System Architecture”, PHI, 5th Ed., 2009.

COURSE OUTCOMES

At the end of the course, a student will be able to

CO1: Describe the functional architecture of computing systems.

CO2: Understand Computer Organization and Design

CO3: Explain CPU design and pipelined contr units

CO4: Show working of Input-Output Organization.

CO5: Differentiate between different types of Memories.

MCA 104: Data Communications and Networking

Course Objectives:

- To covers the theory and practice of data communication between computing devices.
- To introduce the network architecture and topogy, Basics of networking and protocs, OSI network layered models and Application layer protocs.

Syllabus

UNIT: 1

Goals and Applications of Networks, Network structure and architecture, the OSI reference model, services, networks topogy. Transmission methods Synchronous & Asynchronous, Flow Contr, Error Contr, Error Detection methods.

UNIT: 2

Physical Layer: The Physical Layer, Theoretical Basis for Data Communication, Guided Transmission Media, Wireless Transmission, Communication Satellites Digital Signal Encoding Formats – NRZ-L, NRZI, bipolar-AMI

UNIT: 3

Manchester, Differential Manchester, Digital Modulation – ASK, FSK, PSK, PSK, Digitization – Sampling Theorem, PCM, DM, Analog Modulation – Introducing AM, FM, PM, The Mobile Telephone System.

UNIT: 4

The Data Link Layer: Data Link Layer Design Issues, Error Detection and Correlation, Flow Contr Protocs, Stop-and-wait Flow Contr,

UNIT: 5

Sliding – Window Flow Contr, Error Contr, Stop-and-wait ARQ, Go-back-N, Selective-repeat, Example of Data Link Protocs-HDLC
Medium access sub layer: Channel allocations

UNIT: 6

ALOHA Protocs, Carrier Sense Multiple Access Protocs, CSMA with Clision Detection, Clision free protocs, Ethernet, wireless LANs, Blue Tooth, Data Link Layer Switching

UNIT: 7

Network Layer: Point-to-Point network, routing algorithms,

UNIT: 8

Congestion contr, internetworking, Quality Contr, Internetworking

UNIT: 9

The Network Layer in the Internet, IP packet, IP addresses, IPv6

UNIT: 10

Transport Layer: Design Issue, connection management

UNIT: 11

TCP window management, User Datagram Protoc

UNIT: 12

Transmission Contr Protoc and Performance Issues.

UNIT: 13

Application Layer: DNS, Electronic Mail,

UNIT: 14

WWW, MUTIMEDIA.

UNIT: 15

Network Security: Cryptography and Compression Techniques.

TEXT BOOKS:

1. Forouzan, “Data Communication and Networking”, TMH, 4th Edition.
2. A.S. Tanenbaum, “Computer Networks”, PHI, 4th Edition.
3. W. Stallings, “Data and Computer Communication”, Macmillan Press.
4. Comer, “Computer Networks and Internet”, PHI.
5. Comer, “Internetworking with TCP/IP”, PHI.

REFERNCES:

1. W. Stallings, “Data and Computer Communication”, McMillan.
2. J. Martin, “Computer Network and Distributed Data Processing”, PHI.
3. W. Stallings, “Local Networks”, McMillan.

COURSE OUTCOMES

At the end of the course, a student will be able to

CO1. Understand the functions of the different layer of the OSI Protoc Classify their res.

CO2. Apply the functionasof wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs) describe the function of each block.

CO3. Understand the Data Link Layer

CO4. Understand a given problem related TCP/IP protoc developed the network programming.

CO5. Implement Configure DNS DDNS, TELNET, EMAIL, File Transfer Protoc (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls using open source available software and tos.

MCA 105 Optimization Techniques

Course Objectives:

- To understand the theory of optimization methods and algorithms developed for solving various types of optimization problems
- To develop and promote research interest in applying optimization techniques in problems of Engineering and Technology
- To apply the mathematical results and numerical techniques of optimization theory to concrete Engineering problems.

Syllabus

UNIT: 1

Introduction and Classification,

UNIT: 2

Basic concept of optimization

UNIT: 3

Mathematical formulation of optimization problems.

UNIT: 4

Applications and Classification of Optimization Problems - single variable problems, Multivariable problems without constraints

UNIT: 5

Multivariable problems with constraints, Maximization and minimization problems. Single Variable Optimization Necessary and sufficient conditions for optimum; interpolation method quadratic.

UNIT: 6

Region elimination methods-internal halving, Fibonacci.

UNIT: 7

Multivariable Optimization, Optimization of Functions.

UNIT: 8

One Dimensional Search: Analytical Methods: classification, stationary points, direct substitution, constrained variation, penalty function, Lagrangian Multiplier, Kuhn-Tucker theorem.

UNIT: 9

Numerical methods general principles of numerical search, direction of search, final stage in search, direct search, pattern search.

UNIT: 10

Other Optimization Techniques Introduction to geometric,

UNIT: 11

Dynamic and integer programming and genetic algorithms.

UNIT: 12

Application of Geometric Programming: chemical engineering problems with degree of difficulty equal to zero or one with constraints.

UNIT: 13

Applications of Optimization

UNIT: 14

Optimization of staged and discrete processes.

UNIT: 15

Optimal shell-tube heat exchanger design. Optimal pipe diameter.

TEXT/REFERENCE BOOKS:

1. Hiller and Lieberman, Introduction to Operation Research (Seventh Edition) Tata McGrawHill Publishing Company Ltd
2. Ravindren Philips and Sberg, Operation Research Principles and Practice (Second Edition) John Wiley & Sons.

COURSE OUTCOMES

At the end of the course, the student will be able to:

CO1: Understand concept of optimization, Mathematical formulation of optimization problems;

CO2: Understand applications and Classification of Optimization Problems

CO3: apply the theory of optimization methods and algorithms to develop and for solving various types of optimization problems

CO4: Apply dynamic programming principle to Linear programming problems.

CO5: Describe the Applications of Optimization

MCA-106: Programming in C LAB

Course Objective:

- To understand a functional hierarchical code organization.
- To work with textual information, characters and strings.
- To work with arrays of complex objects.
- To understand a concept of functional hierarchical code organization.
- To understand a defensive programming concept.

LIST OF EXPERIMENTS:

- 1 Write a program to calculate the area & perimeter of rectangle.
- 2 Write a program to calculate the area and circumference of a circle for a given radius.
- 3 Write a program to calculate simple interest for a given principal/amount.
- 4 Write a C program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.
- 5 Write a program to find profit and loss (in percentage) of a given cost price and selling price.
- 6 Write a program to find out the maximum among the three given numbers.
- 7 Write C programs that use both recursive and non-recursive functions
To find the factorial of a given integer.
- 8 Write a program to print the list of first 100 odd number.
- 9 Write a program to calculate the sum of the digits of a number and display it in reverse order.
- 10 Write a program to generate a Fibonacci series.
- 11 Write a program to generate the flowing series:


```
*
* *
* * *
* * * *
* * * * *
```
- 12 Write a program to generate the flowing series:


```
0 1
0 1 0
0 1 0 1
0 1 0 1 0
```
- 13 Write a program using a function to check whether the given number is prime or not.
- 14 Write a program to check whether the given string is a palindrome or not.
- 15 Write a C program that uses functions to perform the flowing operations:
To insert a sub-string in to given main string from a given position.
- 16 Write a C program to determine if the given string is a palindrome or not.
- 17 Write a program to swap two variables a & b using pointers.
- 18 Write a program to enter a line of text from keyboard and store it in the file. User should enter file name.
- 19 Write a recursive program for tower of Hanoi problem
- 20 Write a C program that uses functions to perform the flowing:
 - Addition of Matrices.
 - Multiplication of Matrices.

21. Write a program to copy one file to other, use command line arguments.
22. Write a C program to reverse the first n characters in a file. (Note: The file name and n are specified on the command line.)
23. Write a program to perform the following operations on Strings without using String functions
 - To find the Length of String.
 - To concatenate two strings.
 - To find Reverse of a string.
 - To Copy one string to another string.
24. Write a Program to store records of a student in a student file. The data must be stored using Binary File. Read the record stored in "Student.txt" file in Binary code. Edit the record stored in Binary File. Append a record in the Student file.
25. Write a program to count the number of Lowercase, Uppercase numbers and special Characters present in the contents of File.

COURSE OUTCOMES

At the end of the course, a student will be able to

CO:1. Analyze a problem and develop an algorithm to solve it

CO:2. Show the use of the C programming language to implement various algorithms, and producing the basic concepts and terminology of programming in general

CO:3. Solve the problems using features of the C language.

CO:4. Implement the program, compile, debug, recompile and run it.

CO:5. Evaluate and Identify tasks in which the numerical techniques learned are applicable and apply them to write programs, and hence use computers effectively to solve the task.

MCA 107: DATA STRUCTURES & ALGORITHM LAB

Course Objective:

- To develop skills design and analyze simple linear and nonlinear data structures.
- To identify and apply the suitable data structure for the given real world problem.
- To gain knowledge in practical applications of data structures.

List of Experiments

1. Write a program to insert an element at desire position in the array.
2. Write a program to delete an element at desire position from the array.
3. Write a program to replace an element at desire position in the array.
4. Write a program to search (linear search) an element in the array.
5. Write a program to search (binary search) an element in the array.
6. Write a program to addition and multiply of two matrices.
7. Write a program to implementation of stack using array.
8. Write a program to implementation of queue using array.
9. Write a program to implementation link list.
10. Write a program that sorts the array through Bubble sort.
11. Write a program that sorts the array through Quick sort.
12. Write a program that sorts the array through Merge sort.
13. Write a program that sorts the array through Insertion sort.
14. Write a program to BST (binary search tree) addition, deletion and searching.

COURSE OUTCOMES

At the end of the course, a student will be able to

CO:1.Evaluate and analyze the time and space efficiency of the data structure

CO:2. Evaluate the appropriate data structure for given problem

CO:3. Implementation of data structures using C++.

CO:4. Sve the problems of searching and sorting using techniques.

CO:5. Sve the problems using Tree data structure .

MCA 108: Network Programming Lab

Course Objectives:

- To introduce Network related commands and configuration files in Linux Operating System.
- To introduce tools for Network Traffic Analysis and Network Monitoring
- To practice Network Programming using Linux System Calls.
- To design and deploy Computer Networks.

List of Experiments:

1. Study of Different Type of LAN & Network Equipments.
2. Study and Verification of standard Network topologies i.e. Star, Bus, Ring etc.
3. LAN installations and Configurations.
4. Write a program to implement various types of error correcting techniques.
5. Write a program to implement various types of framing methods.
6. Write two programs in C: hello_client and hello_server
 - a. The server listens for, and accepts, a single TCP connection; it reads all the data it can from that connection, and prints it to the screen; then it Closes the connection
 - b. The client connects to the server, sends the string "Hello, world!", then closes the connection
7. Write an Echo_Client and Echo_server using TCP to estimate the round trip time from client to the server. The server should be such that it can accept multiple connections at any given time.
8. Repeat Exercises 6 & 7 for UDP.
9. Repeat Exercise 7 with multiplexed I/O operations.
10. Simulate Bellman-Ford Routing algorithm in NS2

Course Outcomes:

At the end of the course, the student will be able to:

CO1: Apply knowledge of different techniques of error detection and correction to detect and solve error bit during data transmission.

CO2: Understand and building the skills of routing mechanisms.

CO3: Explain how a collision occurs and how to solve it.

CO4: Explain familiar with network tools and network programming.

CO5: Adapt with the basic protocols of computer networks, and how they can be used to assist in network design and implementation.

SECOND SEMESTER

THEORY PAPERS			Marks Allocation			
Code	Subject/Paper	Type	IA	EA	Total	Credits
MCA 201	Object Oriented Programming using C++	Core	30	70	100	4
MCA 202	Database Management System	Core	30	70	100	4
MCA 203	Operating Systems	Core	30	70	100	4
MCA 204	Software Engineering	Core	30	70	100	4
MCA 205	Management Information System	Core	30	70	100	4
PRACTICALS/VIVA-VOCE				Practical	Total	Credits
MCA 206	Object Oriented Programming using C++ Lab	Practical	25	25	50	2
MCA 207	Database Management System Lab using SQL	Practical	25	25	50	2
MCA 208	Software Engineering Lab	Practical	25	25	50	2
TOTAL			225	425	650	26

MCA 201: Object Oriented Programming using C++

Course Objectives

- To introduce the basic concepts of Object Oriented Languages like class, object, data hiding, encapsulation, and abstraction.
- To understand and implement concepts like message passing, inheritance, polymorphism, exception handling and generic programming.

Syllabus

UNIT: 1

OOP Paradigm: Comparison of Programming paradigms, Characteristics of Object-Oriented Programming Languages, Object-based programming languages,

UNIT: 2

Brief History of C++, Structure of a C++ program, Difference between C and C++ - cin, cout, new, delete operators

UNIT: 3

Oops concepts: Objects, Classes, Encapsulation, Data Abstraction, Inheritance, Polymorphism, Dynamic Binding, Message Passing,

UNIT: 4

Abstract data types, Class Component, Object & Class, Constructors Default and Copy Constructor, Assignment operator deep and shallow copying, Access modifiers – private, public and protected. Implementing Class Scope resolution operator,

UNIT: 5

Working with Friend Functions, Using Static Class members. Understanding Compile Time Polymorphism function overloading Rules of Operator Overloading (Unary and Binary) as member function/friend function.

UNIT: 6

Implementation of operator overloading of Arithmetic Operators, Overloading Output/Input, Prefix/Postfix Increment and decrement Operators, Overloading comparison operators, Assignment, subscript and function call Operator, concepts of namespaces.

UNIT: 7

Inheritance: Inheritance, Types of Inheritance, Abstract Classes, Ambiguity resolution using scope resolution operator and Virtual base class, Aggregation, composition vs classification hierarchies,

UNIT :8

Overriding inheritance methods, Constructors and Destructor in derived classes.

UNIT :9

Polymorphism: Polymorphism, Type of Polymorphism – compile time and runtime, Understanding Dynamic polymorphism: Pointer to objects, Virtual Functions pure virtual functions, Abstract Class.

UNIT :10

Generic Programming: Understanding Generic Functions with implementation of searching sorting

algorithm. Overloading of Function Templates. Understanding Class Templates using

UNIT :11

Implementation of Generic stack, linked lists: singly and doubly linked lists, Binary Search Tree basic operations.

UNIT :12

Standard Template Library: – Understanding Components of Standard Template Library, Working of Containers, Algorithms, Iterators and Other STL Elements. Implementation of Sequence and Associative containers for different Algorithms using their Iterator.

UNIT :13

Advanced Input/output Operations, Exception Handling and Manipulating strings, Using istream / ostream member functions, Using Manipulators, Creating Manipulator Functions

UNIT :14

Understanding Implementation of Files, Writing and Reading Objects

UNIT :15

Understanding of working and implementation of Exception Handling

TEXT/ REFERENCE BOOKS:

1. R. Venugopal, Rajkumar, and T. Ravishanker “Mastering C++”, TMH, 1997.
2. S. B. Lippman and J. Lajoie, “C++ Primer”, 3rd Edition, Addison Wesley, 2000.
3. Bruce Eckel, “Thinking in C++”, President, Mindview Inc., Prentice Hall, 2nd Ed.
4. D. Parsons, “Object Oriented Programming with C++”, BPB Publication.
5. Bjarne Stroustrup, “The C++ Programming Language”, Addison Wesley, 3rd Ed.
6. Steven C. Lawlor, “The Art of Programming Computer Science with C++”, Vikas Publication.
7. Schildt Herbert, “C++: The Complete Reference”, Tata McGraw Hill, 4th Ed., 1999.
8. Behrouz A. Forouzan, Richard F. Gilberg, Computer Science - A Structural Approach Using C++”, Cengage Learning, 2004.
9. Nell Dale, “C++ Plus Data Structure”, Jones and Bartlett, 4th Ed., 2010.
10. Nell Dale, Chips Weens, “Programming and Problem Solving with C++”, Jones and Bartlett, 5th Ed., 2010.

COURSE OUTCOMES

At the end of the course, the student will be able to:

CO1: Understand the object-oriented programming features in C++.

CO2: Apply C++ features to program design and implementation.

CO3: Solve the problems using Inheritance and polymorphism.

CO4: Implement generic programming language.

CO3: Understand I/O operations and file handling

Course Objective:

- To understand the different issues involved in the design and implementation of a database system.
- To study the physical and logical database designs, database modeling, relational, hierarchical, and network models
- To understand and use data manipulation language to query, update, and manage a Database
- To develop an understanding of essential DBMS concepts such as: database security, integrity, concurrency, distributed database, and intelligent database, Client/Server (Database Server), Data Warehousing.
- To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.

Syllabus

UNIT :1

Overview and History of DBMS. File System v/s DBMS. Advantage of DBMS

UNIT :2

Describing and Storing Data in a DBMS. Queries in DBMS

UNIT :3

Architecture and Structure of a DBMS. Data Models, Schemas & Instances. Overview of Data Models, hierarchical, Network & Relational model.

UNIT :4

Entity Relationship model: Overview of Data Design Entities, Attributes and Entity Sets, Relationship and Relationship Sets.

UNIT :5

Features of the ER Model- Key Constraints, Participation Constraints, Weak Entities, Class Hierarchies, Aggregation

UNIT :6

Conceptual Data Base and Design with ER Model- Entity v/s Attribute, Entity vs Relationship Binary vs Ternary Relationship and Aggregation v/s ternary Relationship Conceptual Design for a Large Enterprise.

UNIT 7

Relational Data Model and Language: Concepts and Constraints. Relational Algebra - select, project, set theoretic, join operations.

UNIT: 8

Overview of Relational Calculus. SQL - A Relational Database Language.

UNIT: 9

Data Definition commands, View and Queries, transaction commands, Specifying Constraints & Indexes in SQL., aggregate function, Null Values, nested sub queries, Joined relations. Triggers.

UNIT: 10

Schema Refinement: function dependencies & normalization for relational databases:

UNIT: 11

Functional dependencies, normal forms based on primary keys, (1NF, 2NF, 3NF & BCNF),

UNIT: 12

Lossless join and dependency preserving decomposition.

UNIT 13

Transaction, Concurrency & recovery techniques: Basic concept; ACID properties; transaction state;

UNIT: 14

Concurrency contr techniques, locking techniques, time stamp ordering, granularity of data items,

UNIT: 15

Recovery techniques: recovery concepts, database backup and recovery from catastrophic failures. Deadlock and Deadlock handling,

TEXT BOOKS:

1. Elmsari and Navathe, “Fundamentals of Database Systems”, Pearson Education, 5th Ed., 2006.
2. Korth, Silberschatz, “Fundamentals of Database System Concepts”, TMH, 6th Ed., 2010.
3. Desai, B., “An Introduction to Database Concepts”, Galgotia.
4. Sham Tickoo and Sunil Raina, “Oracle 11g with PL/SQL Approach”, Pearson, 2010.

REFERENCEBOOKS:

1. Date C. J., “An Introduction to Database Systems”, Narosa Publishing, 7th Ed., 2005.
2. S. K. Singh, “Database Systems: Concept, Design, and Applications”, Pearson’s Education, 1st Ed., 2008.
3. Kiffer, “Database Systems: An Application oriented Approach”, Pearson.
4. Ullman J. D., “Principals of database systems”, Galgotia .
5. Shio Kumar Singh, “Databases Systems Concepts, Design and Applications,” Pearson,2006.

COURSE OUTCOMES

At the end of the course, a student will be able to

CO1: Describe DBMS architecture & structure and overview of Data models.

CO2: Apply specification of the requirement design the databases using ER model.

CO3: Understand the Relational database design and apply the SQL Queries for access and manage the data.

CO4: Apply Schema Refinement for reducing the redundancy and remove the functional dependencies.

CO5: Implementation the transaction-processing system determines ACID properties.

MCA 203: Operating Systems

OBJECTIVE:

- To become familiar with the fundamental concepts of operating system.
- To become competent in recognizing operating systems features and issues.
- To provide the students with sufficient understanding of operating system design and how it impacts application systems design and performance.

Syllabus

UNIT: 1

Operating System Introduction, Re, Types of OS; Batch Systems, multi programming, time-sharing parallel, distributed and real-time systems, Buffering, Spoin, Operating system structure, Operating system components and services, System calls.

UNIT: 2

Processes: Process Concept, Process Scheduling, Operation on Processes, Cooperating Processes, Threads.

UNIT: 3

CPU Scheduling: Basic Concepts, Scheduling Criteria, Scheduling Algorithms, Multiple-Processor Scheduling, Real-Time Scheduling, Algorithm Evaluation

UNIT: 4

Interprocess Communication and Synchronization: Background, The Critical-Section Problem, Synchronization Hardware, Semaphores, Classical Problems of Synchronization, Critical Regions, Monitors, Message Passing.

UNIT: 5

Memory Management: Background, Logical vs. Physical Address space, swapping, Contiguous allocation, Paging, Segmentation, Segmentation with Paging.

UNIT: 6

Virtual Memory: Demand Paging, Page Replacement, Page-replacement Algorithms, Performance of Demand Paging, Allocation of Frames, Thrashing, Other Considerations, Demand Segmentation.

UNIT: 7

Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock, Combined Approach to Deadlock Handling.

UNIT: 8

Device Management: Techniques for Device Management, Dedicated Devices, Shared Devices, Virtual Devices

UNIT: 9

Device Characteristics - Hardware Consideration, Channels and Contr Units, Independent Device Operation, Buffering, Multiple Paths, Block Multiplexing, Device Allocation Consideration.

UNIT: 10

Secondary - Storage Structure: Disk Structure, Disk Scheduling, Disk Management, Swap-Space

Management, Disk Reliability, Stable storage implementation.

UNIT: 11

File-System Interface: File Concept, Access Methods, Directory Structure.

UNIT 12

File-System Implementation: Introduction, File-System Structure, Basic File System, Allocation Methods, Free-Space Management, Directory Implementation.

UNIT: 13

Security: The Security problem, Goals of protection, Access matrix, Authentication

UNIT: 14

Program threats, System threats, Intrusion detection, Cryptography.

UNIT: 15

Case Study : - Linux Operating System and Windows XP.

TEXT BOOKS:

1. Silberschatz and Galvin, “Operating System Concepts”, John Wiley, 8th Ed., 2009.
2. Milan Kovic., “Operating Systems”, Tata McGraw Hill, 2001
3. Deitel, Deitel and Choffnes, “Operating Systems”, Pearson ,3rd Edition

REFERENCE BOOKS:

1. Tannenbaum, “Operating Systems”, PHI, 4th Ed., 2000.
2. Madnick E. and Donovan J., “Operating Systems”, Tata McGraw Hill, 2001.
3. Flynn McHoes, “Operating System”, Cengage Learning, 2006.
4. Pbitra Pal Choudhury, “Operating System Principles and Design”, PHI, 2009.
5. Sibsankar Halder and Alex A. Aravind, “Operating System”, Pearson, 2009.
6. William Stallings, “Operating Systems Internals & Design Principles”, Pearson Education, 6th Ed., 2009.

COURSE OUTCOMES

At the end of the course, the student will be able to:

CO: 1.Describe familiarity with the fundamental concepts of operating systems and process scheduling.

CO: 2.Explain different Memory Management techniques.

CO: 3.Understand the Deadlock Handling and Device Management feature of OS.

CO:4. Understand the Secondary-Storage Structure and file management system of OS.

CO: 5. Demonstrate the concept of Security in operating system.

MCA 204: SOFTWARE ENGINEERING

Course Objective:

- To understand the importance, limitations and challenges of processes involved in software development.
- To focus on the types of various software models, various software design activities, and to learn cost estimation, software testing, Maintenance and debugging.

Syllabus

UNIT: 1

Software Engineering: Introduction and Definition of Software Engineering. Software Crisis,

UNIT: 2

Software Processes & Characteristics.

UNIT: 3

Software Process Models: Software development life cycle (SWDLC), Software development life cycle models:-Waterfall, Prototype, Evolutionary, RAD and Spiral Models.

UNIT: 4

Software Requirements analysis & specifications: Requirement analysis tasks, Analysis principles

UNIT: 5

Requirement elicitation techniques like FAST, QFD, Requirements analysis using DFD

UNIT: 6

Data dictionaries & ER Diagrams, Requirements documentation, Nature of SRS, Characteristics & organization of SRS.

UNIT: 7

Software Project Management Concepts: The Management spectrum, The People, The Problem, The Process, The Project.

UNIT: 8

Software Project Planning: Size Estimation like lines of Code & Function Count,

UNIT: 9

Cost Estimation Models, COCOMO, Risk Analysis.

UNIT: 10

Software Design: Design fundamentals, Effective modular design: Data architectural and procedural design

UNIT: 11

Design documentation. Function Oriented Design, Object Oriented Design.

UNIT: 12

Cohesion & Coupling: Cohesion & Coupling, Classification of Cohesiveness & Coupling.

UNIT: 13

Software Testing: Testing Fundamental, Characteristics of Testable Software, Test Characteristics

UNIT 14

Testing Techniques:-Black-box testing, White-box testing. Testing Strategies:-Unit Testing,

Integration and System Testing.

UNIT: 15

Software Maintenance: Management of Maintenance, Maintenance Process, Reverse Engineering, Software Re-engineering.

TEXT/ REFERENCE BOOKS:

1. R. S. Pressman, “Software Engineering – A practitioner’s approach”, McGraw Hill Int. Ed.
2. I. Sommerville, “Software Engineering”, Addison Wesley, 2004
3. Rajib Mall, “Fundamental of Software Engineering”, 3rd Edition, PHI Learning Private Limited
4. K. K. Aggarwal and Yogesh Singh, Software Engineering, New Age International Publishers
5. K. K. Aggarwal&Yogesh Singh, “Software Engineering”, 2nd Ed., New Age International, 2005.
6. James Peter, W. Pedrycz, “Software Engineering: An Engineering Approach”, John Wiley & Sons.
7. PankajJalote, “An Integrated Approach to Software Engineering”, Narosa, 3rd Ed., 2005.

COURSE OUTCOMES

At the end of the course, the student will be able to:

CO1: Understand the importance of the stages in the software life cycle and different Software Process Models.

CO2: Analysis of Software Requirements and preparing SRS.

CO3: Planning and Estimation of software project.

CO4: Understand software designing by applying the software engineering principles.

CO5: Perform Software testing, documentation and maintenance.

MCA 205 : Management Information System

Course Objectives:

- Get the knowledge about the important management concepts & their application, to have an insight of various functional departments in an organization.
- Discuss the importance of security, privacy, and ethical issues as they relate to information systems.
- Identify some of the strategies employed to lower costs and improve service.

UNIT – 1 Basics of MIS: Introduction, Structure of MIS, Purpose, Objectives, Benefits, Limitations, Requirements, Characteristics,

UNIT -2: Re of MIS in Organizations, Nature and Scope, Foundation of IS- IS classification,

UNIT-3: General Support System, Information system for decision making , The re of system analyst , Data base management system.

UNIT – 4 System Study: SDLC, System Designing models,

UNIT-5 System Analysis Tos : DFD. Decision Trees, Decision Tables,

UNIT-6 Structured English, Data Dictionary along with its Pros and Cons.

UNIT – 7 Trends and applications of IS: Information Concepts:- Types, Information Quality,

UNIT-8 Dimensions of Information. System Concepts- Kinds of System, System Related Concepts.

UNIT-9 Information Technogy, a Managers overview, managerial overview of Computer hardware & software, Telecommunication, Database management.

UNIT – 10 IS for Business Applications: Business application of Information Technogy,

UNIT-11 internet & electronic commerce, intranet, extranet & Information system for managerial

UNIT-12 decision support system and Types of Decisions in Organization, information system for strategic advantage.

UNIT – 13 Advanced Concepts of IS: Enterprise Resource planning, Supply chain management,

UNIT- 14 Customer Relationship Management (CRM), Procurement Management System, Implementation Process,

UNIT - 15 System Maintenance and System Evaluation, IS Security and Ethical responsibility

Text books/reference books:

1. Brian, “Management Information System”, TMH.
2. Alter, “Information Systems: A Management Perspective” Addison Wesley
3. Jawadegar, “Management Information System”, TMH.
4. Bansal, “Information System Analysis & Design”, TMH.

Course Outcomes

At the end of the course, the student will be able to:

CO1: Understand the usage of MIS in organizations and the constituents of the MIS.
CO2: Understand the classifications of MIS, understanding of functional MIS and the different functionalities of these MIS. This would be flowed by case studyon Knowledge management.
CO3: Define and linking MIS to business strategy and the areas in which MIS would lead to strategic advantage. This would be flowed by case study and guest lecture.
CO4: Apply and Learns the functions and issues at each stage of system development. Further different ways in which systems can be developed are also learnt.
CO5 understanding about emerging MIS technogies like ERP, CRM,SCM and trends in enterprise applications.

Course Objectives:

- To know different programming paradigms.
- To perform the object oriented programming concepts and methodology using c++.
- To develop the conse applications using C++.

LIST OF EXPERIMENTS

1. Write a program to Create Class with Static Member Function.
2. Write a program to define a class to represent a bank account. Include the flowing members
:
 - Data Members
 - Name of the depositor
 - Account number
 - Type of account
 - Balance amount in the account
 - Member Functions
 - To assign initial values
 - To deposit an amount
 - To withdraw an amount after checking the balance
 - To display name and balance

Write a program to test the program.

3. Write a program to using INLINE function.
4. Write a program to using FRIEND function
5. Write a program to using Function Overloading.
6. Write a program to using inheritance (Multiple and Multi Level.) Concept.
7. Write a program which reads a text from the keyboard and displays the flowing information on the screen in two cumns:
 - Number of lines
 - Number of words
 - Number of characters

Strings should be left justified and numbers should be right justified in a suitable field width.

8. Write a program to create files with constructor function.
9. Write a program reading from two files simultaneously.
10. Write program containing a possible exception. Use a try block to throw it and a catch block to handle it properly.

COURSE OUTCOMES

At the end of the course, the student will be able to:

CO1: Apply OOPs features to program design and implementation.

CO2: Create Classes according to the problem and implement programs in C++

CO3: Implement Object Oriented Programs using templates and exceptional handling concepts.

CO4: Perform conse operations, applications and file handling.

CO5: Implement applications using C++.

MCA 207: Database Management System Lab Using SQL

Course Objectives:

- To understand the different issues involved in the design and implementation of a database system.
- To study the physical and logical database designs, database modeling, and relational models.
- To understand and use SQL to query, update, and manage a database.
- To develop an understanding of essential DBMS concepts such as: transaction processing, integrity, concurrency, and recovery in databases.
- To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.

List of Exercises

1. Create the student/employee Table and construct the following queries for the database...
 - 1..1. Create the table for student/employee.
 - 1..2. Find out name of all students.
 - 1..3. Retrieve the list of name and the city of all students.
 - 1..4. List of all students/employee who stay in city "BOMBAY" or city DELHI".
 - 1..5. List of all students /employee who are located in "MADRAS".
2. (1)Apply these Operations on employee table
 - 2..1. Insert
 - 2..2. Select
 - 2..3. Update
 - 2..4. Drop
 - 2..5. Delete
 - 2..6. Alter
3. Create table with attributes emp. No., emp. Name, Designation, Salary, and Department no. Construct for following queries.....
 - .1 Display complete information of all the employees working as a manager.
 - .2 Display name of all the employees working as a clerk.
 - .3 Suppose DA for manager is 75% of salary then display name of all managers.
 - .4 Select names and designation whose salary is greater than 15000.
 5. Apply key constraints as Primary Key, Foreign Key etc as per requirement.
4. Between operation- list of all Employee Name & DOJ (date of joining) to join the Company in 2010
5. Join operation- list of all the employees along with their department information by using join operation.

6. AND/OR operation- make a table that have an employee Perform AND/OR operation.
7. Group by function-
Create the table for facilities having faculty-id, dept. no., designation name and group by similar dept.no. Facilities by using count function.
8. Order by ACS function-
(a) Create a table for emp. Using flowing data:- emp. name, emp age, emp salary, emp city & display the emp salary in ascending and descending order.
9. Max-Min function- create a table for student having similar attributes s_name, S_marks, s_id, s_sec& remark.
 - i. Find the maximum marks obtained by student.
 - ii. Find the minimum marks obtained by student.
 - iii. Sum of all students marks using sum function.
 - iv. Find the average of marks using avg function.
10. Drop operation- Perform Drop Operation.
11. a) Define DBMS.
 - b) Key Component- Entity, Attributes
 - c) SQL
 - 1) DDL
 - 2) DML
 - d) Relational data model-
 - 1) Relation
 - 2) Tuple
 - 3) Domain
 - 4) Degree

COURSE OUTCOMES:

At the end of the course students will be able to

CO1.Designing and creating relational database systems using SQL,

CO2. Demonstrate the use of constraints and relational algebra operations.

CO3.Understand various advanced queries execution such as relational constraints, joins, set operations, aggregate functions, triggers, views and embedded SQL.

CO4: Design and implement database applications on their own.

CO5: Understand of essential DBMS concepts such as: database security, integrity, concurrency,

MCA 208: Software Engineering Lab

Course Objective:

- To help students to develop skills that will enable them to construct software of high quality software that is reliable and reasonably also easy to understand, modify and maintain.
- To foster an understanding of why these skills are important.

To Required: Rational Rose Enterprise Edition

List of Experiments:

1. Development of requirements specification, function oriented design using/SD, object-oriented design using UML, test case design, and implementation using Java and testing. Use of appropriate CASE tos and other tos such as configuration management tos, program analysis tos in the software lifecycle.
2. Develop Software Requirements Specification (SRS) for a given problem in IEEE template.
3. Develop DFD model (level-0, level-1 DFD and Data dictionary) of the project.
4. Develop structured design for the DFD model developed.
5. Developed all Structure UML diagram of the given project.
6. Develop Behavior UML diagram of the given project.

Course Outcomes:

At the end of the course, the student will be able to:

- CO1.** Preparing of software requirements specification for a given problem in IEEE template using UML.
- CO2.** Use of appropriate CASE tos.
- CO3.** Implement models for software applications.
- CO4.** Create DFD's for software applications.
- CO5.** Use the different UML notations for designing software.

THIRD SEMESTER

THEORY PAPERS			Marks Allocation			
Code	Subject/Paper	Type	IA	EA	Total	Credits
MCA 301	Python Programming	Core	30	70	100	4
MCA 302	Cryptography and Network Security	Core	30	70	100	4
MCA 303	Artificial Intelligence	Core	30	70	100	4
MCA 304	PHP	Core	30	70	100	4
	Elective(Select any one)					
MCA 305.A	E-Commerce	Elective	30	70	100	4
MCA 305.B	Big Data Analytics	Elective	30	70	100	4
MCA 305.C	Internet of Things	Elective	30	70	100	4
PRACTICALS/VIVA-VOCE			Major	Practical	Total	Credits
MCA 306	Python Programming Lab	Practical	25	25	50	2
MCA 307	PHP Lab	Practical	25	25	50	2
MCA 308	Seminar	Practical	25	25	50	2
TOTAL			225	425	650	26

MCA 301: Python Programming

UNIT :1

Introduction to Python: Python Installation and Working of it, get familiar with python variables and data types, Operator understanding and its usage, detail study of python blocks,

UNIT :2

Structure Types and mutability: Hands on with conditional blocks using if, else and elif, Hands on examples and study of looping with range

UNIT :3

List and dictionaries. Hands on to organize python code with function, modular approach in python.

UNIT: 4

Exception, Testing and Debugging: Handling if exceptions to handle the code cracks, handling and helping file operations, coding with the exceptional handling and testing Anonymous method, Properties, Indexers, Exception Handling

UNIT :5

Classes and OOP Concepts: Procedural and Object-Oriented Programming, Classes and working with instances, Method overloading

UNIT :6

Pymorphism, importing internal UNIT as well as external UNITs in the code Packages understanding and their usage, hands on with Lambda function in python coding with the use of functions, UNITs and external packages

UNIT :7

Algorithm and Data Structure: Stack, Queue, Tree, ordered list

UNIT :8

Introduction to Recursion, Divide and Conquer Strategy, Greedy Strategy, Graph Algorithms.

UNIT :9

Advance Topics: Regular Expression, Multi thread Programming, Security

UNIT :10

Using Databases in Python: Python MySQL Database Access, Install the MySQLdb and other Packages

UNIT :11

Create Database Connection, CREATE, INSERT, READ, UPDATE and DELETE Operation, DML and DDL Oepration with Databases

UNIT :12

Performing Transactions, Handling Database Errors

UNIT: 13

Python for Data Analysis: Numpy: Introduction to numpy Creating arrays Using arrays and
UNIT: 14

Scalars Pandas: What is a panda? Where it is used?

UNIT: 15

Series in pandas, Matplotlib: Python for Data Visualization

TEXT/REFERENCE BOOKS:

1. Starting Out with Python (2009) Pearson ,Tonny Gaddis
2. Beginning PythonWrox Publication Peter Norton, Alex Samuel
3. Python Algorithms Apress, Magnus LietHetland,
4. Python Object Oriented Programming PACKT Press, Dusty Phillips
5. Python for Unix and Linux System Administration O'Relly, Noad Gift

Course Outcomes:

CO1: To learn basics of Python

CO2: To develop conse application in python

CO3: To Implement Data structures using python.

CO4: To develop database application in python

CO5: To Use various data analysis libraries available in Python

MCA 302: Cryptography and Network Security

Course Objectives:

- This course focuses on the models, tos, and techniques for enforcement of security with some emphasis on the use of cryptography.
- To develop an understanding of security picies (such as authentication, integrity and confidentiality).

Syllabus

UNIT: 1

Multi level model of security, Cryptography, Secret Key Cryptography

UNIT: 2

Modes of Operation, Hashes and Message Digest, Public Key Algorithm,

UNIT: 3

Security Handshake Pitfall, Strong Password Protoc; Case study of real time communication security;

UNIT: 4

Introduction to the Concepts of Security, Security Approaches, Principles of security, Types of attacks.

UNIT: 5

Cryptographic Techniques: Plain text and Cipher text, Substitution Techniques

UNIT: 6

Transposition Techniques Encryption and Decryption, Symmetric and Asymmetric Key Cryptography. Computer-based symmetric Key Cryptographic;

UNIT: 7

Algorithms: Algorithm Types and Modes, An Overview of Symmetric Key Cryptography

UNIT: 8

Data Encryption Standard (DES), International Data Encryption Algorithm (IDEA), Advanced Encryption Standard (AES);

UNIT: 9

Computer-based Asymmetric Key Cryptographic Algorithms; Cryptography, An Overview of Asymmetric Key Cryptography, The RSA algorithm, Symmetric and Asymmetric Key Cryptography Together, Digital Signatures, Knapsack Algorithm;

UNIT: 10

Public Key Infrastructure (PKI) Digital Certificates, Private Key Management , The PKI Model, Public Key Cryptography Standards (PKCS)

UNIT: 11

Internet Security Protocs Secure Socket Layer (SSL) , Secure Hyper Text Transfer Protoc (SHTTP) , Time Stamping Protoc (TSP)

UNIT: 12

Secure Electronic Transaction (SET), SSL versus SET, 3-D Secure Protoc , Electronic Money , Email Security;

UNIT: 13

User Authentication Mechanisms: Authentication Basics, Passwords, Authentication Tokens, Certificate-based Authentication

UNIT: 14

Practical Implementations of Cryptography/Security: Cryptographic Solutions Using Java, Cryptographic Solutions Using Microsoft,

UNIT: 15

Cryptographic Tokens, Security and Operating Systems; Network Security: Brief Introduction to TCP/IP, Firewalls, IP Security, Virtual Private Networks (VPN); Case Studies on Cryptography and Security:

TEXT/REFERENCE BOOKS:

1. Atul Kahate "Cryptography and Network Security" Tata McGraw-Hill
2. Charlie Kaufman, Radia Perlman, Mike Speciner "Network Security" Pearson,
3. J. A. Coopers "Computer Communication Security" TMH,
4. D.W. Davies W. L. Price "Security for Computer Networks"
5. John Wiley Sons, L. Stein "Web Security A step by step Guide" Addison Wesley.

COURSE OUTCOMES

At the end of the course, the student will be able to:

CO1: Understand, compare and apply different encryption and decryption

CO2: Understand the security, cryptography, system attacks and defences against them

CO3: Apply and evaluate the performance of different algorithms for cryptography and encryption.

CO4: Learn Key management system for security management system.

CO5: Apply User Authentication Mechanisms techniques to solve problems related to confidentiality and authentication.

MCA 303 :Artificial Intelligence

Course Objective:

- To introduce the basic principles, techniques, and applications of Artificial Intelligence.
- To Become familiar with basic principles of AI toward problem solving, inference, perception, knowledge representation, and learning.

Syllabus

UNIT :1

Introduction to AI, Various types of production systems, Characteristics of production systems,

UNIT :2

Study and comparison of breadth first search and depth first search. Techniques,

UNIT :3

Other Search Techniques like hill Climbing, Best first Search. An algorithm, AO algorithms etc, and various types of contr strategies.

UNIT :4

Knowledge Representation, Problems in representing knowledge, knowledge representation using propositional and predicate logic, comparison of propositional and predicate logic,

UNIT :5

Resution, refutation, deduction, theorem proving, inferencing,

UNIT :6

monotonic and nonmonotonic reasoning.

UNIT :7

Probabilistic reasoning, Baye's theorem,

UNIT :8

Semantic networks scripts schemas, frames

UNIT :9

Conceptual dependency and fuzzy logic, forward and backward reasoning.

UNIT :10

Game playing techniques like minimax procedure, alpha-beta cut-offs etc, planning

UNIT :11

Study of the block world problem in robotics

UNIT :12

Introduction to understanding and Natural Languages Processing.

UNIT :13

Introduction to learning, various techniques used in learning,

UNIT :14

Introduction to Neural Networks, applications of Neural Networks

UNIT :15

Common sense reasoning, some example of Expert systems.

TEXT/ REFERENCE BOOKS:

1. E.Rich,K Knight-Artificial Intelligence, Tata McGraw Hills.
2. S.Russell,P.Norving-Artificial Intelligence-A Modern Approach, Pearson Education, Asia.
3. Thomas Dean-Artificial Intelligence-Theory & Practice, Pearson Education, Asia.
4. Alison Caursey - The Essence of Artificial Intelligence, Pearson Education, Asia.

COURSE OUTCOMES

At the end of the course, the student will be able to:

CO1: Describe knowledge of the building blocks of AI as presented in terms of intelligent agents.

CO2: Apply basic principles of AI in situations that require problem solving, inference, perception, knowledge representation, and learning.

CO3: Represent various real life problem domains using logic based techniques and use this to perform inference or planning.

CO4: Understanding of Game playing techniques and apply various tools and techniques in natural language processing

CO5: Apply various techniques used in learning to solve the real life problems and understanding of neural networks.

MCA 304 PHP

Course Objective

- To facilitate the student with the basics of server side programming aspects, using php as the primary language.
- To emphasis on the basic php programming constructs which is widely used in server side languages.

Syllabus

UNIT :1

Introduction of web applications. Introduction to web designing with HTML

UNIT :2

Cascaded Style Sheets. Concept of Client Side Scripting and Server Side Scripting.

UNIT :3

Static website vs Dynamic website development. Web Servers: Local Servers and Remote Servers.

UNIT :4

Introduction to PHP, Installing Web servers, PHP configuration in IIS & Apache Web server.

UNIT :5

Data types in PHP, Variables, Constants, operators and Expressions.

UNIT :6

PHP Operator: Conditional Structure - if, switch case & Looping Structure - for, while, do while, for each

UNIT :7

Introduction to Arrays: Initialization of an array, Iterating through an array, Sorting arrays, Array Functions

UNIT :8

Functions: Defining and Calling Functions, Passing by Value and passing By references

UNIT :9

Inbuilt Functions: String Function, Math Function, Date Function and Miscellaneous Function.

UNIT :10

Working with Forms: Get and Post Methods, Query strings, HTML form contrs and PHP,

UNIT :11

Maintaining User State: Cookies, Sessions and Application State.

UNIT :12

Working with Files: Opening and Closing Files, Reading and Writing to Files, Getting Information on Files

UNIT :13

PHP Database Connectivity: Introduction to MYSQL

UNIT :14

Creating database and other operations on database, connecting to a database

UNIT :15

Use a particular database, Sending query to database, Parsing of the query results, Checking data

errors.

TEXT/ REFERENCE BOOKS:

1. Steven Hzner “ PHP: The Complete Reference”
2. Tim Converse, Joyce Park “PHP Bible”, 2nd Edition
3. Dave W. Mercer, Allan Kent, Steven D. Nowicki, David Mercer, Dan Squier, Wankyu Choi with HeowEide-Goodman, Ed Lecky-Thompson, Clark Morgan “Beginning PHP5”

COURSE OUTCOME

At the end of the course, the student will be able to:

CO1: Understand how client side and server-side programming works on the web.

CO2: Understand basic PHP syntax for variable and standard language constructs, such as Conditionals and loops.

CO3: Use PHP built-in functions and creating custom functions.

CO4: Working with Forms to receive and process form submission data and Reading and writing cookies.

CO5: Create a database in php My Admin and process data in a MySQL database.

MCA 305.A: E-Commerce

Course Objective:

- To introduce e-commerce, its impacts on business processes, and keys issues in the development of web-based business information systems and applications.
- To deal with issues of security, electronic payments, mobile commerce and the future aspects of e-commerce.

Syllabus

UNIT :1

Introduction, Definition, Objectives, Advantages and disadvantages

UNIT :2

Forces driving E-Commerce, Traditional commerce Vs. E-Commerce

UNIT :3

E-Commerce opportunities for industries, Growth of E-Commerce.

UNIT :4

E-Commerce Models: Business to consumer, Business to Business, Consumer to Consumer, **UNIT**

:5
other models – Brokerage Model, Aggregator Model, Info-mediary Model

UNIT :6

Community Model and value chain Model.

UNIT :7

Electronic Payment Systems: Special features required in payment systems

UNIT :8

Types of E-payment systems, E-Cash, E-cheque

UNIT :9

credit card, Smart Card, Electronic Purses.

UNIT :10

E-Marketing

UNIT: 11

E-Customer Relationship Management

UNIT: 12

E-Supply Chain Management.

UNIT: 13

Security Issues in E-Commerce: Security risk of E-Commerce

UNIT: 14

Types of threats, Security tos and risk management approach

UNIT: 15

Cyber laws, Business Ethics, EDI Application in business

Text/ Reference Books:

- David Whiteley, “E-Commerce”, Tata McGraw Hill, 2000 Eframi Turban, Jae Lee
- David King, K. Michale Chung, “Electronic Commerce”, Pearson Education, 2000

COURSE OUTCOMES

At the end of the course, the student will be able to:

CO1: Understand a good knowledge of e-commerce, both the technical and business aspects.

CO2: Describe different E-Commerce Models.

CO3: Apply the principles and practices of e-commerce and its related technologies like different e-payment technologies.

CO4: Apply a marketing plan and promotional plan for an e-commerce.

CO5: Analyzing the security issues and security approaches.

MCA 305.B : Big Data Analytics

Course Objectives:

- To provide an overview of an exciting growing field of big data analytics.
- To introduce the tools required to manage and analyze big data like Hadoop, NoSql Map Reduce.
- To teach the fundamental techniques and principles in achieving big data analytics with scalability and streaming capability.
- To enable students to have skills that will help them to solve complex real-world problems in for decision support

Syllabus

UNIT: 1 Introduction to Big Data: Big data features and challenges, Problems with Traditional Large-Scale System , Sources of Big Data, 3 V's of Big Data,

UNIT: 2 Types of Data. Working with Big Data: Google File System.Hadoop Distributed File System (HDFS) - Building blocks of Hadoop (Namenode. Data node.Secondary Namenode.Job Tracker. Task Tracker),

UNIT: 3 Introducing and Configuring Hadoop cluster (Local. Pseudo distributed mode, Fully Distributed mode). Configuring XML files.

UNIT: 4 Writing Map Reduce Programs: A Weather Dataset.

UNIT: 5 Understanding Hadoop API for Map Reduce Framework (old and New).

UNIT: 6 Basic programs of Hadoop Map Reduce: Driver code. Mapper code, Reducer code. Record Reader, Combiner, Partitioner.

UNIT: 7 Hadoop I/O: The Writable Interface. Writable Comparable and comparators. **UNIT: 8** Writable Classes: Writable wrappers for Java primitives. Text. Bytes Writable. Null Writable,

UNIT: 9 Object Writable and Generic Writable. Writable collections. Implementing a Custom Writable: Implementing a Raw Comparator for speed, Custom comparators.

UNIT: 10 Pig:Hadoop Programming Made Easier Admiring the Pig Architecture, Going with the Pig Latin Application Flow.

UNIT: 12 Working through the ABCs of Pig Latin. Evaluating Local and Distributed Modes of Running Pig Scripts,

UNIT: 13 Checking out the Pig Script Interfaces, Scripting with Pig Latin. Applying Structure to Hadoop Data with Hive: Saying Hello to Hive,

UNIT: 14 Seeing How the Hive is Put Together, Getting Started with Apache Hive. Examining the Hive Clients. Working with Hive Data Types.

UNIT: 15 Creating and Managing Databases and Tables, Seeing How the Hive Data Manipulation Language Works, Querying and Analyzing Data.

TEXT/ REFERENCE BOOKS :

1. "Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data" by EMC Education Services
2. "Big Data: Does Size Matter?" by TimandraHarkness
3. "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses" by Michael Minelli

Course Outcomes :

At the end of the course, Students will be able to:

CO1: Understand the key issues in big data management and its associated applications and analyze the Big Data framework like Hadoop.

CO2: Give examples of Basic programs of Hadoop Map Reduce.

CO3: implement the Writable interface, Writable Comparable, Writable Generic, clections.

CO4: Implement of Big Data Analytics using pig to sve data intensive problems and to generate analytics.

CO5: Apply Structure to Hadoop Data with Hive.

MCA 305.C Internet of Things

Course Objectives:

- To explore the interconnection of the physical world and the cyber space.
- To develop & design the IOT Devices and communication models.

Syllabus

UNIT: 1 Introduction to IoT: Definition and characteristics of IoT,

UNIT: 2 Design of IOT: Physical design of IOT, Logical Design of IOT-

UNIT: 3 Functional Blocks, and communication models, communication APIs,

UNIT: 4 IOT enabling Technologies- Wireless Sensor Networks,

UNIT: 5 Cloud computing, big data analytics, embedded systems.

UNIT: 6 IOT Levels and deployment templates.

UNIT: 7 IoT Hardware and Software: Sensor and actuator,

UNIT: 8 Humidity sensors, Ultrasonic sensor, Temperature Sensor,

UNIT: 9 Arduino, Raspberry Pi, LiteOS, RIOTOS, Contiki OS, Tiny OS.

UNIT: 10 Architecture and Reference Model: Introduction, Reference Model and architecture,

UNIT: 11 Representational State Transfer (REST) architectural style, Uniform Resource Identifiers (URIs).

UNIT: 12 Challenges in IoT- Design challenges, Development challenges, Security challenges, Other challenges.

UNIT: 13 IOT and M2M: M2M, Difference and similarities between IOT and M2M,

UNIT: 14 Software defined networks, network function virtualization, difference between SDN and NFV for IoT. Case study of IoT Applications:

UNIT: 15 Domain specific IOTs- Home automation, Cities, environment, Energy, Retail, Logistics, Agriculture, Industry, Health and Lifestyles.

REFERENCE BOOKS:

1. Adrian McEwen, “Designing the Internet of Things”, Wiley Publishers, 2013, ISBN: 978-1-118-43062-0
2. Daniel Kellmerit, “The Silent Intelligence: The Internet of Things”. 2013, ISBN 0989973700

COURSE OUTCOMES

At the end of the course, the student will be able to:

CO1 Understand the building blocks of Internet of Things and characteristics and application areas of IOT.

CO2: Understand of IOT enabling Technologies.

CO3: Implement interfacing of various sensors with Arduino/Raspberry Pi.

CO4: Implement Architecture and Reference Model for in IoT- Design and security aspects

CO5: Understand the revution of Internet in Mobile Devices, Cloud & Sensor Networks and implementation of automation in different domain.

MCA 306: Python Programming Lab

Course Objective:

- To describe the need for Object-oriented programming concepts in Python.
- To infer the supported data structures like lists, dictionaries and tuples in Python.
- To illustrate the application of matrices and regular expressions in building the Python programs.
- To discover the use of external UNITS in creating excel files and navigating the file systems.

List of Experiments

1. Write a program to demonstrate basic data type in python.
2. Write a program to compute distance between two points taking input from the user
3. Write a program add.py that takes 2 numbers as command line arguments and prints its sum.
4. Write a Program for checking whether the given number is an even number or not.
5. Using a for loop, write a program that prints out the decimal equivalents of $1/2$, $1/3$, $1/4$, . . . , $1/10$
6. Write a Program to demonstrate list and tuple in python.
7. Write a program using for loop that loops over a sequence.
8. Write a program using a while loop that asks the user for a number, and prints a countdown from that number to zero.
9. Find the sum of all the primes below two million.
10. By considering the terms in the Fibonacci sequence whose values do not exceed four million, WAP to find the sum of the even-valued terms.
11. Write a program to count the numbers of characters in the string and store them in a dictionary data structure.
12. Write a program to use split and join methods in the string and trace a birthday of a person with a dictionary data structure
13. Write a program to count frequency of characters in a given file. Can you use character frequency to tell whether the given file is a Python program file, C program file or a text file?
14. Write a program to print each line of a file in reverse order.
15. Write a program to compute the number of characters, words and lines in a file.
16. Write a function nearly equal to test whether two strings are nearly equal. Two strings a and b are nearly equal when a can be generated by a single mutation on.
17. Write function to compute gcd, lcm of two numbers. Each function shouldn't exceed one line.
18. Write a program to implement Merge sort.

19. Write a program to implement Selection sort, Insertion sort.

Course Outcomes:

At the end of the course, the student will be able to:

CO1: Create, Test and Debug Python Programs

CO2: Implement Conditionals and Loops for Python Programs

CO3: Use functions and represent Compound data using Lists, Tuples and Dictionaries

CO4: Read and write data from & to files in Python and develop Application using Python.

CO5: Illustrates sort methods in Python Programs.

MCA 307: PHP Lab

Course Objectives:

- To understand how server-side programming works on the web.
- To learn PHP Basic syntax for variable types and calculations.
- To use PHP built-in functions and creating custom functions.
- To understanding POST and GET in form submission.
- To provide the necessary knowledge to design and develop dynamic, databasedriven web applications using PHP.

List of Experiments :-

Experiment 1: Design the flowing static web pages required for online book store.

- a) **Home page:** - the static home page must contains three pages
- b) **Top:** - logo and clege name and links to homepage, login page, registration Page, catalogue page and cart page
- c) **Left:** - at least four links for navigation which will display the catalogue of Respective links
- d) **Right:** - the pages to links in the left frame must be loaded here initially it Contains the description of the website

Experiment 2: Create registration and cart page in the previous created web site.

Experiment 3: Write a java script to validate the flowing fields in a registration page

- a) userName (should contains alphabets and the length should not be less than 6 characters)
- b) userPassword (should not be less than 6 characters)
- c) userEmail (should not contain invalid addresses)
- d) userCity (should select city from drop down)
- e) userGender (Should selectgender)

Experiment 4: Implement CSS on the above create WebPages.

Experiment 5: Write an XML file which displays the book details that includes the flowing:

1) Title of book 2) Author name 3) Edition 4) Price Write a DTD to validate the above XML file and display the details in a table.

Experiment 6: Create a php program to demonstrate the different file handling methods.

Experiment 7: Create a php program to demonstrate the different loops in php.

Experiment 8: Create a php program to demonstrate the different predefined function in array, Math.

Experiment 9: Create a php program to demonstrate the different predefined function in Data & Regular Expression, date.

Experiment 10: Create a HTML form and process the HTML form in PHP.

Experiment 11: Create a php program to connect to MySQL Server.

Experiment 12: Create a php program to execute more SQL queries.

Course Outcomes

At the end of the course, the student will be able to:

- CO1: Understand the PHP and scripting
- CO2: Understand Basics of PHP Language.
- CO3: Analyze with Databases and Forms.
- CO4: Apply methods with cookies.
- CO5: Describe on Data and Tables in MYSQL.

MCA 308: Seminar

Course Objectives:

- To Awareness of how to use values in improving your own professionalism.
- To Learning about personal and communication styles for team building.
- To identify, formulate and present model problems.
- To Learning management of values

Course Outcomes:

At the end of the course, the student will be able to:

CO1: Personalize and create a communication style for individual & team building.

CO2: Use values in improving one's own professionalism

CO3: Develop the higher cognitive abilities that are analysis, synthesis and evaluation.

CO4: Ability to identify, formulate and present model problems.

CO5 : Describe latest technologies in own profession.

FOURTH SEMESTER

THEORY PAPERS			Marks Allocation			
Code	Subject/Paper	Type	IA	EA	Total	Credits
MCA 401	Programming in Java	Core	30	70	100	4
MCA 402	Machine Learning	Core	30	70	100	4
	Elective(Select any one)					
MCA 403.A	Cloud Computing	Elective	30	70	100	4
MCA 403.B	Software Project Management	Elective	30	70	100	4
MCA 403.C	Data mining and Were house	Elective	30	70	100	4
MCA 404	Research and Publication Ethics	Skill	30	70	100	4
PRACTICALS/VIVA-VOCE			Minor	Practical	Total	Credits
MCA 405	Java Programming Lab	Practical	25	25	50	2
MCA 406	Machine Learning Lab	Practical	25	25	50	2
MCA 407	Project	Practical	75	75	150	6
TOTAL			245	405	650	26

MCA 401: Programming in Java

Course Objectives:

- To understand a good design and programming in Java Programming.
- To Create Java programs that coverage the object-oriented features of the Java language, such as encapsulation, inheritance and polymorphism; use data types, arrays and other data collections.
- To implement the error-handling techniques using exception handling.

Syllabus

UNIT :1

The Genesis of Java: The importance of Java to Internet, Java's magic-the byte code, introduction to JDK and JVM, the Java libraries.

UNIT :2

Data Types, Variables and Arrays: Java Programming: Data types, access Specifiers, operators,

UNIT :3

control statements, arrays; Classes: Fundamentals, objects, methods, constructors.

UNIT :4

Usage of this keyword, garbage collection, the finalize() method, overloading methods, using objects as parameters, argument passing, returning objects

UNIT :5

Recursion, introducing access control, understanding static, introducing final, arrays revisited, nested and inner classes, exploring string class, using command-line arguments.

UNIT :6

Inheritance: Inheritance basics, using super, creating a multilevel hierarchy, when constructors are called, method overriding, dynamic method dispatch, using abstract, using final with inheritance, the object class.

UNIT :7

Package, Interfaces: Packages, access protection, importing packages, interfaces.

UNIT :8

Java Library: String handling (only main functions), String Buffer class. Elementary concepts of Input/Output: byte and character streams

UNIT :9

System.in and System.out, print and println, reading from a file and writing in a file.

UNIT :10

Exception Handling: exception-handling fundamentals, exception types, uncaught exceptions, using try and catch, multiple catch clauses, nested try statements, throw, throws, finally, Java's built-in exceptions, creating your own exception subclasses, using exceptions.

UNIT :11

Multithreaded Programming: The Java thread model, the main thread, creating a thread, creating multiple threads, using Alive() and join(), thread priorities,

UNIT :12

Synchronization, inter thread Communication, suspending, resuming, and stopping threads, using multithreading

UNIT :13

Applets: Introduction, Life cycle, creation and implementation, AWT contrs: Button, Label, Text Field, Text Area, Choice lists, list, scrlbars, check boxes, Layout managers

UNIT :14

Elementary concepts of Event Handling: Delegation Event Model,

UNIT :15

Event classes and listeners, Adapter classes, Inner classes. Swings: Introduction and comparison with AWT contrs.

TEXT/ REFERENCE BOOKS :

1. E. Balagurusamy, Programming with Java, TMH
2. Herbert Schildt, The Complete Reference: Java, TMH
3. Horstmann, Core Java, Addison Wesley
4. Rich raposa, Learning Java, Wiley

COURSE OUTCOMES OF JAVA PROGRAMMING

At the end of the course, the student will be able to:

CO1: Understand the fundamental concepts of Java Programming language. Use an integrated development environment to write, compile, run, and test simple object-oriented Java programs.

CO2: Understand the principles and practice of object oriented programming in the construction of robust maintainable programs which satisfy the requirements.

CO3: Implement an application that demonstrates their competency with Java syntax, structure and programming logic, incorporating basic features of the language as well as some features from the I/O (Input/Output) or GUI libraries.

CO4: Apply the concepts of Multithreading and Exception handling to develop efficient and error free codes.

CO5: Use of Java Programming language in the development of small to medium sized application programs that demonstrate professionally acceptable coding and performance standards.

MCA 402: Machine Learning

Course Objectives:

- To introduce students to the basic concepts and techniques of **Machine Learning**.
- To develop skills of using recent **machine learning** software for solving practical problems.
- To gain experience of doing independent study and research.

Syllabus

UNIT :1

Supervised learning algorithm: Introduction, types of learning, application

UNIT :2

Supervised learning: Linear Regression Model, Naïve Bayes classifier Decision Tree

UNIT :3

K nearest neighbor, Logistic Regression, Support Vector Machine, Random forest algorithm

UNIT :4

Unsupervised learning algorithm: Grouping unlabeled items using k-means clustering, Hierarchical Clustering, Probabilistic clustering

UNIT :5

Association rule mining, Apriori Algorithm

UNIT :6

f-p growth algorithm, Gaussian mixture model.

UNIT :7

Introduction to Statistical Learning Theory, Feature extraction-Principal component analysis,

UNIT : 8

Singular value decomposition. Feature selection–feature ranking and subset selection

UNIT: 9

Filter, wrapper and embedded methods

UNIT: 10

Evaluating Machine Learning algorithms and Model Selection.

UNIT: 11

Semi supervised learning, Reinforcement learning : Markov decision process (MDP), Bellman equations

UNIT: 12

Poly evaluation using Monte Carlo, Policy iteration and Value iteration, Q-Learning

UNIT: 13

State-Action- Reward-State-Action (SARSA), Model-based Reinforcement Learning.

UNIT :14

Recommended system, Collaborative filtering, Content-based filtering

UNIT :15

Artificial neural network, Perception, Multilayer network, Back propagation, Introduction to Deep learning.

REFERENCE/TEXT BOOKS:

1. Tom M Mitchell, Machine Learning, McGraw Hill Education
2. Bishop, C. (2006). Pattern Recognition and Machine Learning. Berlin: Springer-Verlag.
3. Duda, Richard, Peter Hart, and David Stork. Pattern Classification. 2nd ed. New York, NY: Wiley-Interscience, 2000. ISBN: 9780471056690.
4. Bishop, Christopher. Neural Networks for Pattern Recognition. New York, NY: Oxford University Press, 1995. ISBN: 9780198538646.
5. Introduction to Machine Learning - Ethem Alpaydin, MIT Press, Prentice hall of India.

COURSE OUTCOMES

At the end of the course, the student will be able to:

CO1: Describe intelligent agents for search and games

CO2: Convert AI problems through programming with Python

CO3: Learning optimization and inference algorithms for model learning

CO4: Make programs for an agent to learn and act in a structured environment.

CO5: Learn recommended system in ML.

MCA 403.A: Cloud Computing

Course Objectives:

- To understand the basics of Cloud Computing.
- To understand the movement from a traditional network infrastructure to a Cloud solution.

Syllabus

UNIT :1

Introduction: Objective, scope and outcome of the course. Introduction Cloud Computing: Nutshell of cloud computing, Enabling Technology, Historical development, Vision, feature

UNIT :2

Characteristics and components of Cloud Computing. Challenges, Risks and Approaches of Migration into Cloud. Ethical Issue in Cloud Computing, Evaluating the Cloud's Business

UNIT :3

Impact and economics, Future of the cloud. Networking Support for Cloud Computing. Ubiquitous Cloud and the Internet of Things

UNIT :4

Cloud Computing Architecture: Cloud Reference Model, Layer and Types of Clouds, Services models, Data centre Design and inter connection Network

UNIT :5

Architectural design of Compute and Storage Clouds. Cloud Programming and Software: Fractures of cloud programming,

UNIT :6

Parallel and distributed programming paradigms-Map Reduce, Hadoop, High level Language for Cloud. Programming of Google App engine.

UNIT :7

Virtualization Technology: Definition, Understanding and Benefits of Virtualization. Implementation Level of Virtualization,

UNIT :8

Virtualization Structure/Tos and Mechanisms, Hypervisor VMware, KVM, Xen. Virtualization: of CPU, Memory, I/O Devices, Virtual Cluster and Resources Management,

UNIT :9

Virtualization of Server, Desktop, Network, and Virtualization of data-centre

UNIT :10

Securing the Cloud: Cloud Information security fundamentals, Cloud security services, Design principles, Policy Implementation, Cloud Computing Security Challenges,

UNIT :11

Cloud Computing Security Architecture .Legal issues in cloud Computing. Data Security in Cloud: Business Continuity and Disaster Recovery ,

UNIT :12

Risk Mitigation, Understanding and Identification of Threats in Cloud, SLA-Service Level Agreements, Trust Management

UNIT :13

Cloud Platforms in Industry: Amazon web services , Google App Engine,

UNIT :14

Microsoft Azure Design, Aneka: Cloud Application Platform-Integration of Private and Public Clouds

UNIT :15

Cloud applications: Protein structure prediction, Data Analysis, Satellite Image Processing, CRM

TEXT/ REFERENCE BOOKS:

1. “ Distributed and Cloud Computing “ By Kai Hawang , GeoffreyC.Fox, Jack J. Dongarra Pub: Elsevier
2. Cloud Computing ,Principal and Paradigms, Edited By RajkumarBuyya, JemesBroberg, A. Goscinski, Pub.- Wiley
3. Pub.- Wiley
4. Kumar Saurabh, “Cloud Computing” , Wiley Pub
5. Krutz , Vines, “Cloud Security “ , Wiley Pub
6. Velte, “Cloud Computing- A Practical Approach” ,TMH Pub

COURSE OUTCOMES

CO1: Analyze the Cloud computing setup with it's vulnerabilities and applications using different architectures.

CO2: Design different workflows according to requirements and apply map reduce programming model.

CO3: Apply and design suitable Virtualization concept, Cloud Resource Management and design scheduling algorithms.

CO4: Create combinatorial auctions for cloud resources and design scheduling algorithms for computing clouds and societal issues invved in addressing the security issues of cloud computing.

CO5: Assess cloud Storage systems and Cloud security, the risks invved, its impact and develop cloud application

MCA 403.B: Software Project Management

Course Outcomes:

- Identify the different project contexts and suggest an appropriate management strategy.
- Practice the re of professional ethics in successful software development.
- Identify and describe the key phases of project management.
- Determine an appropriate project management approach through an evaluation of the business context and scope of the project.
- Understand the fundamental principles of software project management & have a good knowledge of responsibilities of project manager.

UNIT: 1

Project Management: The management spectrum, the people, the product, the process, the project, the W5HH principle

UNIT: 2

Critical practices Metrics for Process and Project: Metrics in the process and project Domains, software measurements, and metrics for software quality

UNIT: 3

Integrating metrics within software process, metrics for small organizations, establishing a software metrics program.

UNIT: 4

Estimation: Observations, Project planning Process, software scope and feasibility, resources, software project estimation

UNIT: 5

Decomposition techniques, empirical estimation models, estimation for object oriented projects,

UNIT: 6

Estimation for Agile development and web engineering projects, the make/buy decision.

UNIT: 7

Project Scheduling: Basic concepts, project scheduling, defining a task set and task network, scheduling, earned value analysis

UNIT: 8

Risk Management: Reactive V/S proactive Risk Strategies, software risks, Risk identification, Risk projection, risk refinement, risk mitigation, monitoring and management

UNIT: 9

The RMMM plan Quality Planning: Quality Concepts, Procedural Approach to Quality Management, Quantitative Approaches to Quality Management, Quantitative Quality Management Planning, Setting the Quality Goal, Estimating Defects for Other Stages, Quality Process Planning, Defect Prevention Planning.

UNIT: 10

Quality Management: Quality Concepts, Software Quality assurances, software reviews, formal technical reviews

UNIT: 11

Formal approaches to SQA, Statistical Software Quality assurances, Change Management: software Configuration Management

UNIT: 12

The SCM repository, SCM Process, Configuration Management for Web Engineering

UNIT: 13

Project Execution And Closure: Reviews. The Review Process, Planning, Overview and Preparation, Group Review Meeting, Rework and Flow-up, One-Person Review,

UNIT: 14

Guidelines for Reviews in Projects, Data Collection, Analysis and Control Guidelines, Introduction of Reviews and the NAH Syndrome. Project Monitoring and Control: Project Tracking, Activities Tracking

UNIT: 15

Defect Tracking, Issues Tracking, Status Reports, Milestone Analysis, Actual Versus Estimated Analysis of Effort and Schedule, Monitoring Quality, Risk-Related Monitoring. Project Closure: Project Closure Analysis, The Role of Closure Analysis, Performing Closure Analysis.

Text/References:

1. R. S. Pressman, Software Engineering, TMH, 7th ed.
2. Pankaj Jalote, Software project management in practice, Addison-Wesley
3. B. Hughes & M. Cotterell, Software Project Management, TMH

COURSE OUTCOMES

At the end of the course, the student will be able to:

- CO: 1. Explain principles of the project lifecycle and how to identify opportunities to work with learners on relevant and appropriate project scenarios to share this understanding
- CO: 2. critically evaluate and discuss the issues around project management and its application in the real world with course participants and learners
- CO: 3. Choose project management techniques for IT projects to initiate, plan, execute and evaluate a project and work in teams to create a project plan for a project scenario that includes key tasks, critical path, dependencies and a realistic timeline.
- CO: 4. Present strategies for gaining confidence in managing projects through simple project planning examples
- CO: 5. Review, Project Execution and closure

MCA 403.C: Data mining and Where house

Course Objectives:

- To Approach business problems data-analytically by identifying opportunities to derive business value from data.
- To Know the basics of data mining techniques and how they can be applied to extract relevant business intelligence

Syllabus

UNIT: 1

The Compelling Need for data warehousing: Escalating Need for strategic information, failures of Past decision-support systems, operational versus decision-support systems

UNIT: 2

Data warehousing – the only viable sution, data warehouse defined Data warehouse – The building Blocks: Defining Features, data warehouses and data marts

UNIT: 3

Overview of the components, metadata in the data warehouse Defining the business requirements: Dimensional analysis, AP operations: Drill-down and rl-up, slice-and-dice or rotation

UNIT: 4

Principles of dimensional modeling: the STAR schema, STAR Schema Keys, Advantages of the STAR Schema Dimensional Modeling: Updates to the Dimension tables

UNIT: 5

Miscellaneous dimensions, the snowflake schema, aggregate fact tables, families of STARS, Steps for the Design & Construction of Data warehouse: Framework, Architecture

UNIT: 6

Type of AP Servers: RAP, MAP, Data warehouse implementation tils & techniques.

UNIT: 7

Data Mining, Data Mining of what kind of Data, Knowledge discovery process (KDD) , What kind of patterns can be mined

UNIT: 8

AP versus data mining, data mining and the data warehouse,

UNIT: 9

Data mining functionalities, classification Systems, Data processing: Cleaning, Integration & transformation, Reduction.

UNIT: 10

Data mining primitives: What defines a Data Mining Task?

UNIT: 11

Languages and System Architectures, Association Rule Mining,

UNIT: 12

Mining of Single dimensional, Boean association rules.

UNIT: 13

Data Mining Query language (DMQL), Cluster Analysis: Partitioning , Hierarchical Density , Grid & Model based methods ., Major Data Mining Techniques

UNIT: 14

Cluster detection, decision trees, memory-based reasoning, link analysis, neural networks, genetic algorithms, moving into data mining

UNIT: 15

Data Mining Applications, Benefits of data mining & applications.

TEXT BOOKS:

1. Jiawei Han, Micheline Kamber, “Data Mining concepts and Techniques”, Elsevier.
2. Paul Raj Poonia, “Fundamentals of Data Warehousing”, John Wiley & Sons, 2003.
3. W. H. Inmon, “Building the Operational Data Store”, 2nd Ed., John Wiley, 1999
4. Sam Anahony, “Data Warehousing in the Real World: A Practical Guide for Building Decision Support Systems”, John Wiley, 2004.
5. Jarke, “Fundamentals of Data Warehouse”, Springer

COURSE OUTCOMES:

At the end of the course, the student will be able to:

CO1: Analyze the Need for data warehousing and Define Features of Data mining.

CO2: Understand dimensional modeling and Steps for the Design & Construction of Data warehouse.

CO3: Discover Data mining for Knowledge discovery to solve problems and Data processing.

CO4: Understand the Data Mining primitives.

CO5: Apply Data Mining Query language for data mining for Business Intelligence Application.

MCA 404: Research and Publication Ethics

Course Objectives:

- To identify the concept of research.
- To identify the scientific conduct of research.
- To understand the publication Ethics.
- To understand Open access publications and publication misconduct.
- To understand the Research Data and Research Metrics.

UNIT: 1

Introduction to Philosophy: definition, nature and Scope Concept,

UNIT: 2

Branches Ethics: definition, moral philosophy

UNIT: 3

Nature of moral judgments and reaction

UNIT: 4

Ethics with respect to science and research, Intellectual honesty and research integrity

UNIT: 5

Scientific misconducts: Falsification, Fabrication, and Plagiarism (FFP)

UNIT: 6

Redundant publications: duplicate and overlapping publications, salami slicing, Selective reporting and misrepresentation of data.

UNIT: 7

Publication ethics: definition, introduction and importance, Best practices /Standards setting initiatives and guidelines: COPE. WAME etc.,

UNIT: 8

Conflicts of interest, Publication misconduct: definition, concept, problems that lead to unethical behavior and vice versa, types

UNIT: 9

Violation of publication ethics, authorship and contributorship, Identification of publication misconduct, complaints and appeals, Predatory publishers and journals

UNIT: 10

Open access publications and initiatives, SHEERPA/RoMEO online resource to check publisher copyright & Self – archiving policies, Software to identify predatory publications developed by SPPU.

UNIT: 11

Journal finder /Journal suggestion tools viz .JANE. Elsevier Journal Finder, Springer Journal Suggester, etc., Subject specific ethical issues, FFP, authorship

UNIT: 12

Conflicts of interest, Complaints and appeals: examples and fraud from India and abroad, Use of plagiarism software like Turnitin, Urkund and other open source software tools

UNIT: 13

Databases: Indexing databases, Citation databases: Web of Science, Scopus etc.

UNIT: 14

Research Metrics: Impact Factor of Journal as per Journal Citation Report, SNIP, SJR, IPP, Cite Score

UNIT: 15

Metrics: h-index, g index, i10 index, altmetrics

TEXT/REFERENCE BOOKS:

1. Bird, A.(2006). Philosophy of Science. Routledge
2. MacIntyre, Alasdair (1967) A Short History of Ethics. London
3. P.Chaddah, (2018) Ethics in Competitive Research: Do not get Scooped; do not get Plagiarized, ISBN:978-9387480865
4. National Academy of Sciences, National Academy of Engineering and Institute of Medicine. (2009). On Being a Scientist: A Guide to responsible conduct in Research: Third Edition, National Academies Press.
5. Indian National Science Academy (INSA), Ethics in Science Education, Research and Governance (2019), ISBN:978-81-939482-1-7 http://www.insaindia.res.in/pdf/Ethics_Book.pdf.

COURSE OUTCOMES

At the end of the course, the student will be able to:

- | COs | Statement After completion of this course, students will be able to: |
|-----|--|
| CO1 | Understand the concept of research. |
| CO2 | Understand the scientific conduct of research. |
| CO3 | Understand the publication Ethics. |
| CO4 | Understand Open access publications and publication misconduct. ct. |
| CO5 | Understand the Research Data and Research Metrics. |

MCA 405 : JAVA PROGRAMMING LAB

Course Objectives:

- To understand object oriented features of java and implementing it in java programming.
- To learn and understand inheritance, interfaces, multithreading and exception handling.
- To understand different input/output objects (input vs. output, character vs. byte, data vs. processing, object) and methods and the structure of the java.io package.
- To learn and understand the use of applets and file handling.

List of experiments

Practical 1: Write a program to compute the sum of the digits of a given integer number.

Practical 2: Given a number, write a programming using (while/ do..while/for) loop to reverse the digits of the number. For example, the number 12345 should be written as 54321.

Practical 3: Write a program (making use of class and methods), which will read a string and rewrite it in the alphabetical order. For example, the word JAIPUR should be written as AIJPRU.

Practical 4: Write a program that accepts a shopping list of five items from the command line and stores them in a vector.

Practical 5: Write a program to show the application of interface and abstract class.

Practical 6: Define an exception called “No Match Exception” that is thrown when a string is not equal to “India”. Write a program that uses this exception.

Practical 7: Write a program to implement multithreading making use of Thread class and/or Runnable interface.

Practical 8: Write a program to implement the concept of packages.

Practical 9: Develop an applet that receives three numeric values as input from the user and then displays the largest of the three on the screen. Write a HTML page and test the applet.

Practical 10: Develop an applet which runs a banner with text “Welcome to Jagannath University” making use of multithreading.

COURSE OUTCOMES:

At the end of the course, a student will be able to

CO1. Use an integrated development environment to write, compile, run, and test simple object-oriented Java programs.

CO2. Read and make elementary modifications to Java programs that solve real-world problems. And Identify Java code utilities in applets, Java packages, and classes.

CO3. Analyze a computer program to solve specified problems and to use the Java SDK environment to create, debug and run simple Java programs.

CO4. Develop programs for inheritance, multithreading, applets, exception handling and file handling.

CO5: write multithreaded programs HTML page and test the applet

MCA 406 : Machine Learning Lab

Course Objective:

- To Make use of Data sets in implementing the machine learning algorithms
- To Analyze and evaluate simple algorithms for pattern classification.
- To implement the machine learning concepts and algorithms in any suitable language of choice.

List of Experiments

1. Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.
2. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.
3. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample
4. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets
5. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.
6. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.
7. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.
8. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.
9. Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.
10. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.

Course Outcomes:

At the end of the course, the student will be able to:

CO1: Build intelligent agents for search and games

CO2: Solve AI problems through programming with Python

CO3: Learning optimization and inference algorithms for model learning

CO4: Design and develop programs for an agent to learn and act in a structured environment.

CO5 : Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points.

MCA 407: Project

Course Objective:

- To introduce the concept and methods required for the construction of large software intensive system.
- To develop a broad understanding of the discipline of software engineering and management of software system.
- To provide an understanding of both theoretical and methodological issues invve in modern software engineering project management and focus strongly on practical techniques.

Course Outcomes

At the end of the course, the student will be able to:

CO1: Understand programming language concepts, particularly Development languages and object-oriented concepts or go through research activities.

CO2: Plan, analyze, design and implement a software project or gather knowledge over the field of research and design or plan about the proposed work.

CO3: Demonstrate the ability to locate and use technical information from multiple sources.

CO4: Demonstrate the ability to communicate effectively in speech and writing.

CO5: Learn to work as a team and to focus on getting a working project done on time with each student being held accountable for their part of the project.