UNIVERSITY OF ENGINEERING AND MANAGEMENT, KOLKATA

DEPARTMENT OF COMPUTER APPLICATIONS

PROGRAM: MASTER OF COMPUTER APPLICATIONS

DETAILED SYLLABUS

BATCH: 2021 – 2023

UNIVERSITY OF ENGINEERING & MANAGEMENT, NEW TOWN,

UNIVERSITY AREA, PLOT NO. III, B/5, NEW TOWN RD,

ACTION AREA III, NEWTOWN, NEW TOWN, WEST BENGAL 700160

Table of Contents

Syllabus Structure

1st Year 1st Semester

Course Code	Course Title	Total No. of Contact Hours				Total No. of Credits
		Lecture (L)	Tutorial (T)	Practical (P)	Total Hours	
1st Semester (Theory)						
MCA101	Computer Organization and Architecture	3	1	0	4	3
MCA102	Computer Programming with C	3	1	0	4	4
MCA103	Data Structures with C	3	1	0	4	4
MCA104	Discrete Mathematical Structure	3	1	0	4	3
MCA105	Business English and Communication	3	1	0	4	3
IVC(MC)101	Essential Studies for Professionals - I	3	1	0	4	0
Total of Theory					24	17
1st Semester (Practical)						
MCA191	Micro Programming and Architecture Laboratory	0	0	3	3	3
MCA192	C Programming Laboratory	0	0	3	3	3
MCA193	Data Structures with C Laboratory	0	0	3	3	3
Total of Practical					9	9
1st Semester (Sessional)						
IVC(MC)102	Skill Development for Professionals - I	2	1	0	3	0
MC181	Mandatory Additional Requirements (Co-Curricular/Extra-Curricular Activity)	0	0	0	0	1
Total of Sessional					3	1
Total of Semester					36	27

1st Year 2nd Semester

Course Code	Course Title	Total No. of Contact Hours				Total No. of Credits
		Lecture (L)	Tutorial (T)	Practical (P)	Total Hours	
2nd Semester (Theory)						
MCA201	Database Management System	3	1	0	4	4
MCA202	Object-Oriented Programming with Java	3	1	0	4	4
MCA203	Data Communication & Computer Networks	3	1	0	4	3
MCA204	Graphics and Multimedia	3	1	0	4	3
MCA205	Statistics and Numerical Techniques	3	1	0	3	3
IVC(MC)201	General Studies & Current Affairs - I	3	1	0	4	0
Total of Theory					24	17
2nd Semester (Practical)						
MCA291	Database Management System Laboratory	0	0	3	3	3
MCA292	Object-Oriented Programming with Java Laboratory	0	0	3	3	3
Total of Practical					6	6
2nd Semester (Sessional)						
IVC(MC)102	Competitive Aptitude Training - II	2	1	0	3	0
MCA281	Mandatory Additional Requirement (Co-Curricular/ Extra Curricular Activity)	0	1	0	1	1
Total of Sessional					4	1
Total of Semester					34	24

2nd Year 1st Semester

Course Code	Course Title	Total No. of Contact Hours				Total No. of Credits
		Lecture (L)	Tutorial (T)	Practical (P)	Total Hours	
3rd Semester (Theory)						
MCA301	Operating Systems and Systems Software	3	1	0	4	3
MCA303	Data Science and Data Analytics	3	1	0	4	4
MCA304	Software Engineering & TQM	3	1	0	4	4
MCA305	Values and Ethics	2	0	0	2	1
MCA306	Environment & Ecology	2	0	0	2	3
MCA(GS)301	General Studies & Current Affairs - III	3	1	0	4	0.5
Total of Theory					22	15.5
3rd Semester (Practical)						
MCA391	Minor Project	0	0	12	12	6
MCA392	Data Science & Data Analytics Laboratory	0	0	3	3	3
MCA393	Operating Systems Laboratory (Unix)	0	0	3	3	3
MCA394	Software Project Management Laboratory	0	0	3	3	3
Total of Practical					21	15
3rd Semester (Sessional)						
MCA381	Industrial Training	0	0	0	0	2
MCA(GS)381	Competitive Aptitude Training - III	2	1	0	3	0.5
MC381	Mandatory Additional Requirement (Co-Curricular/Extra Curricular Activity)	0	1	0	1	1
Total of Sessional					4	1.5
Total of Semester					46	33

2nd Year 2nd Semester

Course Code	Course Title	Total No. of Contact Hours				Total No. of Credits
		Lecture (L)	Tutorial (T)	Practical (P)	Total Hours	
4th Semester (Theory)						
MCA401 A/B/C/D	Elective - I	3	1	0	4	3
MCA402 A/B/C	Elective - II	3	1	0	4	3
MCA403	Operation Research & Optimisation Techniques	3	0	0	3	3
MCA405	Management & Accounting	2	0	0	2	2
MCA(GS)401	General Studies & Current Affairs - IV	3	1	0	4	0.5
Total of Theory					21	11.5
4th Semester (Practical)						
MCA491	Major Project	0	0	30	30	15
Total of Practical					30	15
4th Semester (Sessional)						
MCA481	Seminar	0	0	0	3	1
MCA(GS)481	General Studies & Current Affairs - IV	2	1	0	3	0.5
MC481	Mandatory Additional Requirement (Co-Curricular /Extra Curricular Activity)	0	1	0	1	1
Total of Sessional					7	2.5
Total of Semester					58	29

Elective No.	Course Code	Topic	Elective No.	Course Code	Topic
I	MCA401A	Distributed Database Management	II	MCA402A	Compiler Design
	MCA401B	Image Processing		MCA402B	Mobile Computing
	MCA401C	Parallel Programming		MCA402C	Embedded Systems
	MCA401D	Cloud Computing			

University of Engineering and Management, Kolkata

Syllabus for MCA Admission Batch 2021, 1st Semester

Course Name: Computer Organisation and Architecture **Credit: 3**

Course Code: MCA101 **Lecture Hours: 40**

Name of the Course: Computer Organization and Architecture	
Course Code: MCA101	**Semester: 1st**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 100
Tutorial: 1	Continuous Assessment: 100
Credit: 3	
Aim:	
1	To have a thorough understanding of the basic structure and operation of a digital computer.
2	To study the different communication methods with I/O devices and standard I/O interfaces.
3	To learn the architecture and assembly language programming of 8085 microprocessor.
Objective:	
1	Understanding Logic gates, flip flops and counters.
2	Clear Understanding of Computer Architecture.
3	Clear Understanding of Pipeline processing, RISC and CISC architectures.
4	Develop a base for advanced microprocessors.
Pre-Requisite:	
1.	Proficiency in basic Digital Electronics

Course Outcome:	
1.	Summarize the fundamental components of a basic computer system and its organization.
2.	Apply arithmetic and logical microoperations of binary number systems.
3.	Analyze control unit design and concept of pipelining.
4.	Classify memory hierarchy and examine numerical problems based on it.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	3	-	-	-	-	-	2	-	2	-	-	-
CO2	3	2	2	2	-	-	1	-	-	2	-	2	-	-	-
CO3	2	2	3	3	-	-	1	-	-	2	-	2	-	-	-
CO4	3	2	3	2	-	-	-	-	-	2	-	2	-	-	-

Module number	Topic	Sub-topics
1	**Structure of Computers and Computer Arithmetic**	Computer types, Functional units, Basic operational concepts, von Neumann Architecture, Bus Structures, Software, Performance, Multiprocessors and Multicomputer, Data representation, Fixed and Floating point, Error detection and correction codes Addition and Subtraction, Multiplication and Division algorithms, Floating-point Arithmetic Operations, Decimal arithmetic operations.
2	**Basic Computer Organization and Design**	Instruction codes, Computer Registers, Computer Instructions and Instruction cycle. Timing and Control, Memory-Reference Instructions, Input-Output and interrupt. Central processing unit: Stack organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Complex Instruction Set Computer (CISC) Reduced Instruction Set Computer (RISC), CISC vs RISC.
3	**Register Transfer, Micro-Operations and Micro-Programmed**	Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Micro-Operations, Logic Micro-Operations, Shift Micro-Operations, Arithmetic logic shift unit,

	Control	Control Memory, Address Sequencing, Micro-Program example, Design of Control Unit.
4	**Memory System:**	Memory Hierarchy, Semiconductor Memories, RAM(Random Access Memory), Read Only Memory (ROM), Types of ROM, Cache Memory, Performance considerations, Virtual memory, Paging, Secondary Storage, RAID.
5	**Input-Output:**	I/O interface, Programmed IO, Memory Mapped IO, Interrupt Driven IO, DMA.
6	**Multiprocessors**	Characteristics of multiprocessors, Interconnection structures, Inter Processor Arbitration, Interprocessor Communication and Synchronization, and Cache Coherence.

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ISSN/ISBN**	**Name of the Publisher**
M. Moris Mano	Computer System Architecture	3rd Ed	Pearson/PHI
Reference Books: **1. Carl Hamacher, Zvonks Vranesic, SafeaZaky (2002), Computer Organization, 5th edition, McGraw Hill, New Delhi, India.**			

Name of the Course: Micro Programming and Architecture Laboratory	
Course Code: MCA191	**Semester: 1st**
Duration: 12 Weeks.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Practical: 3	**Practical Sessional Internal continuous evaluation**: 100
Credit: 3	**Practical Sessional external examination:** 100

Aim:	
1	To have a thorough understanding of the basic structure and operation of a digital computer.
2	To study the different communication methods with I/O devices and standard I/O interfaces.
3	To learn the architecture and assembly language programming of 8085 microprocessor.
Objective:	
1	Understanding Logic gates, flip flops and counters.
2	Clear Understanding of Computer Architecture.
3	Clear Understanding of Pipeline processing, RISC and CISC architectures.
4	Develop a base for advanced microprocessors.
Pre-Requisite:	
1.	Proficiency in basic Digital Electronics
Course Outcome:	
CO1.	**Design and implement micro-operations** using VHDL to simulate basic control unit functions and instruction execution.
CO2.	**Model and simulate data path and control path architectures** using VHDL for various arithmetic, logical, and sequential operations.
CO3.	**Develop and verify VHDL-based programs** for memory interfacing, input/output operations, and processor functional blocks.
CO4.	**Apply debugging and simulation tools** to analyze, test, and validate VHDL-based micro-architectural designs for correctness and performance.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	2	-	-	-	-	-	-	1	-	-	-
CO2	3	3	3	-	3	-	-	-	-	-	-	2	-	-	-
CO3	3	3	3	-	3	-	-	-	-	-	-	2	-	-	-
CO4	3	3	2	2	3	-	-	-	-	-	-	3	-	-	-

Module Number	Topic	Suggested Assignments
1	**Structure of Computers and Computer Arithmetic**	1. Write a VHDL code to study and perform about logic gates. 2. Write a VHDL code to study and perform about De'Morgan's Theorem. 3. Write a VHDL code to study and perform about NAND and NOR as a universal gates. 4. Write a VHDL code to design and implement circuit that converts binary code to gray code.
2	**Basic Computer Organization and Design**	1. Write a VHDL code to study and perform about Half Adder and full Adder. 2. Write a VHDL code to study and perform about Half substractor and full substractor. 3. Write a VHDL code to design 3-bit odd/even parity generator and checker. 4. Write a VHDL code to study and perform about R-S and D flip flop. 5. Write a VHDL code to study and perform about J-K and T flip flop. 6. Write a VHDL code to study and perform about Master slave JK flip flop.
3	**Register Transfer, Micro-Operations and Micro-Programmed Control**	1. Write a VHDL code to realize Boolean functions using multiplexer. 2. Write a VHDL code to study and perform about Decoder and Demultiplexer. 3. Write a VHDL code to study the use of decoder for BCD to seven segment LED display. Write a VHDL code to study universal shift register

Course Name: Computer Programming with C Credit: 4

Course Code: MCA102 Lecture Hours: 40

Name of the Course: Computer Programming with C	
Course Code: MCA102	**Semester: 1st**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 100
Tutorial: 1	Continuous Assessment: 100
Credit: 4	
Aim:	
1	To gain Knowledge of Various aspects of algorithm development
2	To enhance Ability to identify qualities of a good solution
3	To implement learned algorithm design techniques and data structures to solve problems.
Objective:	
1	The fundamental design, analysis, and implementation of basic data structures.
2	Basic concepts in the specification and analysis of programs.
3	Principles for good program design, especially the uses of data abstraction.
4	Significance of algorithms in the computer field
Pre-Requisite:	
1.	Proficiency in one high-level programming language
Course Outcome:	
1.	will be able to develop simple applications in C using basic constructs
2.	will be able to design and implement applications in C using Arrays and Strings
3.	will be able to design and implement applications in C using Functions and Pointers
4.	will be able to develop applications in C using the Structures and design applications using sequential and random-access file processing.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	1	3	0	0	0	1	1	0	2	3	1	1
CO2	3	3	3	1	3	0	0	0	1	1	0	2	3	1	1
CO3	3	3	3	1	3	0	0	0	1	1	0	2	3	1	1
CO4	3	3	3	1	3	0	0	0	1	1	0	2	3	1	1

Module number	Topic	Sub-topics
1	**Basics of 'C' Programming**	Fundamentals of algorithms: Notion of algorithm, Notations used for assignment statements and basic control structures. Introduction to 'C': General structure of ' C' program, Header file, 'main ()' function. Fundamental constructs of 'C': Character set, tokens, keywords, Identifiers, Constants - number constants, character constants, string constants, Variables. Data types in 'C': Declaring variables, data type conversion. Basic Input and Output functions: input and output statements using printf(), scanf() functions. Assignments and expressions: simple assignment statements, arithmetic operators, shift operators, bitwise operators, sizeof operator
2	**Control structures**	Conditional statements: Relational operators, logical operators, if statement, if-else statements, nested if-else statements, if-else ladder, switch statement. Looping statements: while loop, do-while loop, for loop. Branching Statements: goto statement, use of 'break' and 'continue' statements.
3	**Arrays and structure**	3.1 Characteristics of an array, One dimension and two dimensional arrays, concept of multi-dimensional arrays. 3.2 Array declaration and Initialization. 3.3 Operations on Arrays. 3.4 Character and String input/output and String related operations. 3.5 Introduction and Features of Structures, Declaration and Initialization of Structures, array of structures. 3.6 Type def, Enumerated Data Type

Module number	Topic	Sub-topics
4	**Functions**	Concept and need of functions. Library functions: Math functions, String handling functions, other miscellaneous functions such as getchar(), putchar(),

		malloc(), calloc(). Writing User-defined functions - function definition, functions declaration, function call, scope of variables - local variables, global variables. Function parameters: Parameter passing- call by value & call by reference, function return values, function return types, declaring function return types, The 'return' statement. Recursive functions.
5	**Pointers**	Introduction to Pointers: Definition, use of pointers, '*' and '&' operators, declaring, initializing, accessing pointers. Pointer arithmetic. Pointer to array. Pointer and Text string. Function handling using pointers. Pointers to structure.
6	**File handling**	Creation of the new file Opening an existing file Reading from the file Writing to the file Deleting the file

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ ISSN/ ISBN**	**Name of the Publisher**
E. Balagurusamy	Programming in ANSI C	7th Ed	McGraw Hill Education
Reference Books: **Let us C** by *Yashavant Kanetkar*, 19th Edition., **The C Programming Language** by *Brian W. Kernighan* and *Dennis Ritchie*, 2nd Edition **Mastering C** by *K. R. Venugopal*			

Name of the Course: C Programming Laboratory	
Course Code: MCA192	**Semester: 1st**
Duration: 12 Weeks	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Practical: 3	Practical Sessional Internal continuous evaluation: 100
Credit: 3	Practical Sessional external examination: 100

Aim:	
1	To gain Knowledge of Various aspects of algorithm development
2	To enhance Ability to identify qualities of a good solution
3	To implement learned algorithm design techniques and data structures to solve problems.
Objective:	
1	The fundamental design, analysis, and implementation of basic data structures.
2	Basic concepts in the specification and analysis of programs.
3	Principles for good program design, especially the uses of data abstraction.
4	Significance of algorithms in the computer field
Pre-Requisite:	
1.	Proficiency in one high-level programming language
Course Outcome:	
1.	will be able to develop simple applications in C using basic constructs
2.	will be able to design and implement applications in C using Arrays and Strings
3.	will be able to design and implement applications in C using Functions and Pointers
4.	will be able to develop applications in C using the Structures and design applications using sequential and random-access file processing.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	1	3	0	0	0	1	1	0	2	3	1	1
CO2	3	3	3	1	3	0	0	0	1	1	0	2	3	1	1
CO3	3	3	3	1	3	0	0	0	1	1	0	2	3	1	1
CO4	3	3	3	1	3	0	0	0	1	1	0	2	3	1	1

Module Number	Topic	Suggested Assignments
2	**Control structures**	1. Write a C program to find sum and average of three numbers. 2. Write a C program to find the sum of individual digits of a given positive integer. 3. Write a C program to generate the first n terms of the Fibonacci sequence. metrices from the console, verifies if metrics multiplication is possible or not. Then multiplies the metrices and prints the 3rd metrics. 4. Write a C program to generate prime numbers between 1 to n. 5. Write a C program to Check whether given number is Armstrong Number or Not. 6. Write a C program to evaluate the algebraic expression (ax+b)/(ax-b). 7. Write a C program to check if the given number is perfect number? 8. Write a C program to check if given number is strong number? 9. Write a program to print your name without using any semicolon in the program. 10. Write a program to convert temperature in Celsius to Fahrenheit and vice-versa. 11. Write a C program to check whether a number is Palindrome or not. 12. Write a C program to find maximum between two numbers. 13. Write a C program to find maximum between three numbers. 14. Write a C program to check whether a number is negative, positive or zero. 15. Write a C program to check whether a number is divisible by 5 and 11 or not within the range 100 to 500. 16. Write a C program to check whether a number is even or odd. 17. Write a C program to check whether a year is a leap year or not. 18. Write a C program to check whether a character is alphabet or not. 19. Write a C program to input any alphabet and check whether it is vowel or consonant. 20. Write a C program to input any character and check whether it is an alphabet, digit or special character.
3	**Arrays and structure**	1. Write a program to store marks for n number of student in an array and print their marks. 2. Write a program which stores the marks of subject Mathematics and English of n number of students in an array and then prints their individual total marks. 3. Write a program to insert an element in an array in a particular position. 4. Write a program to delete an element from a particular position of an array.

		5. Write a program to convert a decimal number taken as input from user to corresponding binary number and store the result in an array. 6. Write a program to input a binary number in an array and convert into corresponding decimal number. 7. Write a program to find the smallest and the largest elements in an array. 8. Write a program for deleting duplicate elements in an array. 9. Write a program to search for a particular element in an array. 10. Write a program to sort n elements (ascending order). 11. Write a program to find second highest number from the array without using sorting. 12. Write a program to perform addition and subtraction between two matrices. 13. Write a program to transpose a matrix. 14. Write a program to add the elements of each row and each column of a matrix. 15. Write a program to perform the multiplication of two matrices. 16. Write a program to check whether a matrix is identity matrix or not. 17. Write a program to check whether a matrix is sparse matrix or not 18. Write a C program to create a structure named company which has name, address, phone and no Of Employee as member variables. Read name of company, its address, phone and no Of Employee. Finally display these members‟ value. 19. Define a structure “complex” (typedef) to read two complex numbers and perform addition, subtraction of these two complex numbers and display the result. 20. Write a C program to read Roll No, Name, Address, and Age marks of 12 students in the BCT class and display the details from the function.
4	**Functions**	1. Write a C program to add, subtract, multiply and divide two integers using a user-defined type function with return type. 2. Write a C program to calculate sum of first 20 natural numbers using recursive function. 3. Write a C program to generate Fibonacci series using recursive function. 4. Write a C program to swap two integers using call by value and call by reference methods of passing arguments to a function. 5. Write a C program to find sum of digits of the number using Recursive Function. 6. Write a C program to read an integer number and print the reverse of that number using recursion. 7. Write a C program to find maximum and minimum between two numbers using functions. 8. Write a C program to check whether a number is even or odd using functions. 9. Write a C program to check whether a number is

		prime, Armstrong or perfect number using functions. Write a C program to find power of any number using recursion.
5	**Pointers**	1. Write a C program to find the sum of all the elements of an array using pointers. 2. Write a C program to swap value of two variables using pointer. 3. Write a C program to add two numbers using pointers. 4. Write a C program to input and print array elements using pointer. 5. Write a C program to copy one array to another using pointer. 6. Write a C program to swap two arrays using pointers. 7. Write a C program to reverse an array using pointers. 8. Write a C program to search for an element in array using pointers. 9. Write a C program to add two 2 X 2 matrix using pointers. 10. Write a C program to multiply two 2 X 2 matrix using pointers. 11. Write a C program to find length of string using pointers. 12. Write a C program to copy one string to another using pointer. 13. Write a C program to concatenate two strings using pointers. 14. Write a C program to compare two strings using pointers. 10. Write a C program to find a substring from a given string using pointers.
6	**File handling**	1. Write a C Program to list all files and sub-directories in a directory. 2. Write a C Program to count number of lines in a file. 3. Write a C Program to print contents of file. 4. Write a C Program to copy contents of one file to another file. 5. Write a C Program to merge contents of two files into a third file. 6. Write a C program to delete a file.

Course Name: Data Structure with C Credit: 4

Course Code: MCA103 Lecture Hours: 40

Name of the Course: Data Structure with C	
Course Code: MCA103	**Semester: 1st**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 100
Tutorial: 1	Continuous Assessment: 100
Credit: 4	
Aim:	
1	To gain Knowledge of Various aspects of algorithm development
2	To enhance Ability to identify qualities of a good solution
3	To implement learned algorithm design techniques and data structures to solve problems.
Objective:	
1	The fundamental design, analysis, and implementation of basic data structures.
2	Basic concepts in the specification and analysis of programs.
3	Principles for good program design, especially the uses of data abstraction.
4	Significance of algorithms in the computer field
Pre-Requisite:	
1.	Proficiency in one high level programming language
Course Outcome:	
1.	On completion of this course students are expected to learn various data structures, their usages, merits andlimitations.
2.	On completion of this course students are expected to design and analyze various algorithms.
3.	On completion of this course students are expected to do a comparative analysis among different data structuresand decide on the appropriate data structure to be used in a given scenario.
4.	On completion of this course students are expected to acquire adequate knowledge and skills to solve a real lifesoftware problem.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	2	1	2	-	-	-	-	-	-	1	1	2	2
CO2	3	3	1	2	1	-	-	-	-	-	-	2	3	2	1
CO3	3	2	2	3	2	-	-	-	-	-	-	1	3	2	1
CO4	2	2	3	2	2	-	-	-	-	-	-	2	3	2	2

Module number	Topic	Sub-topics
1	**Algorithm Concept**	Algorithm concept, Time Complexity, Space Complexity, Running Time– Worst Case, Best Case, Average Case, time space trade-off, Algorithm Efficiency-Linear loops, Logarithmic loops, Nested loops, Time complexity comparison- Polynomial vs Exponential, Algorithm Notations-Big O , Big Omega, Theta Notation
2	**Introduction to Data Structure, Array**	Program Efficiency, Data Structure-definition, usage, examples, Selection of Appropriate Data Structure, Data Structure-some terminologies, Classification of Data Structure, Fundamental difference between Linear and Non-linear Data Structure with examples, Operations on Linear Data Structure Introduction to Linear Data Structure-Array, 1D, 2D arrays, Row/Column major representation, sparse matrix
3	**Linear Data Structure-Linked List**	Linked List-Introduction, Representation, Memory Allocation, Types- Singly, circular, doubly, doubly & circular, Operations on various linked lists-Count, Traverse/Display, Search, Insert, Delete
4	**Linear Data Structure-Stack**	Introduction, Stack Operations – Push, Pop, Peek, Representation of Stack (Array, Linked List), Application of Stack: Reversing a list, Parentheses checker, Conversion of an infix expression into a postfix expression, Evaluation of a postfix expression, Conversion of an infix expression into a prefix Expression, Evaluation of a prefix expression, Recursion, Tower of Hanoi
5	**Linear Data Structure-Queue**	Introduction, Queue Operations – Enqueue, Dequeue, Peep, Representation of Queue (Array, Linked List), Types of Queues- Circular Queue, Deque, Priority Queue, Multiple Queue; Various operations (Enqueue, Dequeue, Peep) on the above mentioned queues-Both iterative & recursive implementation; Application of Queue
6	**Searching & Sorting**	Searching- Types of Searching (Linear Search, Binary Search, Interpolation Search), Comparison among various Searching techniques Sorting-Types, Methods (Bubble Sort, Insertion Sort, Selection Sort, Quick Sort, Merge Sort), Technique, Explanation, Algorithm and Examples on various sorting methods, Comparison of various sorting algorithms in terms of time complexity (Average case, Worst case)

List of Books Text Books:			
Name of Author	**Name of Author**	**Name of Author**	**Name of Author**
Reema Thareja	Reema Thareja	Reema Thareja	Reema Thareja
Reference Books:			
Tenenbaum	Data Structure Using C & C++	2nd Ed	PEI
Kruse, Tondo & Leung	Data Structures & Program Design in C	2nd Ed	PHI
Loudan	Mastering Algorithms With C		SPD/O'REILLY
Radhaganesan	C and Data Structures		Scitech Publications

Name of the Course: **Data Structures with C Laboratory**	
Course Code: MCA193	**Semester: 1st**
Duration: 12 Weeks.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Practical: 3	Practical Sessional Internal continuous evaluation: 100
Credit: 3	Practical Sessional external examination: 100

Aim:	
1	To gain Knowledge of Various aspects of algorithm development
2	To enhance the Ability to identify qualities of a good solution
3	To implement learned algorithm design techniques and data structures to solve problems.
Objective:	
1	The fundamental design, analysis, and implementation of basic data structures.
2	Basic concepts in the specification and analysis of programs.
3	Principles for good program design, especially the uses of data abstraction.
4	Significance of algorithms in the computer field
Pre-Requisite:	
1.	Proficiency in one high-level programming language
Course Outcome:	
1.	On completion of this course, students are expected to learn various data structures, their usages, merits andlimitations.
2.	On completion of this course, students are expected to design and analyze various algorithms.
3.	On completion of this course students are expected to do a comparative analysis among different data structuresand decide on the appropriate data structure to be used in a given scenario.
4.	On completion of this course students are expected to acquire adequate knowledge and skills to solve a real-lifesoftware problem.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	1	2	-	-	-	1	-	-	1	1	2	1
CO2	3	3	2	2	3	-	-	-	1	-	-	2	3	2	1
CO3	3	3	2	3	3	-	-	-	1	-	-	2	3	2	2
CO4	2	2	3	2	2	-	-	-	2	2	2	3	3	2	2

Module Number	Topic	Suggested Assignments
2	**Control structures**	1. Write a C program to find sum and average of three numbers. 2. Write a C program to find the sum of individual digits of a given positive integer. 3. Write a C program to generate the first n terms of the Fibonacci sequence. metrices from the console, verifies if metrics multiplication is possible or not. Then multiplies the metrices and prints the 3rd metrics. 4. Write a C program to generate prime numbers between 1 to n. 5. Write a C program to Check whether given number is Armstrong Number or Not. 6. Write a C program to evaluate the algebraic expression (ax+b)/(ax-b). 7. Write a C program to check if the given number is perfect number? 8. Write a C program to check if given number is strong number? 9. Write a program to print your name without using any semicolon in the program. 10. Write a program to convert temperature in Celsius to Fahrenheit and vice-versa. 11. Write a C program to check whether a number is Palindrome or not. 12. Write a C program to find maximum between two numbers. 13. Write a C program to find maximum between three numbers. 14. Write a C program to check whether a number is negative, positive or zero. 15. Write a C program to check whether a number is divisible by 5 and 11 or not within the range 100 to 500. 16. Write a C program to check whether a number is even or odd. 17. Write a C program to check whether a year is a leap year or not. 18. Write a C program to check whether a character is alphabet or not. 19. Write a C program to input any alphabet and check whether it is vowel or consonant. 20. Write a C program to input any character and check whether it is an alphabet, digit or special character.
3	**Arrays and structure**	1. Write a program to store marks for n number of student in an array and print their marks. 2. Write a program which stores the marks of subject Mathematics and English of n number of students in an array and then prints their individual total marks. 3. Write a program to insert an element in an array in a particular position. 4. Write a program to delete an element from a particular position of an array.

		5. Write a program to convert a decimal number taken as input from user to corresponding binary number and store the result in an array. 6. Write a program to input a binary number in an array and convert into corresponding decimal number. 7. Write a program to find the smallest and the largest elements in an array. 8. Write a program for deleting duplicate elements in an array. 9. Write a program to search for a particular element in an array. 10. Write a program to sort n elements (ascending order). 11. Write a program to find second highest number from the array without using sorting. 12. Write a program to perform addition and subtraction between two matrices. 13. Write a program to transpose a matrix. 14. Write a program to add the elements of each row and each column of a matrix. 15. Write a program to perform the multiplication of two matrices. 16. Write a program to check whether a matrix is identity matrix or not. 17. Write a program to check whether a matrix is sparse matrix or not 18. Write a C program to create a structure named company which has name, address, phone and no Of Employee as member variables. Read name of company, its address, phone and no Of Employee. Finally display these members‟ value. 19. Define a structure “complex” (typedef) to read two complex numbers and perform addition, subtraction of these two complex numbers and display the result. 20. Write a C program to read Roll No, Name, Address, and Age marks of 12 students in the BCT class and display the details from the function.
4	**Functions**	1. Write a C program to add, subtract, multiply and divide two integers using a user-defined type function with return type. 2. Write a C program to calculate sum of first 20 natural numbers using recursive function. 3. Write a C program to generate Fibonacci series using recursive function. 4. Write a C program to swap two integers using call by value and call by reference methods of passing arguments to a function. 5. Write a C program to find sum of digits of the number using Recursive Function. 6. Write a C program to read an integer number and print the reverse of that number using recursion. 7. Write a C program to find maximum and minimum between two numbers using functions. 8. Write a C program to check whether a number is even or odd using functions. 9. Write a C program to check whether a number is

		prime, Armstrong or perfect number using functions. 10. Write a C program to find power of any number using recursion.
5	**Pointers**	1. Write a C program to find the sum of all the elements of an array using pointers. 2. Write a C program to swap value of two variables using pointer. 3. Write a C program to add two numbers using pointers. 4. Write a C program to input and print array elements using pointer. 5. Write a C program to copy one array to another using pointer. 6. Write a C program to swap two arrays using pointers. 7. Write a C program to reverse an array using pointers. 8. Write a C program to search for an element in array using pointers. 9. Write a C program to add two 2 X 2 matrix using pointers. 10. Write a C program to multiply two 2 X 2 matrix using pointers. 11. Write a C program to find length of string using pointers. 12. Write a C program to copy one string to another using pointer. 13. Write a C program to concatenate two strings using pointers. 14. Write a C program to compare two strings using pointers. 15. Write a C program to find a substring from a given string using pointers.
6	**File handling**	1. Write a C Program to list all files and sub-directories in a directory. 2. Write a C Program to count number of lines in a file. 3. Write a C Program to print contents of file. 4. Write a C Program to copy contents of one file to another file. 5. Write a C Program to merge contents of two files into a third file. 6. Write a C program to delete a file.

Course Name: Discrete Mathematical Structure Credit: 3

Course Code: MCA104 Lecture Hours: 40

Name of the Course: Discrete Mathematical Structure	
Course Code: MCA104	**Semester: 1st**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 100
Tutorial: 1	Continuous Assessment: 100
Credit: 3	
Aim:	
Sl. No.	
1	To provide a strong foundation in discrete mathematical structures essential for computing applications.
2	To develop analytical and logical reasoning skills for problem-solving in computer science.
3	To enable students to apply mathematical concepts in areas like algorithms, graph theory, and automata theory.
Objective:	
Sl. No.	
1	To understand and apply fundamental concepts of set theory, relations, and functions in computing.
2	To develop proficiency in mathematical logic, combinatorics, and recurrence relations for algorithm design.
3	To explore graph theory and its applications in networks, trees, and optimization algorithms.
4	To introduce automata theory, grammars, and fuzzy logic for theoretical computation models.
Pre-Requisite:	
Sl. No.	
1.	Basic knowledge of mathematics, including algebra and logic.
Course Outcome:	
1.	Apply set theory, relations, and functions to model real-world computing problems.
2.	Analyze and solve problems using mathematical logic, combinatorial techniques, and recurrence relations.
3.	Utilize graph theory concepts and algorithms for solving computational and network-related problems.
4.	Design and analyze finite automata, grammars, and fuzzy logic systems for computational applications.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	1	2	1	0	0	1	1	1	2	3	2	2
CO2	3	3	2	2	2	1	0	1	1	1	1	2	2	3	2
CO3	3	3	3	2	3	2	1	1	1	1	2	2	3	2	2
CO4	3	2	3	1	3	1	0	1	1	1	2	2	3	2	3

Module number	Topic	Sub-topics
1	**Foundations of Discrete Mathematics**	Set Theory: Foundations, Mapping (Bijective, Surjective, Injective) Relations: Equivalence Relations, Posets, Lattices
2	**Mathematical Logic and Combinatorics**	Mathematical Induction Propositional Logic, Logical Equivalence Permutations and Combinations
3	**Linear Data Structure-Linked List**	Generating Functions Recurrence Relations
4	**Graph Theory and Its Applications**	Concepts of Graph Theory: Sub-graphs, Cyclic Graphs Trees, Spanning Trees, Binary Trees Graph Algorithms: Kruskal's, Prim's, Dijkstra's, Floyd-Warshall's, DFS, BFS Graph Isomorphism and Homomorphism
5	**Automata Theory and Fuzzy Systems**	Finite Automata: Construction & Conversion of NFA, DFA, State Minimization Mealy & Moore Machines Grammars: Type 0, 1, 2, 3 Fuzzy Sets: Basic Properties

Books:

1. Theory of Computer Science, Mishra & Chandrasekharan, PHI
2. Discrete Mathematics for Comp. Scientists & Mathematicians, Mott, Kandel & Baker, PHI
3. Discrete Mathematical Structure, C.L.Liu,TMH
4. Discrete Mathematical Structure, G.S.RAO,New Age International
5. Discrete Mathematics With Applications, Rosen, TMH, 5th Ed
6. Discrete Mathematics, Ash & Ash, MH.
7. Discrete Mathematical Structure, Somasundaram, PHI
8. Discrete Mathematical Structure, Dubey, EXCEL BOOKS
9. Discrete Mathematics, Iyenger, VIKAS
10. Discrete Structure and Graph Theory, Bhisma Rao,Scitech

Course Name: Business English and Communication Credit: 3

Course Code: MCA105 Lecture Hours: 33

Name of the Course: Business English and Communication	
Course Code: MCA105	**Semester: 1st**
Duration: 33 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 100
Tutorial: 1	Continuous Assessment: 100
Credit: 3	

Aim:	
Sl. No.	
1	Making the students industry-ready.
2	Making the students relevant in the contemporary society.
3	Making the students prepared to analyze and solve problems through listening, speaking, reading and writing skills.

Objective:	
Sl. No.	
1	To develop effective business writing and communication skills.
2	To enhance oral communication and presentation abilities among students.
3	To help students learn to prepare various business documents and technical reports.
4	To improve listening and reading comprehension.

Pre-Requisite:	
Sl. No.	
1.	Basic English Proficiency, Listening and Speaking Skills, Reading and Writing Skills, Academic and Social Contexts, and Familiarity with Corporate Ethics.

Course Outcome:	
1.	Achieve competence in grammar, syntax, and vocabulary fundamentals.
2.	Effectively communicate in academic and social contexts.
3.	Develop readiness for the industry and understand corporate ethics.
4.	Acquire basic proficiency in English encompassing reading, listening, comprehension, writing, and speaking skills.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	2	-	2	-	1	2	3	-	3	3	-	-
CO2	3	3	-	2	-	3	1	2	3	3	-	3	3	-	-
CO3	2	3	-	2	-	3	1	3	2	3	-	3	3	-	-
CO4	3	3	-	2	-	2	-	2	2	3	-	3	3	-	-

Module number	Topic	Sub-topics
1	**Introduction to Business Communication.**	- Importance of effective communication in business. - Types of business communication: Internal and External. - Communication process and barriers. - Strategies for effective communication. - Traditional and digital communication channels. - Effective use of email, memos, and business letters. - Communication through social media and professional networks.
2	**Writing Skills Development**	- Formats and styles of business letters. - Writing formal and informal business letters. - Common types of business letters: Inquiry, Complaint, Application, and Appreciation. - Structure of technical reports. - Writing abstracts, executive summaries, and conclusions. - Incorporating visuals and data in reports.
3	**Oral Communication Skills**	- Preparing and delivering business presentations. - Using multimedia in presentations. - Techniques for effective public speaking. - Prepared speech exercises. - Extempore speech practice. - Role-playing business scenarios.
4	**Listening and Reading Skills**	- Importance of active listening in business. - Techniques for improving listening skills. - Listening comprehension exercises. - Developing reading comprehension. - Strategies for effective reading. - Comprehension tests and exercises.
5	**Practical Communication Applications**	- Principles of organizing written material. - Structuring content for clarity and impact. - Editing and proofreading techniques - Designing effective posters for business presentations. - Visual and textual balance. - Presenting posters in professional settings.
6	**Practical Communication Skill Development**	- Interactive sessions on negotiation and persuasion. - Group discussions and teamwork exercises

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ISSN/ ISBN**	**Name of the Publisher**
R C Sharma and Krishna Mohan	Business Correspondence & Report Writing	ISBN 978-9385965050 (5th ed)	McGraw Hill Education
Reference Books:			
Matthukutty Monippally	Business Communication Strategies	ISBN 978-0070435773	McGraw Hill Education
K.R. Lakshminarayanan	English for Technical Communication	Volume 1 & 2 Combined Edition	SCITECH PUBLICATIONS (INDIA) PVT LTD
Asha Kaul	Business Communication	Second Edition	PHI Learning
Dr. Anjali Ghanekar	Communication Skills for Effective Management	ISBN 978-8186314500 (19th ed)	Everest Publishing House

Course Name: Essential Studies for Professionals–I Credit: 0

Course Code: IVC(MC)101 Lecture Hours: 20

Name of the Course: Essential Studies for Professionals–I	
Course Code: IVC(MC)101	**Semester: 1st**
Duration: 20 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 2	End Semester Exam: 100
Tutorial: 0	Continuous Assessment: 100
Credit: 0	
Aim:	
1	To enhance quantitative aptitude and logical reasoning for effective problem-solving in professional environments.
2	To develop oral, listening, and reading communication skills for workplace interactions.
3	To equip students with practical communication skills, including professional correspondence and presentations.
Objective:	
1	To strengthen quantitative and logical reasoning for analytical decision-making.
2	To improve verbal and non-verbal communication skills, including articulation and clarity.
3	To enhance listening and reading comprehension for better information processing.
4	To develop practical communication techniques, such as presentations, business communication, and public speaking.
Pre-Requisite:	
1.	Basic understanding of English language and fundamental mathematical concepts.

Module number	**Topic**	**Sub-topics**
1	**Quantitative Aptitude**	**<u>A. Quant Foundation</u> 1. Number System(Chapter 1)** **2. HCF and LCM (Chapter 2)** **3. Decimal Fractions**

		(Chapter 3) **4. Simplification** **(Chapter 4)** **5. Square roots and cube roots (Chapter 5)** **6. Percentage** **(Chapter 11)-** Basic concept of percentage & its shortcut rules & their applications. **7. Ratio and Proportion (Chapter 13)-** Basic concept of Ratio & Proportion, Shortcut tricks & their applications. **8. Partnership** **(Chapter 14)** concept, rules & Applications, Percentage Advanced problems & shortcuts. **Profit & Loss (Chapter 12)-** Basic concept, formulae, shortcut tricks & their application.
2	**Logical Reasoning**	**1. Coding and Decoding (Chapter 4)** i. Conditional Coding, ii. Word-Pattern Coding, iii. Chinese Coding, **2. Direction Sense Test (Chapter 8)** i. Direction Sense Test, ii. Direction Distance Test, iii. Shadow based Questions. 3. **Series Completion (Chapter 1)** i. Alphabet Series, ii. Random Series, iii. Number Series, iv. Letter Gap, v. Missing Number Series, vi. Series Completion **4. Blood Relations (Chapter 5)** i. Family Tree Questions ii. Indication Type BR, iii. Coding Blood Relations, iv. Miscellaneous Blood Relations
3	**Oral Communication Skills**	- Preparing and delivering business presentations. - Using multimedia in presentations. - Techniques for effective public speaking. - Prepared speech exercises. - Extempore speech practice. - Role-playing business scenarios.
4	**Listening and Reading Skills**	- Importance of active listening in business. - Techniques for improving listening skills. - Listening comprehension exercises. - Developing reading comprehension. - Strategies for effective reading. - Comprehension tests and exercises.
5	**Practical Communication Applications**	- Principles of organizing written material. - Structuring content for clarity and impact. - Editing and proofreading techniques

		- Designing effective posters for business presentations. - Visual and textual balance. - Presenting posters in professional settings.
6	**Practical Communication Skill Development**	- Interactive sessions on negotiation and persuasion. - Group discussions and teamwork exercises

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ISSN/ISBN**	**Name of the Publisher**
R C Sharma and Krishna Mohan	Business Correspondence & Report Writing	ISBN 978-9385965050 (5th ed)	McGraw Hill Education
Reference Books:			
Matthukutty Monippally	Business Communication Strategies	ISBN 978-0070435773	McGraw Hill Education
K.R. Lakshminarayanan	English for Technical Communication	Volume 1 & 2 Combined Ed.	Scitech Publications (India) Pvt Ltd
Asha Kaul	Business Communication	Second Edition	PHI Learning
Dr. Anjali Ghanekar	Communication Skills for Effective Management	ISBN 978-8186314500 (19th ed)	Everest Publishing House

Name of the Course: Skill Development for Professionals - I		
Course Code: IVC(MC)181		**Semester: 1st**
Duration: 20 Hrs.		**Maximum Marks: 100**
Teaching Scheme		**Examination Scheme**
Theory: 2		End Semester Exam: 100
Tutorial: 0		Continuous Assessment: 100
Credit: 0		
Aim:		
1	To develop verbal reasoning, analytical thinking, and problem-solving skills essential for competitive exams and job recruitment.	
2	To enhance English language proficiency, focusing on grammar, comprehension, and professional communication.	
3	To build expertise in data interpretation, improving the ability to analyze and solve quantitative problems efficiently.	
Objective:		
1	To strengthen English grammar, vocabulary, and comprehension skills for competitive aptitude tests.	
2	To develop logical and analytical reasoning through structured problem-solving techniques.	
3	To enhance proficiency in data interpretation, including tabular and graphical analysis.	
4	To improve formal communication skills, such as official letter writing, for professional settings.	
Pre-Requisite:		
1.	Basic understanding of English grammar and elementary mathematical reasoning.	

Module number	Topic	Sub-topics
1	**Verbal English-1:**	1) Introduction of Parts of speech: Introduction, Brief discussion of Parts of speech 2) What are nouns, Kinds of Nouns, Rules & Applications. 3) Definition of Pronoun, Examples, Rules & Application 4) Definition of Subject Verb Agreement, Rules and Examples. 5) Basic Application of Vocabulary (Synonyms and Antonyms) 6) Reading Comprehension.

		7) Official Letter Writing **Parts of Speech** 1. **Identify Parts of Speech**: o Provide a paragraph andask students to identify and label each word's partof speech (noun, verb, adjective, adverb, pronoun, preposition,conjunction, interjection). 2. **Parts of Speech Matching**: o Create a list of words and a list of parts of speech. Ask students to matcheach word to the correct part of speech. 3. **Parts of Speech Sentences**: Ask students to write sentences using specific parts of speech (e.g., writea sentence with at least one noun, one verb, one adjective, and one adverb). **Nouns** 1. **Noun Identification**: o Provide a list of sentences and ask students to underline or highlight the nouns. 2. **Types of Nouns**: o Provide examples of common, proper, abstract, and collective nouns. Ask students to classify given nouns into thesecategories. 3. **Noun Plurals**: o Give a list of singular nouns and ask students to write their plural forms. **Pronouns** 1. **Pronoun Replacement**: o Provide sentences with nouns and ask students to replace the nouns withappropriate pronouns. 2. **Pronoun Agreement**: o Create sentences withpronouns and ask studentsto correct any errors inpronoun-antecedent agreement. 3. **Types of Pronouns**: Provide a list of pronounsand ask students to classify them intocategories (personal, possessive, reflexive,demonstrative, interrogative, relative, indefinite). **Synonyms** 1. **Synonym Matching**: o Provide a list of words anda list of synonyms. Ask students to match each word with its synonym. 2. **Synonym Sentences**: o Give sentences with underlined words and ask students to rewrite thesentences using synonyms for the underlined words.

		3. **Synonym Stories**: o Ask students to write a short story using a list of provided words and their synonyms. **Antonyms** 1. **Antonym Matching**: o Provide a list of words anda list of antonyms. Ask students to match each word with its antonym. 2. **Antonym Sentences**: o Give sentences with underlined words and ask students to rewrite thesentences using antonyms for the underlined words. 3. **Antonym Pairs**: Ask students to create a list of ten words and writetheir antonyms next tothem.
2	**Data Interpretation level-I**	**Calculating Totals and Averages:** Provide a table with sales data over several months. Ask students to calculate the total sales and average sales for each month. **Comparing Data:** Provide a table with data on two or more products or categories. Ask students to compare the data and determine which product/category performed better based on different criteria (e.g., sales, growth rate).

Syllabus for MCA Admission Batch 2021, 2nd Semester

Course Name: Database Management Systems Credit: 4

Course Code: MCA201 Lecture Hours: 40

Name of the Course: Database Management Systems	
Course Code: MCA201	**Semester: 2nd**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 4	
Aim:	
1.	To gain Knowledge of technology used to manage data from a database
2.	To enhance Ability to identify Data into information, Information into knowledge and Knowledge to the action
3.	To gain Understanding of ORACLE software
Objective:	
1.	This course introduces the core principles and techniques required in the design and implementation of database systems.
2.	This course focus on relational database management systems, including database design theory: E-R modeling, data definition and manipulation languages, database security and administration.
3.	It covers essential DBMS concepts such as: Transaction Processing, Concurrency Control and Recovery
4.	It provides students with theoretical knowledge and practical skills in the use of databases and database management systems in information technology applications.
Pre-Requisite:	
1.	Concepts of computer programming (like programming in C --Files concepts).
Course Outcome:	
1.	Understand the basic concepts and the applications of database systems.
2.	Master the basics of SQL and construct queries using SQL.
3.	Understand the relational database design principles.
4.	Familiar with the basic issues of transaction processing and concurrency control.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	1	1	1	0	0	0	1	0	2	2	3	1
CO2	3	3	2	2	3	1	0	0	0	1	0	2	3	3	1
CO3	3	3	3	1	3	1	0	0	0	1	0	2	3	3	1
CO4	3	3	3	2	3	1	0	0	0	1	0	2	3	3	1

Module number	Topic	Sub-topics
1	**Introduction: Database System Applications**	Database System Applications, Purpose ofDatabase Systems, View of Data, DatabaseLanguages – DDL, DML, Relational Databases, Database Design, Data Storage and Querying, Transaction Management, Database Architecture, Data Mining and Information Retrieval, Specialty Databases, Database Users and Administrators, History of Database Systems. Introduction to Data base design: Database Design andER diagrams, Entities, Attributes and Entitysets, Relationships and Relationship sets, Additional features of ER Model, Conceptual Design with the ER Model, Conceptual Design for Large enterprises.
	Relational Model	Introduction to the Relational Model, Integrity Constraints over Relations, Enforcing Integrity constraints, Querying relational data, Logical data base Design: ER to Relational, Introduction to Views, Destroying /Altering Tables and Views.
2	**Relational Algebraand Calculus**	Preliminaries, Relational Algebra, Relational calculus – Tuple relational Calculus, Domain relational calculus, Expressive Power of Algebra and calculus. SQL: Queries, Constraints, Triggers: Formof Basic SQL Query, UNION,INTERSECT, and EXCEPT, Nested Queries, Aggregate Operators,NULL values Complex Integrity All JNTU World Constraints in SQL, Triggers and Active Data bases, Designing ActiveDatabases.
3	**Schema Refinement and Normal Forms**	Introduction to Schema Refinement,Functional Dependencies - Reasoning about FDs, Normal Forms, Properties of Decompositions, Normalization, Schema Refinement in Database Design, Other Kinds of Dependencies.
4	**Transaction Management**	Transactions, Transaction Concept, A Simple Transaction Model, Storage Structure, Transaction Atomicity and Durability, Transaction Isolation, Serializability, Transaction

		Isolation and Atomicity Transaction Isolation Levels, Implementation of Isolation Levels.
	Concurrency Control	Lock–Based Protocols, Multiple Granularity, Timestamp-Based Protocols, Validation-Based Protocols, Multiversion Schemes. Recovery System-Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm, Buffer Management, Failure with loss of nonvolatile storage, Early Lock Releaseand Logical Undo Operations, Remote Backup systems.
5	**Storage and Indexing**	Overview of Storage and Indexing: Data on External Storage, File Organization and Indexing, Index Data Structures, Comparison of File Organizations. Tree- Structured Indexing: Intuition for tree Indexes, Indexed Sequential Access Method (ISAM)
	B+ Trees	A Dynamic Index Structure, Search, Insert,Delete. Hash- Based Indexing: StaticHashing, Extendible hashing, LinearHashing, Extendible vs. Linear Hashing.

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ ISSN/ ISBN**	**Name of the Publisher**
Abraham Silberschatz, Henry F.Korth, et al.	Database System Concepts	Seventh Edition	McGraw-Hill
Reference Books:			
Raghu Ramakrishnan and Johannes Gehrke	Database Management Systems	ISE	McGraw-Hill

Name of the Course: Database Management System Laboratory	
Course Code: MCA291	**Semester: 2nd**
Duration: 12 Weeks.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Practical: 3	Practical Sessional Internal continuous evaluation: 100
Credit: 3	Practical Sessional external examination: 100

Aim:	
1.	To gain Knowledge of technology used to manage data from a database
2.	To enhance Ability to identify Data into information, Information into knowledge and Knowledge to the action
3.	To gain Understanding of ORACLE software
Objective:	
1.	This course introduces the core principles and techniques required in the design and implementation of database systems.
2.	This course focus on relational database management systems, including database design theory: E-R modeling, data definition and manipulation languages, database security and administration.
3.	It covers essential DBMS concepts such as: Transaction Processing, Concurrency Control and Recovery
4.	It provides students with theoretical knowledge and practical skills in the use of databases and database management systems in information technology applications.
Pre-Requisite:	
1.	Concepts of computer programming (like programming in C -Files concepts).
Course Outcome:	
CO1.	**Design and implement relational databases** using SQL, incorporating tables, keys, constraints, and normalization principles to ensure data integrity and efficiency.
CO2.	**Apply SQL queries** to perform data retrieval, manipulation, and transaction control operations for solving real-world data problems.
CO3.	**Develop advanced SQL features**, including joins, subqueries, views, and indexing to optimize performance and functionality.
CO4.	**Demonstrate the use of PL/SQL** for writing procedures, functions, cursors, and triggers to automate and enhance database operations.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	1	1	1	0	0	0	1	0	2	2	3	1
CO2	3	3	2	2	3	1	0	0	0	1	0	2	3	3	1

CO3	3	3	3	1	3	1	0	0	0	1	0	2	3	3	1
CO4	3	3	3	2	3	1	0	0	0	1	0	2	3	3	1

Module number	Topic	Sub-topics
1	**Introduction: Database System Applications**	Annexure – I (SQLQuery based Lab – Assignments) Assignment – 1: Design E-R Diagrams for Different casestudies
	Relational Model	Assignment – 2: Case Studies usingbasic SQL RelationalAlgebra Operations
2	**Relational Algebraand Calculus**	Assignment – 3: Case Studies usingbasic SQL RelationalAlgebra Operations
3	**Schema Refinement and Normal Forms**	Assignment – 4: SQL based assignment on different normal forms.
4	**Transaction Management**	Assignment – 5: SQL-based assignment onTransaction Management
	Concurrency Control	Assignment – 6: SQL-based assignment on Transaction Management
5	**Storage and Indexing**	Assignment – 6: Implement B+ tree in Python
	B+ Trees	

Annexure – I (SQL based Lab – Assignments)

Assignment – 1:

Consider the following relational schema for the Office of the Controller of Examinations Application. Student (Rollno, Name, Dob, Gender, Doa, Bcode);
Implement a check constraint for Gender
Date of Admission
Branch (Bcode, Bname, Dno); Department (Dno, Dname);
Course (Ccode, Cname, Credits, Dno); Branch_Course (Bcode, Ccode, Semester);
Enrolls (Rollno, Ccode, Sess, Grade);
For Example,

SESS can take values ‘APRIL 2013’, ‘NOV 2013’

Implement a check constraint for grade Value Set (‘S’, ‘A’, ‘B’, ‘C’, ‘D’, ‘E’, ‘U’);

Students are admitted to Branches and they are offered by Departments. A branch is offered by only one department.

Each branch has a set of Courses (Subjects). Each student must enroll during a semester. Courses are offered by Departments. A course is offered only by one department. If a student

is unsuccessful in a course he/she must enroll for the course during next session. A student has successfully completed a course if the grade obtained by is from the list (A, B, C, D, and E).

A student is unsuccessful if he/she have grade 'U' in a course. Primary Keys are underlined.
Questions
These are questions for assignment 1
Question (A): Develop a SQL query to list details of Departments that offer more than 3 branches.
Question (B): Develop a SQL query to list the details of Departments that offer more than 6 courses.
Question (C): Develop a SQL query to list the details of courses that are common for more than 3 branches.
Question (D): Develop a SQL query to list students who got 'S' in more than 2 courses during single enrollment.
Question (E): Create a view that will keep track of the roll number, name and number of courses, a student has completed successfully.

Assignment – 2:

Consider the following relations for an Order Processing Database application in a Company.
Customer (Customerno varchar2 (5), Cname varchar2 (50)); Implement check constraints to check Customerno starts with 'C'.
Cust_Order (Orderno varchar2(5), Odate Date, Customerno references Customer, Ord_amt number(8)); Implement check constraints to check Orderno starts with 'O'.
Ord_amt is derived attribute (default value is 0);
Item (Itemno varchar2 (5), Item_name varchar2 (30), unit_price number (5)); Implement check constraint to check Itemno starts with 'I'.
Order_item (Orderno references Cust_order, Itemno references item, qty number (3));
Primary Key is underlined. Questions
These are questions for assignment 2. The solution is available after the last question.
Question (A): Develop DDL to implement above schema enforcing primary key, check constraints and foreign key constraints.
Question (B): Populate Database with rich data set.
Question (C): Develop SQL query to list the details of customers who have placed more than 3 orders.
Question (D): Develop a SQL query to list details of items whose price is less than the average price of all items in each order.
Question (E): Develop a SQL query to list the orderno and number of items in each order.
Question (F): Develop a SQL query to list the details of items that are present in 25% of the orders.
Question (G): Develop an update statement to update the value of Ord_amt.
Question (H): Create a view that keeps track of detail of each customer and number of Order placed.

Assignment – 3:

Q3: Consider the following relational schema
Staff (Staffno number (5), Name varchar2 (30), Dob Date, Gender Char (2), Doj Date, Designation varchar2 (30), Basic_pay number (6), Deptno varchar2 (5));
Gender must take value 'M' or 'F'.
Dept (Deptno varchar2 (5), Name varchar2 (30));

Skill (Skill_code varchar2 (5), Description varchar2 (30), Charge_Outrage number (3));
Staff_skill (Staffno number (5), Skill_code varchar2 (5));
Project (Projectno varchar2 (5), Pname varchar2 (5), Start_Date Date, End_Date Date, Project_Manager_Staffno number (5)); Project Number must start with 'P'.
Works (Staffno number (5), Projectno varchar2 (5), Date_Worked_On Date, Intime Timestamp, Outtime Timestamp);
Primary Key is underlined. Questions
These are questions for assignment 3. The solution is available after the last question.
Question (A): Develop DDL to implement the above schema specifying appropriate data types for each attributes and enforcing primary key, check constraints and foreign key constraints.
Question (B): Populate the database with rich data set.
Question (C): Develop a SQL query to list the departmentno and number of staff in each department,
Question (D): Develop a SQL query to list the details of staff who earn the AVG basic pay of all staff.
Question (E): Develop a SQL query to list the details of staff who have more than 3 skills.
Question (F): Develop a SQL query to list the details of staff who have skills with a charge outrate greater than 60 per hour.
Question (G): Create a view that will keep track of the department number, department name, the number of employees in the department and total basic pay expenditure for the department.
Question (H): Develop a SQL query to list the details of Depts which has more than 5 staff working in it.
Question (I): Develop a SQL query to list the details of staff who have more than 3 skills.

Assignment – 4:

Consider the following relational schema for a banking database application. Customer (Cid, Cname);
Branch (Bcode, Bname);
Account (Ano, Atype, Balance, Cid, Bcode);
An account can be a saving account or a current account. Check Atype in 'S' or 'C'. A customer can have both types of accounts. Transaction (Tid, Ano, Tttype, Tdate, Tamount);
Ttype can be 'D' or 'W'.
D – Deposit, W – Withdrawal Primary Key is underlined. Questions
These are questions for assignment 4. The solution is available after the last question.
Question (A): Develop DDL to implement the above schema specifying an appropriate data type for each attribute enforcing primary key, check constraints and foreign key constraints.
Question (B): Populate the database with a rich data set.
Question (C): Develop a SQL query to list the details of customers who have a saving account and a current account.
Question (D): Develop a SQL query to list the details of branches and the number of accounts in each branch.
Question (E): Develop a SQL query to list the details of branches where the number of accounts is less than the average number of accounts in all branches.
Question (F): Develop a SQL query to list the details of customers who have performed three transaction on a day.
Question (G): Create a view that will keep track of branch details and the number of accounts in each branch.

Assignment – 5 :

Let us consider the following database schema. As you can see in below figure, there are four tables (Existing Database)
- Projects, Employees, ProjectEmployees, and JobOrders. Recently, the Customers table has also been added to the database to store the customers' information. As you can see in the

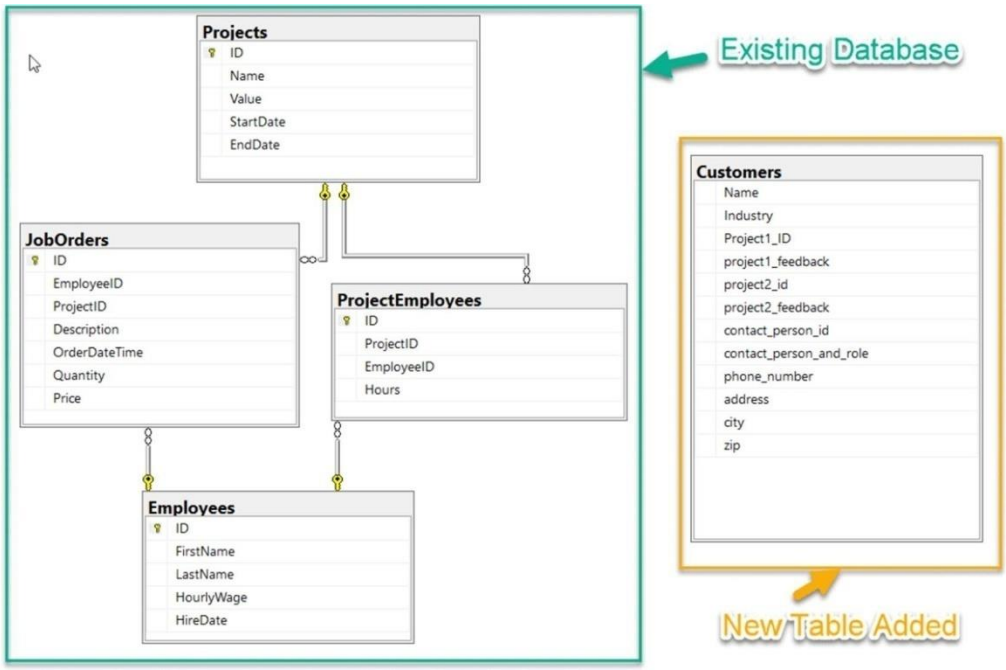


diagram below, the Customers table has not been designed in a proper way to support the normal forms, let's go ahead and fix it.

The Customers table in the diagram violates all the three rules of the first normal form.
We do not see any Primary Key in the table.
The data is not found in its most reduced form. For example, the column ContactPersonAndRole can be divided further into two individual columns - ContactPerson and ContactPersonRole.
Also, we can see there are two repeating groups of columns in this table - (Project1_ID, Project1_FeedBack) and (Project2_ID, Project2_Feedback). We need to get these removed from this table.

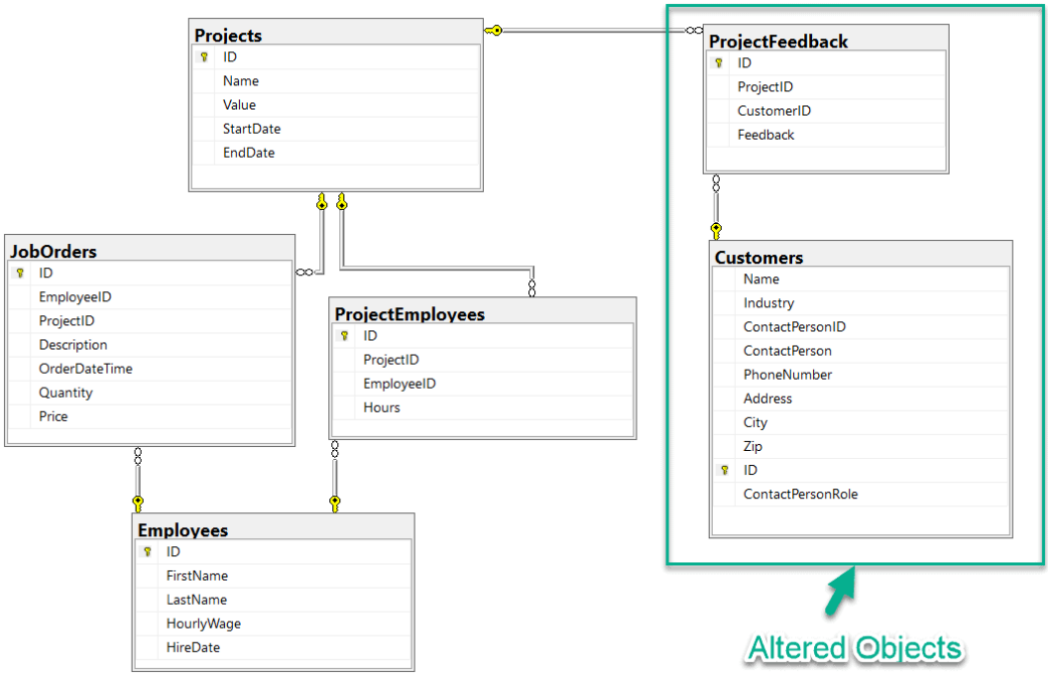
The diagram below shows dummy data stored in the Customers table.

Name	Industry	Project1_ID	Project1_Feedback	Project2_ID	Project2_Feedback	ContactPersonID	ContactPersonAndRole	PhoneNumber	Address	City	Zip
Zydus Cadilla	Pharma	2455	Amazing Work!			133	Dave, HoD	555-55-5555	1, Landing Street	York	23456
HDFC	Finance	9855	Nice job!	4924	Fantastic!	146	Mark, Ops Lead	222-22-2222	2, Times Square	London	86421
ICICI	Finance	3965	Well done.			122	Peter, Analyst	444-44-4444	3, Garden Street	Brussels	53864

a. Add a primary key to this table. For this, add a new column *ID* with datatype as *INT* and also assign it as an *Identity* column.
b. split the column ContactPersonAndRole into two individual columns. This can be done in two steps as follows:
   i. Rename the original column from ContactPersonAndRole to ContactPerson.
   ii. Add a new column for ContactPersonRole.
c. Finally, in order to satisfy the third rule of the First Normal Form, move the columns *Project1_ID*, *Project1_Feedback*, *Project2_ID*, and *Project2_Feedback* into a new table.

This can be done by creating a new table *ProjectFeedbacks* and link it back with the *Customers* and the *Projects* table which remove the above-mentioned columns from the *Customers* table and create a new table *ProjectFeedbacks* with Foreign Key references to the *Customers* and *Projects* table.

The database schema after applying all the rules of the first normal form should be as below.



If you see the database schema diagram above, you can see that the *ContactPerson*, *ContactPersonRole* and the *PhoneNumber* do not directly relate to the *ID* of the *Customers* table. That is because the primary key refers to a customer and not to any person or role or the phone number of the contact person.

1. Remove all these columns from the *Customers* table which do not relate to the primary key of the table directly.
2. Once, the columns are removed from the *Customers* table, now create a new table that'll store the data for the contact persons. Let us create a new table *ContactPersons* and relate it to the *Customers* table with a foreign key relation

Assignment – 6:

Implement B+ tree using any Programming Language.

List of Minor Projects Based on SQL

1. Blood Donation Management System
2. Cooking Recipe Website
3. Library Database Management System
4. Online Retail Database Software
5. Inventory Management System
6. Voice Commands Transport Enquiry System
7. Carbon-Emission Calculator
8. Railway Control System Database
9. Student Database Management

10. Hospital Management System
11. Payroll Management System
12. Grocery Store Sales

Course Name: Object-Oriented Programming Credit: 4

Course Code: MCA202 Lecture Hours: 40

Name of the Course: Object-Oriented Programming	
Course Code: MCA202	**Semester: 2nd**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 4	
Aim:	
1.	To gain the knowledge of basic object-oriented programming techniques.
2.	Learning the underlying concepts of Java Programming.
3.	Get industry ready with the coding skills.
Objective:	
1.	To understand the basic concepts and fundamentals of platform independent object-oriented language.
2.	To demonstrate skills in writing programs using exception handling techniques and multithreading.
3.	To understand streams and efficient user interface design techniques.
4.	To understand the basic concepts and fundamentals of platform independent object-oriented language.
Pre-Requisite:	
1.	Basics of programming language.
2.	Logic building skills.
Course Outcome:	
1.	Understand and apply fundamental concepts of Object-Oriented Programming such as classes, objects, inheritance, polymorphism, abstraction, and encapsulation to develop reusable and maintainable software.
2.	Design and implement Java programs using control structures, arrays, strings, exception handling, and file I/O operations to solve real-world problems.
3.	Develop modular applications using Java packages, interfaces, multithreading, and collections to improve performance and reusability.
4.	Build basic GUI-based applications and perform database connectivity using AWT/Applets and JDBC to demonstrate end-to-end software development skills.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	1	0	2	2	1	0	0	1	1	1	2	2	1	0
CO2	3	2	3	2	2	1	0	0	2	1	1	2	3	1	0
CO3	3	3	3	3	3	2	0	2	3	2	2	3	3	3	1
CO4	3	3	3	3	3	2	0	2	3	2	2	3	3	3	2

Module number	Topic	Sub-topics
1	**OOPs Concept**	Object, Class, Data abstraction, Data encapsulation, Inheritance, Polymorphism, Dynamic binding
	An overview of Java	History of Java, Java features, JVM, Comparison between Java and C++, Idea of Java Development Kit (JDK), learn to run Java program through the command line.
	Data Concept	Data Types, Variables, Arrays and constants Tokens in Java (Identifiers, Literals, Keywords, Operator)
	Control Statements	Simple if statement, if-else statement, Nesting of if-else statement, switch statement
	Iteration Statement	for loop, while loop, do-while loop
	Classes and Objects	Creating main() in a separate class, Methods with parameters, Methods with a return type, Method overloading, Passing Objects as Parameters, Passing Values to methods and Constructor, Abstract classes
2	**String and String Buffer**	Use of different functions
	Inheritance	Basic concepts, types of inheritance, use of super keyword, overriding methods.
	Packages, Interfaces	User-defined package, standard packages, import package, Class path, how to create interface, use and extend interface
	Multithreaded Programming	Overview, Thread Life cycle, Advantages of

		multithreading over multitasking, Thread Creation, Synchronized threads, Synchronized Methods
3	**Exception Handling**	Overview of exception, Compile time errors Run time errors, try-catch, use of multiple catch Blocks, finally block, throwing an exception, using the throw and throws statement.
	Collections	Collections, Iteration, Set and SortedSet, List, Map and SortedMap, Legacy Collection Types
4	**Stream**	Byte Streams, Input Stream, Output Stream Character Streams (Reader, Writer), How Files and Streams Work, Working with Reader classes (InputStreamReader, BufferedReader)
	Applets	Applet vs. Application, Applet class, Advantages of Applet, Applet
	Abstract Window Toolkit	GUI Components, Interface and Classes of AWT Package, Swings, Labels, Buttons, Check Boxes, Radio button, Text Area, Text Field, Scrollbar, Panels, Layout managers, Simple event-driven programming with Text Field and Button

List of Books			
Text Books			
Name of Author	**Title of the book**	**Edition/ ISSN/ ISBN**	**Name of the Publisher**
Herbert Schildt	Java: The Complete Reference	Eleventh Edition	McGraw-Hill
Ken Arnold, David Holmes, James Gosling, Prakash Goteti	The Java Programming Language	Third Edition	Pearson Education
E. Balagurusamy	Programming with Java	Fourth Edition	McGraw-Hill
Reference Books:			
Core Java An Integrated Approach (Black Book)	Core Java An Integrated Approach (Black Book)	First Edition	Dreamtech Press
Kogent Learning Solutions	Web Technologies, Black Book	First Edition	Dreamtech Press
Paul Deitel, Harvey Deitel	Java How to Program: Early Objects	Eleventh Edition	Pearson Education
Kathy Sierra, Bert Bates, Trisha Gee	Head First Java: A Brain-Friendly Guide	Third Edition	Shroff/O'Reilly

Name of the Course: Object Oriented Programming with Java Laboratory	
Course Code: MCA292	**Semester: 2nd**
Duration: 12 Weeks.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Practical: 3	Practical Sessional Internal continuous evaluation: 100
Credit: 3	Practical Sessional external examination: 100

Aim:	
1.	To gain the knowledge of basic object-oriented programming techniques.
2.	Learning the underlying concepts of Java Programming.
3.	Get industry ready with the coding skills.
Objective:	
1.	To understand the basic concepts and fundamentals of platform independent object-oriented language.
2.	To demonstrate skills in writing programs using exception handling techniques and multithreading.
3.	To understand streams and efficient user interface design techniques.
4.	To understand the basic concepts and fundamentals of platform independent object-oriented language.
Pre-Requisite:	
1.	Basics of programming language.
2.	Logic building skills.
Course Outcome:	
1.	Students should have an idea of how to work with different datatypes, operators, conditional statements and iterative statements in Java.
2.	Students should have an idea of how to work with strings, arrays, and different collection interfaces.
3.	Students should be able to use and design programs using their advanced data structures, I-O Streams, AWT, and GUI Programming using Applets and Swings.
4.	Students will learn to work with object-oriented programming constructs in Java and make small projects based on them.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	1	1	1	0	0	0	1	0	2	2	3	1
CO2	3	3	2	2	3	1	0	0	0	1	0	2	3	3	1
CO3	3	3	3	1	3	1	0	0	0	1	0	2	3	3	1
CO4	3	3	3	2	3	1	0	0	0	1	0	2	3	3	1

Module number	Topic	Sub-topics
1	**OOPs Concept**	**Annexure – I (Programming based Lab – Assignments)**
	An overview of Java	Assignment – 1: Basic Programming and Command Line Arguments
	Data Concept	
	Control Statements	Assignment – 2: Constructors & Inheritance
	Iteration Statement	for loop, while loop, do-while loop
	Classes and Objects	Assignment – 3: Flow Control
2	**String and String Buffer**	Assignment – 4: Inheritance and Dynamic Polymorphism
	Inheritance	
	Packages, Interfaces	Assignment – 5: Abstract class & Interface in Java.
	Multithreaded Programming	Assignment – 6: Threads, Multithreading & Thread Synchronization
3	**Exception Handling**	Assignment – 7: Exception Handling& Collections
	Collections	
4	**Stream**	Assignment – 8: Keyboard input and string handling in Java
	Applets	
	Abstract Window Toolkit	

Annexure – I (Programming Based Lab – Assignments)

Assignment – 1: Basic Programming and Command Line Arguments

1. Write a Java Program to print your Name entered through the command line as an argument.
2. Write a Java program to convert Temperature from Fahrenheit to Celsius and vice versa.
3. Write a Java program to add two numbers.
4. Write a Java Program to find the area and Perimeter of a rectangle.
5. Write a program in Java to find the maximum of three numbers.
6. Write a Java Program to check whether a given year is a leap year.
7. Create four different classes with three of them containing the function main.Save the file with a different name than that of the class name and run each of the classes with the main function.
8. Write a Java program to reverse a number entered as a command line argument.
9. Write a Java program to count the number of digits entered through the command line argument.
10. Write a Java program to find all the multiples of 3 within a given range where the starting and ending values are entered through a command line argument.

Assignment – 2: Constructors & Inheritance

1. Write a class, Grader, which has an instance variable, score, an appropriate constructor and appropriate methods. A method, lettergrade (), that returns the letter grade as O/E/A/B/C/F. Now write a demo class to test the Grader class by reading a score from the user, using it to create a Grader object after validating that the value is not negative and is not greater than 100. Finally, call the letterGrade() method to get and print the grade.
2. Write a class, Commission, which has an instance variable, sales; an appropriate constructor; and a method, commission() that returns the commission. Now write a demo class to test the Commissionclass by reading a sale from the user, using it to create aCommission object after validating that the value is not negative.Finally, call the commission() method to get and print the commission. If the sales are negative, your demo should print themessage "Invalid Input".
3. For a Mobile Shop project, create a "Telephone" class with details like mobile_id, model_name and available_quantity in "Phone" package. Inherit from this class and create a class for"smart_phone" with necessary information like enabled_5G, foldable anddual_screen in package "Smart". The customer executive tries to display all smart_phone details (mobile_id, model_name, available_quantity, enabled_5G, foldable and dual_screen) and updates the quantity information, whenever the customer purchases the smart_phone. Write the necessary java programs to implement this scenario and test with user inputs.
4. An educational institution maintains a database of its employees. The database is divided into a number of classes whose hierarchical relationships are shown below. Write all the classes and define the methods to create the database and retrieve individual information as and when needed. Write a driver program to test the classes. Stab (code, name), Teacher (subject, publication) is a Staff, Officer (grade) is a Staff, Typist (speed) is a Staff RegularTypist (remuneration) is a Typist, and CasualTypist (daily wages) is a Typist.

Assignment – 3: Flow Control

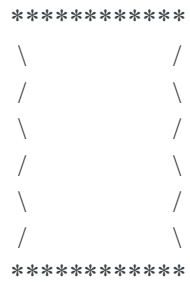
1. The process of finding the largest value (i.e., the maximum of a group of values) is used frequently in computer applications. For example, a program that determines the winner of a sales contest would input the number of units sold by each salesperson. The salesperson who sells the most units wins the contest. Build a Java application that inputs a series of 10 integers and determines and prints the largest integer. Your program should use at least the following three variables:
   a. counter: A counter to count to 10 (i.e. to keep track of how many numbers have been input and to determine when all 10 numbers have been processed).
   b. number: The inter most recently input by the user.
   c. largest: The largest number found so far.

   Note: Every time the sales figure of one employee is entered, the application should ask the user if they want to enter any more sales figures of a salesperson!

2. Write an application that prompts the user to enter the size of the side of a square, and then displays a hollow square of that size made of asterisks. Your program should work for squares of all side lengths between 1 and 20.
3. Write a program to compute the following formula.

   e= 1/0!+ 1/1! +½! + ⅓!+.....+ 1/n!

4. Using an enhanced for (for-each) loop, copy the content of one 3-dimensional array to another 3-dimensional array and display its contents.
5. Create the following vase pattern using a loop:



Assignment – 4: Inheritance and Dynamic Polymorphism

1. Create a general class ThreeDObject and derive the classes Box, Cube, Cylinder and Cone from it. The class ThreeDObject has methods wholeSurfaceArea( ) and volume( ). Override these two methods in each of the derived classes to calculate the volume and whole surface area of each type of three-dimensional object. The dimensions of the objects are to be taken from the users and passed through the respective constructors of each derived class. Write a main method to test these classes.
2. Create a base class Building that stores the number of floors of a building, the number of rooms and its total footage. Create a derived class House that inherits the Building and also stores the number of bedrooms and bathrooms. Demonstrate the working of the classes.
3. In the earlier program, create a second derived class Office that inherits the Building and stores the number of telephones and tables. Now demonstrate the working of all three classes.
4. Create a base class Distance which stores the distance between two locations in miles and a method travelTime(). The method prints the time taken to cover the distance when the speed is 60 miles per hour. Now in a derived class DistanceMKS, override travelTime() so that it prints the time assuming the distance is in kilometres and the speed is 100 km per second. Demonstrate the working of the classes.
5. Create a base class called "vehicle" that stores the number of wheels and speed. Create the following derived classes –"car" that inherits "vehicle" and also stores the number of passengers.
   "truck" that inherits "vehicle" and also stores the load limit.
   Write a main function to create objects of these two derived classes and display all the information about "car" and "truck". Also, compare the speed of these two vehicles - car and truck and display which one is faster.

Assignment – 5: Abstract class & Interface in Java.

1. Design an abstract class having two methods. Create Rectangle and Triangle classes by inheriting the shape class and override the above methods to suitably implement for Rectangle and Triangle class.
2. Write a program to create a class named Vehicle having protected instance variables regnNumber, speed, colour, ownerName and a method showData( ) to show "This is a vehicle class". Inherit the Vehicle class into subclasses named Bus and Car having individual private instance variables route Number in Bus and manufacturer Name in Car and both of them having showData( )method showing all details of Bus and Car respectively with the content of the super class's showData ( ) method.
3. Create an interface Department containing attributes deptName and deptHead. It also has abstract methods for printing the attributes. Create a class hostel containing hostelName, hostelLocation and numberofRooms. The class contains methods for getting and printing the

attributes. Then write a Student class extending the Hostel class and implementing the Department interface. This class contains attributes studentName, regdNo, electiveSubject andavgMarks. Write suitable getData and printData methods for this class. Also, implement the abstract methods of the department interface. Write a driver class to test the Student class. The program will be menu driven containing the options:

i) Admit new student
ii) Migrate a student
iii) Display details of a student

For the third option, a search is to be made on the basis of the entered registration number.

4. Create an abstract class Accounts with the following details:

Data Members:

(a) Balance
(b) accountNumber
(c) accountHoldersName
(d) address

Methods:

(a) withdrawl()- abstract
(b) deposit()- abstract
(c) display() to show the balance of the account number

Create a subclass of this class SavingsAccount and add the following details:

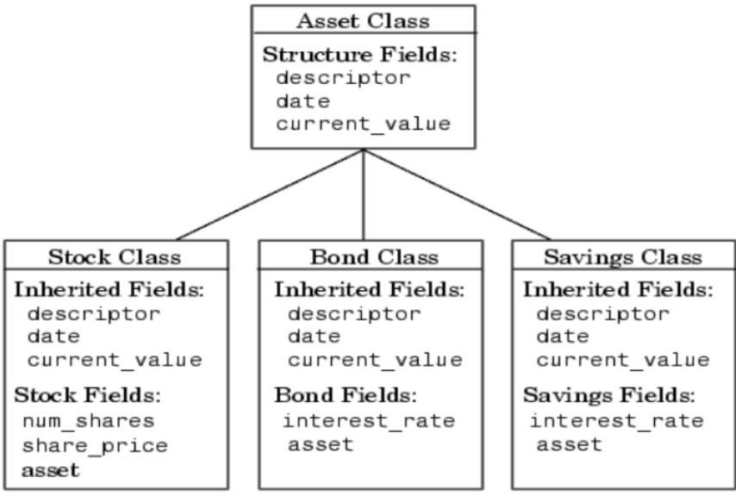
Data Members:

(a) rateOfInterest

Methods:

(a) calculateAount()

5. Implement the below Diagram.



Here, Asset class is an abstract class containing an abstract methoddisplayDetails() method. Stock, bond and Savings class inherit the Asset class and displayDetails() method is defined in every class.

Assignment – 6: Threads, Multithreading & Thread Synchronization

1. Write a Java program in which a total of 4 threads should run. Set different priorities for the thread.
2. Write a Java Program to Create a Thread that Implements the Runnable Interface.
3. Write a Java Program to Check the Priority Level of a Thread.

4. Write a Java Program Defining Thread by Extending the Thread class.
5. Write a Java Program to Get the Name of a Running Thread.
6. Write a Java Program to Stop a Thread.
7. Write a Java Program to Check Whether Define a Thread Class Without Defining run() Method in the Class.
8. Write a Java Program to Show that Method Will be Verified Whether it is Synchronized or Not.
9. Create 4 threads with priority 1,3,5,7 respectively. Update a counter in each of the threads for 10 ms. Print the final value of the count for each thread.
10. Write a Java Program to Use Method Level Synchronization.

Assignment – 7: Exception Handling& Collections

1. Write a Java program using try and catch to generate Array Index Out of Bound Exception and Arithmetic Exception.
2. Write a class that keeps a running total of all characters passed to it(one at a time) and throws an exception if it is passed a non-alphabetic character.
3. Write a program that takes a value at the command line for which the factorial is to be computed. The program must convert the string toits integer equivalent. Three possible user input errors can prevent the program from executing normally.
   - The first error occurs when the user provides no argument while executing the program, and an arrayIndexOutOfBoundsException is raised. You must write a catch block for this.
   - The second error is NumberFormatException which is raised in case the user provides a non-integer (float double) value at the command line.
   - The third error is IllegalArgumentException. This needs to be thrown manually if the value at the command line is 0.
4. Create a user-defined exception named CheckArgument to check the number of arguments passed through the command line. If the number of arguments is less than 5, throw the CheckArgumentexception, and print the addition of all the five numbers.
5. Write a Java program to create a custom Exception that would handle at least 2 kinds of Arithmetic Exceptions while calculating a given equation (e.g. X+Y*(P/Q) Z-I).
6. Given an element write a program to check if an element(value) exists in ArrayList.
7. Write a program to convert LinkedList to ArrayList.
8. Write a program to iterate TreeMap in java.

Assignment 8: Keyboard input and string handling in Java

1. Write a Java program for calculating Factorial. Number should be taken through user input (Using Scanner, BufferedReader both).
2. Write a Java program to reverse a string. (String will be taken as user input through the console).
3. Write a Java Program to Find the Length of the String.
4. Write a Java Program to Remove the White Spaces from a String.
5. Write a Java Program to Use the Equals Method In a String Class.
6. Write a Java Program to Count and Replace the First Occurrence of a String.
7. Write a Java Program to Validate an Email Address Format.
8. Write a Java Program to Access the Index of the Character or String.
9. Write a Java Program to Find First and Last Occurrence of a given character in a String.
10. Write a Java Program to Store String Literals Using String Buffer.

Course Name: Data Communication & Computer Networks Credit: 3

Course Code: MCA203 Lecture Hours: 40

Name of the Course: Data Communication & Computer Networks Credit	
Course Code: MCA203	**Semester: 2nd**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 3	
Aim:	
1.	To gain Knowledge of uses and services of Computer Network
2.	To enhance Ability to identify types and topologies of network.
3.	To gain Understanding of analog and digital transmission of data.
Objective:	
1.	To deliver comprehensive view of Computer Network.
2.	To enable the students to understand the Network Architecture, Network type and topologies.
3.	To understand the design issues and working of each layer of OSI model.
4.	To familiarize with the benefits and issues regarding Network Security.
Pre-Requisite:	
1.	Knowledge of basic data communication & network security.
Course Outcome:	
1.	Identify the different components in a Communication System and their respective roles.
2.	Describe the technical issues related to the Networks
3.	Defining the standard model and protocols of networking
4.	Understand the basics of data communication, networking, internet and their importance.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	1	0	0	2	1	0	0	0	1	0	1	2	1	0
CO2	2	3	2	2	2	1	0	0	0	1	0	1	2	2	1
CO3	3	3	2	1	3	1	0	0	0	1	0	2	3	2	1
CO4	3	2	1	0	2	1	0	0	0	1	0	2	2	2	0

Module number	Topic	Sub-topics
1	**Introduction to Networks & Network Model**	Introduction to communication systems, components, Transmission Impairments, and Performance criteria of a communication system. Goals of computer Network, network classification, Components and Topology, categories of network[LAN, MAN, WAN]; Internet: brief history, internet today; Protocols and standards; OSI and TCP/IP model
2	**Physical Layer**	**Data, signal and Transmission:** Analog and Digital, Transmission modes, Overview of data[analogue & digital], signal[analogue & digital], transmission [analogue & digital] & transmission media [guided & unguided]; Circuit switching: time division & space division switch, TDM bus; Telephone Network.
3	**Data Link Layer**	**Data link layer:** Types of errors, framing [character and bit stuffing], error detection & correction methods; Flow control; Protocols: Stop & wait ARQ **Medium access sublayer:** Point-to-point protocol, FDDI, token bus, token ring; Reservation, polling, concentration; Multiple access protocols: ALOHA, CSMA, FDMA, TDMA, CDMA; Ethernet
4	**Network Layer**	Concepts of Internetworking & devices: Repeaters, Hubs, Bridges, Switches, Router, Gateway; **Addressing**: Internet address, classful address, **Routing:** techniques, static vs. dynamic routing Protocols: IP, IPV6.
5	**Transport Layer**	Process to process delivery; Details of UDP; Details of TCP; Congestion control algorithm: Leaky bucket algorithm, Tokenbucket algorithm, Quality of services [QoS]
6	**Application Layer**	Details of Application Layer protocols/services such as HTTP, FTP, Telnet, SMTP & WWW and other
7	**Cryptography & Satellite Communication**	Introduction to data security & cryptography (private key, public key, ISO standards), Digital Signature, Firewalls [technology & applications] Brief concepts of Satellite Communication such as LEO, GEO.

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ ISSN/ ISBN**	**Name of the Publisher**
Behrouz A Forouzan	Data Communication & Networking	4th Ed	TMH
Andrew S. Tannenbaum	Computer Networks	6th Ed	PHI

Reference Books:			
William Stallings	Data & Computer Communications	10th Ed	PHI
Douglas E. Comer	Computer Networks and Internets with Internet Applications	4th Ed	Pearson
Jean Warland	Communication Networks: A First Course	2nd Ed	TMH
Ed Title	Schaum's Outline of Computer Networking	2nd Ed	TMH

Course Name: Graphics and Multimedia Credit: 3

Course Code: MCA204 Lecture Hours: 40

Name of the Course: Graphics and Multimedia	
Course Code: MCA204	**Semester: 2nd**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 3	
Aim:	
1.	To provide a theoretical foundation in computer graphics and multimedia technologies.
2.	To equip students with mathematical and algorithmic principles for graphics and multimedia processing.
3.	To introduce graphics standards, rendering models, and multimedia frameworks used in real-world applications.
Objective:	
1.	To explain core graphics concepts, including transformations, clipping, and rendering.
2.	To analyze graphics algorithms like rasterization, polygon filling, and shading.
3.	To explore multimedia components, standards (JPEG, MPEG, MIDI), and design methodologies.
4.	To understand the theoretical aspects of graphics and multimedia systems without hands-on implementation.
Pre-Requisite:	
1.	Basic knowledge of programming logic and mathematical concepts (linear algebra, geometry, trigonometry).
Course Outcome:	
1.	Explain the fundamental concepts of computer graphics and multimedia.
2.	Analyze graphics algorithms for drawing, clipping, and shading.
3.	Describe multimedia technologies, standards, and methodologies.
4.	Evaluate the theoretical models of graphics rendering and multimedia processing.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	2	2	1	1	0	1	0	2	3	1	2
CO2	3	3	2	2	2	1	1	0	0	1	0	2	3	2	2
CO3	3	2	2	1	2	3	2	1	0	2	1	2	3	3	2
CO4	3	3	2	2	2	2	1	1	0	1	1	3	3	2	3

Module number	Topic	Sub-topics
1	**Introduction to Computer Graphics**	Application of Computer Graphics, Graphics Devices, Cathode Ray Tube, Raster Scanning, Raster Refresh graphics displays.
2	**Graphics Operations and Drawing Algorithms**	Graphics Operations –2D & 3D Graphics, Bezier, B-Spline, Hermite, Bresenham Line & Circle Drawing Algorithms, Polygon filling, Edge Filling Algorithms.
3	**Clipping and Visible Surface Detection**	**Clipping Techniques:** Cohen-Sutherland subdivision line clipping algorithm, Mid-Point subdivision algorithm, 2-dimensional clipping algorithm (Convex Boundaries & Partially visible lines), Cyrus-Beck algorithm for Partially & Totally Visible Lines) **Visible Surface Detection:** Floating Horizon Algo., Upper & Lower Horizon, Roberts algo, Warnock algo, Scan-line Z-buffer algo.
4	**Rendering and Shading Techniques**	Rendering- introduction (illumination models), shading- Gouraud Shading, Phong Shading. Shadowing- Shadow Algorithms
5	**Introduction to GKS and Multimedia**	Introduction to GKS (Graphical Kernel System). Multimedia, concepts, design, hardware, standards – MPEG, JPEG, MIDI, multimedia design methodology, development and testing

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ ISSN/ ISBN**	**Name of the Publisher**
Hearn & Baker	Computer Graphics	2nd Ed	PHI
Reference Books:			
Rogers	Procedural & Mathematical Elements in Computer Graphics		TMH
Plastock	Computer Graphics	Schaum Outline Series	TMH

Course Name: Statistics and Numerical Techniques Credit: 3

Course Code: MCA205 Lecture Hours: 40

Name of the Course: Statistics and Numerical Techniques	
Course Code: MCA205	**Semester: 2nd**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 3	
Aim:	
1.	To develop a strong theoretical foundation in statistical methods and numerical techniques for solving computational problems.
2.	To equip students with mathematical tools for data analysis, approximation, and problem-solving in computing.
3.	To enhance analytical thinking for decision-making and error estimation in computational models.
Objective:	
1.	To introduce fundamental statistical concepts like probability, distributions, and hypothesis testing.
2.	To explore numerical techniques for root finding, interpolation, differentiation, and integration.
3.	To analyze error propagation and the stability of numerical algorithms.
4.	To apply statistical and numerical methods in computing and real-world problem-solving.
Pre-Requisite:	
1.	Basic knowledge of algebra, calculus, and programming logic.
Course Outcome:	
1.	Explain fundamental statistical and numerical methods used in computing.
2.	Apply probability and statistical techniques to analyze data and draw conclusions.
3.	Utilize numerical techniques to solve mathematical and computational problems.
4.	Evaluate the accuracy, efficiency, and limitations of statistical and numerical methods.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	2	1	1	0	0	1	0	2	2	3	2
CO2	3	3	2	2	2	2	1	0	0	1	0	2	2	3	2
CO3	3	3	2	2	2	1	1	0	0	1	1	3	3	2	3
CO4	3	3	2	2	2	2	1	1	0	1	1	3	3	3	3

Module number	Topic	Sub-topics
1	**Descriptive Statistics and Curve Fitting**	Statistics - measure of central tendency, dispersion (Moments, Skewness & Kurtosis). Least square curve fitting - linear & non-linear.
2	**Probability and Parameter Estimation**	Probability, introduction to mass function, density function, distribution function (Binomial, Poisson, Normal), estimation of parameters (unbiasedness-concept of noise/error, consistency).
3	**Interpolation and Inverse Interpolation**	Interpolation-Newton's Forward, Backward, Sterling & Bessel's Interpolation formulae, Lagrange's Interpolation. Inverse Interpolation.
4	**Numerical Integration**	Integration - Trapezoidal, Simpson's 1/3rd, Weddle's Rule, Romberg Integration, Gauss- Legendre two & three points formula, Newton Cotes Formula.
5	**Root Finding Methods**	Solution of any equation - Method of Iteration, Method of Bisection, Newton-Raphson Method, Regula-Falsi method and Secant Method.
6	**Solving Linear and Differential Equations**	Solution of system of linear equations - Gauss Elimination Method, Gauss-Jacobi, Gauss-Seidel, LU factorization and Tri-diagonalization. Solution of differential equations - Picard's method, Euler-modified method, Taylor's Series method, Runge-Kutta method, Milne's Predictor-Corrector method.

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ ISSN/ ISBN**	**Name of the Publisher**
Hearn & Baker	Computer Graphics	2nd Ed	PHI
Reference Books:			
Rogers	Procedural & Mathematical Elements in Computer Graphics		TMH
Plastock	Computer Graphics	Schaum Outline Series	TMH

Course Name: General Studies & Current Affairs - I Credit: 0

Course Code: IVC(MC)201 Lecture Hours: 20

Name of the Course: Essential Studies for Professionals–II	
Course Code: IVC(MC)201	**Semester: 1st**
Duration: 48 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 2	End Semester Exam: 100
Tutorial: 0	Continuous Assessment: 100
Credit: 0	

Aim:	
1	To enhance quantitative aptitude and logical reasoning for effective problem-solving in professional environments.
2	To develop oral, listening, and reading communication skills for workplace interactions.
3	To equip students with practical communication skills, including professional correspondence and presentations.
Objective:	
1	To strengthen quantitative and logical reasoning for analytical decision-making.
2	To improve verbal and non-verbal communication skills, including articulation and clarity.
3	To enhance listening and reading comprehension for better information processing.
4	To develop practical communication techniques, such as presentations, business communication, and public speaking.
Pre-Requisite:	
1.	Basic understanding of English language and fundamental mathematical concepts.

Module number	**Topic**	**Sub-topics**
1	**GK, Current Affairs and Economics**	GK and Current Affairs – Based on Monthly Magazines provided and recent news of national and international importance. Newspaper Reading: The Economic Times. 1. **Basic economics** -Types of Economy, Feature of Indian Economy (**BECC-101, Block-1, Unit-1,Unit-2, Unit-3)** 2. HDI(BECC111, Block-2 http://egyankosh.ac.in//handle/1 23456789/81256

		3. **Sectors of the economy and their analysis: Primary (Agriculture, Mining, etc), Secondary (Industry, various policies), Tertiary (services, etc.) (Textbook: Indian Economy: Misra & Puri, Chapter- 30,32)** 4. Liberalisation, Privatisation and Globalisation (LPG)(IGNOU, BECC-114, Block- 6) http://egyankosh.ac.in//handle/1 23456789/90547 5. **RBI & Its Function**- Board of Governance, Operation. Credit control policies- CRR, SLR, Bank rate, Repo rate, Reverse Repo rate, Prime lending rate, MSF, LAF, FERA, FEMA. **(BECC-113, Unit-1)** http://egyankosh.ac.in//handle/123 456789/89589 6. **Budget (Union, Railway),** Concept of revenue, expenditure & different types of deficit. (BECC-109, Block- 3, Unit-9) http://egyankosh.ac.in//handle/1 23456789/76561

References

1. Indian Economy-Ramesh Singh

Name of the Course: Competitive Aptitude Training – II	
Course Code: IVC(MC)202	**Semester: 1st**
Duration: 20 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 2	End Semester Exam: 100
Tutorial: 0	Continuous Assessment: 100
Credit: 0	

Aim:	
1	To develop verbal reasoning, analytical thinking, and problem-solving skills essential for competitive exams and job recruitment.
2	To enhance English language proficiency, focusing on grammar, comprehension, and professional communication.
3	To build expertise in data interpretation, improving the ability to analyze and solve quantitative problems efficiently.

Objective:	
1	To strengthen English grammar, vocabulary, and comprehension skills for competitive aptitude tests.
2	To develop logical and analytical reasoning through structured problem-solving techniques.
3	To enhance proficiency in data interpretation, including tabular and graphical analysis.
4	To improve formal communication skills, such as official letter writing, for professional settings.

Pre-Requisite:	
1.	Basic understanding of English grammar and elementary mathematical reasoning.

Module number	Topic	Sub-topics
1	**Quantitative Aptitude**	Average- Concept on average, different missing numbers in average estimation, shortcuts & their application. Mixture & Allegation – Proportion & mixtures in percentages, populations & liquids, shortcuts & their application. Number System- concept of different numbers, remainder theorem, factors. Time & Work and Pipe & Cistern- Basic concept, Different problems & their shortcut tricks.Time, Speed & Distance Boat & Stream

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ISSN/ISBN**	**Name of the Publisher**
R.S Agarwal	Quantitative Aptitude for Competitive Examination		S.Chand

Syllabus for MCA Admission Batch 2021, 3rd Semester

Course Name: Operating Systems and Systems Software Credit: 3

Course Code: MCA301 Lecture Hours: 40

Name of the Course: Operating Systems and Systems Software	
Course Code: MCA301	**Semester: 3rd**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 3	
Aim:	
1	To understand the system architecture of an operating system
2	Ability to apply CPU scheduling algorithms to manage tasks.
3	Initiation into the process of applying memory management methods and allocation policies.
4	Knowledge of methods of prevention and recovery from a system deadlock.
Objective:	
1	To deliver a detailed knowledge of integral software in a computer system – Operating System.
2	To understand the workings of an operating system as a resource manager.
3	To familiarize the students with Process and Memory management.
4	To describe the problem of process synchronization and its solution.
Pre-Requisite:	
1	You should know about Computer Architecture and Organization.
2	Proficiency in C or another programming language.
3	Familiarity with Assembly language.
Course Outcome:	
1	Understand Operating System Concepts: Gain knowledge about operating system functions, generations, processes, and threads.
2	Analyze process management, scheduling algorithms, and concurrency mechanisms for efficient resource allocation.
3	Evaluate memory management techniques, file systems, and security mechanisms in operating systems.
4	Learn File Handling and Process Control: Understand the basics of File, Device, and Disk Storage Management

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	1	2	1	0	0	1	1	0	3	2	0	1
CO2	3	3	3	2	2	1	0	0	1	1	0	2	3	1	1
CO3	3	3	2	2	2	2	1	2	1	1	0	2	3	1	1
CO4	3	2	3	1	2	1	0	0	1	1	0	3	3	0	1

Module number	Topic	Sub-topics
1	**Introduction**	1. Introduction to Operating Systems 2. Hardware Support for Operating Systems 3. Resource Management 4. Operating System Architectures
2	**Process Management**	5. Fundamentals of Process Management 6. Process Scheduling 7. Process Communication and Synchronization 8. Deadlocks 9. Multi-threading
3	**Memory Management**	10. Basic Memory Management 11. Virtual Memory
4	**File Management**	12. File Systems 13. File System Implementation
5	**Input –Output Management**	14. Basics of I/O Management 15. Disk Management
6	**Security and Protection** **Advanced Operating System**	16. Security Issues 17. Protection Mechanisms 18. Distributed Operating Systems

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ ISSN/ ISBN**	**Name of the Publisher**
Naresh Chauhan	Principles of Operating Systems	1st Ed/ 9780198082873	Oxford Press
Reference Books:			
Abraham Silberschatz, Peter B. Galvin	Operating System Concept	9th Ed/ 9788126554270	WILEY
Andrew S. Tanenbaum	Modern Operating Systems	4th Ed/ 9789332575776	Pearson Education
William Stallings	Operating Systems	9th Ed/ 9789352866717	Pearson Education
Sumitabha Das	UNIX: Concepts and Applications (Lab Reference)	4th Ed/ 9780070635463	McGraw Hill Education

MCA393: Operating Systems Laboratory (Unix)

Name of the Course: Operating Systems Laboratory (Unix)	
Course Code: MCA393	**Semester: 3rd**
Duration: 12 Weeks.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 0	End Semester Exam: 70
Tutorial: 0	Continuous Assessment: 30
Credit: 3	

Aim:	
1	To understand the system architecture of an operating system
2	Ability to apply CPU scheduling algorithms to manage tasks.
3	Initiation into the process of applying memory management methods and allocation policies.
4	Knowledge of methods of prevention and recovery from a system deadlock.

Objective:	
1	To deliver a detailed knowledge of integral software in a computer system – Operating System.
2	To understand the workings of an operating system as a resource manager.
3	To familiarize the students with Process and Memory management.
4	To describe the problem of process synchronization and its solution.

Pre-Requisite:	
1	You should know about Computer Architecture and Organization.
2	Proficiency in C or another programming language.
3	Familiarity with Assembly language.

Course Outcome:	
CO1	Implement and demonstrate operating system concepts including process creation, thread management, and system calls using Unix/Linux commands and programming.
CO2	Apply process scheduling algorithms and synchronization mechanisms through practical implementation of inter-process communication and concurrency control in Unix environment.
CO3	Evaluate and implement memory management techniques, file system operations, and security mechanisms using Unix system programming and shell scripting.
CO4	Develop proficiency in file handling, device management, and disk storage operations through hands-on experience with Unix file systems and storage management tools.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	1	3	1	0	1	2	1	1	2	3	1	1
CO2	3	3	2	2	3	1	0	2	2	1	1	2	3	2	2
CO3	3	2	3	2	3	1	1	3	1	1	1	2	3	2	2
CO4	3	2	2	1	3	1	1	2	2	1	1	2	3	2	1

Module number	Topic	Sub-topics
1	**Introduction**	1. Basic Unix Commands
2	**Process Management**	1. C Programs for Process Scheduling 2. Implementation of Banker's Algorithm
3	**Memory Management**	1. C programs to simulate contiguous memory allocation techniques 2. C programs to simulate the paging technique
4	**File Management**	1. Unix commands on file operations 2. C program for file organization technique.
5	**Input –Output Management**	1. C programs to simulate contiguous memory allocation techniques 2. C programs to simulate the paging technique
6	**Security and Protection** **Advanced Operating System**	1. Unix commands on file operations 2. C program for file organization technique.

Course Name: Data Science and Data Analytics Credit: 3

Course Code: MCA303 Lecture Hours: 40

Name of the Course: Data Science and Data Analytics	
Course Code: MCA303	**Semester: 3rd**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 3	
Aim:	
1	To provide a comprehensive understanding of data science and analytics techniques for extracting meaningful insights.
2	To develop proficiency in data preprocessing, statistical analysis, and machine learning for decision-making.
3	To introduce big data technologies, cloud computing, and AI-based applications in modern analytics.
Objective:	
1	To explore data management, warehousing, and mining techniques for structured analysis.
2	To apply statistical methods and data visualization for insightful interpretation.
3	To understand machine learning concepts such as classification, clustering, and predictive modelling.
4	To introduce big data platforms, cloud computing, and IoT in data-driven applications.
Pre-Requisite:	
1	Basic knowledge of statistics, programming (Python/R), and databases.
Course Outcome:	
1	Explain data management, mining, and preprocessing techniques for analytics.
2	Apply statistical and machine learning methods to analyze and interpret data.
3	Utilize big data technologies, cloud computing, and IoT for data-driven solutions.
4	Implement machine learning models for real-world applications using Python/R.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	2	2	1	1	0	1	1	1	2	3	3	2
CO2	3	3	3	2	3	1	1	0	1	1	2	3	3	3	3
CO3	3	3	3	2	3	2	2	1	1	2	2	3	3	3	3
CO4	3	3	3	2	3	2	2	1	1	2	2	3	3	3	3

Module number	Topic	Sub-topics
1	**Introduction to Data Management**:	Brief idea about Data Warehousing, Architecture and Data Flows, Data pre-processing before analysis, Data preparation, OLAP & OLTP, Case study.
2	**Introduction to Data Mining:**	Brief idea about Data Mining, It's goals and techniques, Architecture and KDD Process, Knowledge representation methods.
3	**Statistics and Analytics:**	Data Visualization, Summarize and describe data sets using a measures such as Central tendency and variability, Learn probability, Central Limit Theorem and much more to draw inferences
4	**Introduction to Big Data Analytics:**	Understand the basic concepts of Big Data and Hadoop as processing platforms for Big Data, Managing Big Data - Learn and Use Hadoop Ecosystem tools for data ingestion, extraction and management. Introduction to Hive.
5	**Cloud Computing:**	Introduction to Cloud Computing, types, services, applications, Security & research scope. Internet of Things:
6	**Introduction to IOT and WSN:**	Introduction to IOT and WSN, Basic concepts of Robotics Using Arduino & Rasberry Pi Programming.
7	**Introduction to NLP & AI**	Introduction to artificial intelligence, Brief idea about Natural Language Processing.
8	**Basic concepts of Machine Learning**	To implement linear regression, Data classification, Data clustering – To learn how to create segments based on similarities using K-Means and Hierarchical clustering, Case study using Python.
9	**Applications of Machine Learning.**	Time series, Decision trees, Support Vector Machine, Neural Networks, Case Study Using MATLAB.

List of Books Text Books:

1. "Data Mining : Concepts and Techniques" by Jiawei Han and Micheline Kamber
2. "Artificial Intelligence and Soft Computing: Behavioral and Cognitive Modeling of the Human Brain" by Amit KonarLogic & Prolog Programming, Saroj Kaushik, New Age International
3. "Big Data" by Anil Maheshwari
4. "Wireless Sensor Netwroks" by Ian F. Akyildiz & Mehmet Can Vuran
5. "Wireless Ad Hoc and Sensor Networks : Theory and Applications" by Xian Yang Li
6. "Mastering Cloud Computing : Foundations and Applications Programming" by Rajkumar Buyya
7. "Fundamentals of Neural Networks: Architectures, Algorithms and Applications" by L. Fausett

Name of the Course: Data Science and Data Analytics Laboratory	
Course Code: MCA392	**Semester: 3rd**
Duration: 12 Weeks.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Practical: 3	Practical Sessional Internal continuous evaluation: 100
Credit: 3	Practical Sessional external examination: 100

Aim:	
1	To gain Knowledge of Various aspects of data science and data analytics.
2	To enhance the ability to identify qualities of a good solution of AI, Big Data, Data Mining etc.
3	To implement learned analytical techniques and data science to solve problems.
Objective:	
1	Provide you with the knowledge and expertise to become a proficient data scientist.
2	Demonstrate an understanding of statistics and machine learning concepts that are vital for data science.
3	Produce Python code to statistically analyze a dataset.
4	Critically evaluate data visualizations based on their design and use for communicating stories from data.
Pre-Requisite:	
1	Basic knowledge of statistics, programming (Python/R), and databases.
Course Outcome:	
1	Explain how data is collected, managed and stored for data science.
2	Understand the key concepts in data science, including their real-world applications and the toolkit used by data scientists.
3	Implement data collection and data mining techniques using database.
4	Understand handling of big data.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	2	2	1	1	0	1	1	1	2	3	3	2
CO2	3	3	3	2	3	1	1	0	1	1	2	3	3	3	3
CO3	3	3	3	2	3	2	2	1	1	2	2	3	3	3	3
CO4	3	3	3	2	3	2	2	1	1	2	2	3	3	3	3

Module number	Topic	Sub-topics
1	**Introduction to Data Management**:	1. Write a program for displaying reversal of a number. 2. Implement python script to read person's age from keyboard and display whether he is eligible for voting or not.

		3. Implement python script to check the given year is leap year or not. 4. Implement Python Script to generate prime numbers series up to n 5. To display elements of list in reverse order. 6. Implement python script to accept line of text and find the number of characters, number of vowels and number of blank spaces in it. 7. Write a program which makes use of function to display all such numbers which are divisible by 7 but are not a multiple of 5, between 1000 and 2000. Implement a python script for factorial of number by using recursion.
2	**Introduction to Data Mining:**	1. Write a program which accepts a sequence of comma-separated numbers from console and generate a list and a tuple which contains every number. Suppose the following input is supplied to the program: 34, 67, 55, 33, 12, 98. Then, the output should be: ['34', '67', '55', '33', '12', '98'] ('34',67', '55', '33', '12', '98'). 2. Write Python script to copy file contents from one file to another. 3. Implement a python script to check the element is in the list or not by using Linear search & Binary search. 4. Implement a python script to arrange the elements in sorted order using Bubble, Selection, Insertion and Merge sorting techniques. 5. Write a python program by using exception handling mechanism. 6. Write a python program to perform various database operations (create, insert, delete, update).
3	**Statistics and Analytics:**	1. Write a program to compute summary statistics such as mean, median, mode, standard deviation and variance of the given different types of data. 2. Write a program to demonstrate Regression analysis with residual plots on a given data set.

7	**Introduction to NLP & AI**	Python lab for text analysis 1. Choose some book-length document and download it. 2. Count its characters, lines and words. 3. Count sentences, vocabulary, and the like. 4. Show collocations, common context, concordance, and similar relationships among the words. 5. Plot a lexical dispersion or two. 6. Plot a frequency distribution of the most common words.
8	**Basic concepts of Machine Learning**	1. Write a program to demonstrate the working of the decision tree-based ID3 algorithm. 2. Write a program to implement the Naïve Bayesian classifier for a sample training data set stored as a .CSV file.
9	**Applications of Machine Learning.**	1. Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. 2. Write a program to implement k-Means clustering algorithm to cluster the set of data stored in .CSV file.

Course Name: Software Engineering & TQM Credit: 4

Course Code: MCA304 Lecture Hours: 40

Name of the Course: Software Engineering & TQM	
Course Code: MCA304	**Semester: 3rd**
Duration: 12 Weeks.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Practical: 3	Practical Sessional Internal continuous evaluation: 100
Credit: 4	Practical Sessional external examination: 100
Aim:	
1	To gain knowledge of various aspects of software engineering project management.
2	To enhance ability to identify qualities of a good solution
3	To implement learned algorithm/design techniques to solve problems
Objective:	
1	The fundamental knowledge of software engineering
2	The different basic models need to implement different project problems
3	The various design methods to develop the software system
4	The quality and other issues related to the software products and systems
Pre-Requisite:	
1	Knowledge in fundamental theories of computer science and one programming language
Course Outcome:	
1	On completion of this course students are expected to learn fundamentals and different models of software engineering.
2	On completion of this course students are expected to learn different aspects of requirement analysis in software project management.
3	On completion of this course students are expected to learn various types of software design and concepts of coding.
4	On completion of this course students are expected to learn different types of testing and quality issues.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	1	2	0	0	1	0	1	1	2	3	1	2
CO2	3	3	3	2	2	1	0	2	1	2	3	2	3	1	2
CO3	3	3	3	2	3	0	0	1	0	1	2	3	3	2	3
CO4	3	3	3	3	3	1	1	2	1	2	3	3	3	1	3

Module number	Topic	Sub-topics
1.	**Introduction and Software Process Models**	1. Make a comparative studies of different models of software development process
2.	**Requirement Engineering and Software Project Management**	2. Write an SRS. 3. Compute function points using the method of FPA to determine the cost of s/w project 4. Implement COCOMO using the different formulas 5. Implement Gantt Chart and determine milestones 6. Implement PERT-CPM method
3.	**Software Design and Coding**	7. Implement the Cyclomatic Complexity of coding 8. Implement and evaluate the Halstead's Metrics of Coding 9. Implement Dharma's metrics 10. Implement polymorphism factor formula. 11. Implement inheritance formula
4.	**Testing and Software Quality**	12. Implement H-K information factor. 13. Implement EMV method

Name of the Course: Software Project Management Laboratory	
Course Code: MCA394	**Semester: 3rd**
Duration: 12 Weeks.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Practical: 3	Practical Sessional Internal continuous evaluation: 100
Credit: 3	Practical Sessional external examination: 100

Aim:	
1	To gain knowledge of various aspects of software engineering project management.
2	To enhance ability to identify qualities of a good solution
3	To implement learned algorithm/design techniques to solve problems
Objective:	
1	The fundamental knowledge of software engineering
2	The different basic models need to implement different project problems
3	The various design methods to develop the software system
4	The quality and other issues related to the software products and systems
Pre-Requisite:	
1	Knowledge in fundamental theories of computer science and one programming language
Course Outcome:	
1	On completion of this course students are expected to learn fundamentals and different models of software engineering.
2	On completion of this course students are expected to learn different aspects of requirement analysis in software project management.
3	On completion of this course students are expected to learn various types of software design and concepts of coding.
4	On completion of this course students are expected to learn different types of testing and quality issues.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	1	2	0	0	1	0	1	1	2	3	1	2
CO2	3	3	3	2	2	1	0	2	1	2	3	2	3	1	2
CO3	3	3	3	2	3	0	0	1	0	1	2	3	3	2	3
CO4	3	3	3	3	3	1	1	2	1	2	3	3	3	1	3

Module number	Topic	Sub-topics
1.	**Introduction and Software Process Models**	Software, Software Engineering, Myths, Software Process, Work Products, Importance of SoftwareEngineering, Standard for Software Process, Waterfall Model, Prototyping Model, Iterative Enhancement Model, Spiral Model, RAD model.
2.	**Requirement Engineering and Software Project Management**	Software Requirements, Types of Requirements, Requirement Engineering Cycle, Requirements Specification document, Characteristics of Requirements,Requirement verification and validation, Role of Management in Software Development, ProjectEstimation Techniques, Staffing, Scheduling, Earned Value Analysis, SoftwareRisks, SoftwareConfiguration Management,Software Process and Project metrics.
3.	**Software Design and Coding**	Process, Data and Behavioural Modelling, Design Concepts, Modularity, Architectural design,Coupling and Cohesion, Top-downand bottom-up design, Object- oriented Analysis, Function-oriented and Object-Oriented Designapproach, Software DesignDocument, Coding styles anddocumentation,
4.	**Testing and Software Quality**	Testing principles, testing strategies, Black-box and White- box Testing Techniques, Levels of testing -unit, integration, system, regression, Test Plan, Test Cases Specification, Software debugging,Software Maintenance, Software Quality Factors, ISO , SEI CMM, CMMI, Software Reliability. Software Availability.

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ ISSN/ ISBN**	**Name of the Publisher**
Rajib Mall	Fundamentals of Software Engineering	4th ed	PHI
Reference Books:			
Roger S. Pressman	Software Engineering, A Practitioners Approach	7th ed	MGH

Course Name: Values and Ethics Credit: 1

Course Code: MCA305 Lecture Hours: 40

Name of the Course: Values and Ethics	
Course Code: MCA305	**Semester: 3rd**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 3	
Aim:	
1	To gain knowledge of various aspects general ethics and energy in life.
2	To get ability to identify relations among technology, engineering and human aspects
3	To implement values in various aspects of life with morality.
Objective:	
1	An ability to analyze a problem, then identify and formulate the computing requirements appropriate to its solution
2	Development of Solutions- An ability to design, implement and evaluate a Computer based problems with appropriate consideration for public health and safety, cultural, societal and environmental considerations.
3	Conduct investigations of complex problem – An ability to design and conduct experiments, as well as to analyze and interpret data to reach valid conclusions.
4	An ability to analyze a problem, then identify and formulate the computing requirements appropriate to its solution
Pre-Requisite:	
1	Knowledge in General Studies, Fundamentals of Computers, Proficiency in Communication Skills.
Course Outcome:	
1	Understanding the importance and role of science, technology and engineering as knowledge and social-professional world, know the technological growth.
2	To realize the importance of energy as resource and crisis in energy, understand the effect of degradation and pollution of environment, introduce eco-friendly technology.
3	To choose the appropriate technology for development, understand the transfer, assessment and impact of technology, learn the role of human resource in engineering, man-machine interaction, impact of automation, introduce human-centric technology.
4	To determine the relation between profession and human values like value

	crisis in society, life, personality and mental health. know the role/importance of values in law, justice in Indian perspective, know the aesthetic values, learning the relation between morality and ethics and virtue ethics.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	0	1	3	2	0	0	1	0	3	3	0	1
CO2	2	0	1	0	0	3	3	0	0	0	0	2	2	0	0
CO3	3	3	3	1	2	2	2	0	1	1	2	3	3	1	2
CO4	2	0	1	0	0	2	1	3	0	2	0	2	1	0	0

Module number	Topic	Sub-topics
1.	**Introduction and Relation with Energy**	Science, Technology and Engineering as Knowledge and as Social and Professional Activities Effects of Technological Growth Rapid Technological growth and depletion of resources. Reports of the Club of Rome. Limits of growth. Energy Crisis; Renewable Energy Resources Environmental degradation.
2.	**Human, Technology and Engineering Ethics**	Technologies. Environmental Regulations. Environmental Ethics Appropriate Technology Movement of Schumacher Human Operator in Engineering projects and industries. Problems of man machine interaction. Impact of assembly line and automation. Engineering profession: Ethical issues in engineering practice. Conflicts between business demands and professional ideals. Social and ethical Responsibilities of Technologists
3.	**General Values**	Nature of values: Value Spectrum of a 'good' life Psychological values: Integrated personality; mental health
4.	**Other Types of Values and Morality**	The modern search for a 'good' society, Moral and ethical values: Nature of moral judgments; canons of ethics; Ethics of virtue; ethics of duty; ethics of responsibility

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ ISSN/ ISBN**	**Name of the Publisher**
S.K. Sarangi	Values & Ethics of Profession & Business	2nd ed	Asian Books
Reference Books:			
Manna, Chakraborti	Values and Ethics in Business and Profession	1st ed	PHI
Chattopadhyay, Singh	Ethics & Values for Engineers & Managers	1st ed	HPH

Course Name: Environment and Ecology Credit: 2

Course Code: MCA306 Lecture Hours: 40

Name of the Course: Environment and Ecology	
Course Code: MCA306	**Semester: 3rd**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 2	
Aim:	
1	Imparting knowledge about the environment and ecosystem around us.
2	Imparting knowledge about the natural resources, biodiversity, and the importance of their conservation
3	Environmental Management and Pollution Control
Objective:	
1	Students will gain knowledge about the environment and ecosystem.
2	Students will learn about natural resources, biodiversity, and the importance of their conservation
3	To make students aware of problems of environmental pollution, its impact on humans and the ecosystem, and control measures.
4	At the end of the course, students will learn about waste disposal measures and environmental management.
Pre-Requisite:	
1	NA
Course Outcome:	
1	Define Environmental factors and the basic components of the ecosystem.
2	Understand and explain the importance of Plantation.
3	List the pollutants and analyze the importance of reducing/ controlling environmental pollution.
4	Analyze the importance of Biohazards, Environmental and Social safety

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	1	3	3	2	1	2	1	2	2	2	3	2	1	2	2
CO2	3	2	3	2	2	3	1	2	2	1	1	1	3	1	3
CO3	2	2	1	3	1	2	3	3	1	1	2	3	3	3	1
CO4	1	3	1	3	3	2	2	3	2	3	2	1	1	1	2

Module number	Topic	Sub-topics
I	**Overview**	Basic ideas of environment, basic concepts, man, society & environment, their interrelationship Mathematics of population growth and associated problems, Importance of population study in environmental engineering, the definition of resource, types of resource, renewable, non-renewable, potentially renewable, effect of excessive use vis-à-vis population growth, Sustainable Development. Materials balance: Steady state conservation system, steady state system with non-conservative pollutants, step function. Importance, scope and principles of EIA.
II	**Ecology**	Elements of ecology: System, open system, closed system, the definition of ecology, species, population, community, definition of ecosystem- components types and function. (1L) Structure and function of the following ecosystem: Forest ecosystem, Grassland ecosystem, Desert ecosystem, Aquatic ecosystems, Mangrove ecosystem (special reference to Sundar ban); Food chain [definition and one example of each food chain], Food web.(2L) Biogeochemical Cycle- definition, significance, flow chart of different cycles with only elementary reaction [Oxygen, carbon, Nitrogen, Phosphate, Sulphur]. (1L) Biodiversity- types, importance, Endemic species, Biodiversity Hot-spot, Threats to biodiversity, Conservation of biodiversity.(2L)
III	**Air Pollution**	Atmospheric Composition: Troposphere, Stratosphere, Mesosphere, Thermosphere, Tropopause and Mesopause. (1L) Energy balance: Conductive and Convective heat transfer, radiation heat transfer, simple global temperature model [Earth as a black body, earth as albedo], Problems.(1L) Green house effects: Definition, impact of greenhouse gases on the global climate and consequently on sea water level, agriculture and marine food. Global warming and its consequence, Control of Global warming. Earth's heat budget.(1L) Lapse rate: Ambient lapse rate Adiabatic lapse rate, atmospheric stability, temperature inversion (radiation inversion).(2L) Atmospheric dispersion: Maximum mixing depth, ventilation coefficient, effective stack height, smokestack plumes and Gaussian plume model.(2L) Definition of pollutants and contaminants, Primary and secondary pollutants: emission standard, criteria pollutant. Sources and effect of different air pollutants- Suspended particulate matter, oxides of carbon, oxides of nitrogen,

		oxides of sulphur, particulate, PAN. (2L) Smog, Photochemical smog and London smog. Depletion Ozone layer: CFC, destruction of ozone layer by CFC, impact of other green-house gases, effect of ozone modification. (1L) Standards and control measures: Industrial, commercial and residential air quality standard, control measure (ESP. cyclone separator, bag house, catalytic converter, scrubber (ventury), Statement with brief reference). (1L)
IV	**Water Pollution**	Pollutants of water, their origin and effects: Oxygen demanding wastes, pathogens, nutrients, Salts, thermal application, heavy metals, pesticides, volatile organic compounds.DO, 5-day BOD test, Seeded BOD test, BOD reaction rate constants, Effect of oxygen demanding wastes on river [deoxygenating, reaeration], COD, Oil, Greases, pH. Lake: Eutrophication [Definition, source and effect]. Waste water standard [BOD, COD], Water Treatment system,primary and secondary treatments, tertiary treatment definition.Water pollution due to the toxic elements. USEPA and WHO guidelines for drinking water.
V	**Lithosphere**	Lithosphere; Internal structure of earth, rock and soil (1L). Solid Waste: Municipal, industrial, commercial, agricultural, domestic, pathological and hazardous solid wastes; Recovery and disposal method- Open dumping, Land filling, incineration, composting, recycling. Solid waste management and control (hazardous and biomedical waste).(2L)
VI	**Noise pollution**	Definition of noise, effect of noise pollution, noise classification [Transport noise, occupational noise, neighbourhood noise] (1L) Definition of noise frequency, noise pressure, noise intensity, noise threshold limit value, equivalent noise level, L10 (18hr Index) ,n Ld.Noise pollution control. (1L)
VII	**Environmental Management**	Environmental impact assessment, Environmental Audit, Environmental laws and protection act of India, Different international environmental treaty/ agreement/ protocol. (2L)

List of Books Text Books:

1. Environmental Science, Cunningham,TMH
2. Environmental Science,Wright & Nebel,PHI
3. Fundamentals of Ecology,Dash, TMH
4. Environmental Pollution Control Engineering,C.S.Rao,New Age International
5. Environmental Pollution Analysis,S.N.Khopkar,New Age International
6. Environmental Management, N.K. Oberoi, EXCEL BOOKS
7. Environmental Management, Mukherjee, VIKAS
8. Ecosystem Principles & Sustainable Agriculture, Sithamparanathan, Scitech

Course Name: General Studies & Current Affairs - III Credit: 0.5

Course Code: MCA(GS)301 Lecture Hours: 20

Name of the Course: General Studies & Current Affairs - III	
Course Code: MCA(GS)301	**Semester: 3rd**
Duration: 48 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 2	End Semester Exam: 100
Tutorial: 0	Continuous Assessment: 100
Credit: 0.5	

Aim:	
Sl. No.	
1	To enhance quantitative aptitude and logical reasoning for effective problem-solving in professional environments.
2	To develop oral, listening, and reading communication skills for workplace interactions.
3	To equip students with practical communication skills, including professional correspondence and presentations.

Objective:	
Sl. No.	
1	To strengthen quantitative and logical reasoning for analytical decision-making.
2	To improve verbal and non-verbal communication skills, including articulation and clarity.
3	To enhance listening and reading comprehension for better information processing.
4	To develop practical communication techniques, such as presentations, business communication, and public speaking.

Pre-Requisite:	
Sl. No.	
1.	Basic understanding of English language and fundamental mathematical concepts.

Module number	Topic	Sub-topics
1	**History**	1. 1. Pre sultanate age: Md. Bipin Karim, Aluptagin, Sabuktagin, Sultan Mamud, Md. Ghori 2. .Delhi Sultanate: Slave dynasty, Khalji dynasty, Tughlaw dynasty, Sayyed dynasty, Lodhi dynasty 3. Bhakti and Sufi movement: Kabir, Gurunanak, Chaitanya, Namdev 4. Mughal Period: Babur, Humayun, Sher shah suri, Akbar, Jehangir, Shah Jahan, Aurangzeb, Aministrative system, Din-i-ilahi, Art and architecture, Land revenue system
2	**Geography**	1. Drainage system • Types of river (Perennial, Non perennial, Inland drainage) • Courses of river: Upper, Middle, Lower courses • Landforms carved out by river based on the courses. • Basic terminologies: Antecedent rivers, Consequent rivers, Fault guided river, Tributary, Distributary • Indian river system (Himalayan, Peninsular, Coastal) 2. Types of Irrigation in India • Well • Tanks • Canal 3. Problems of irrigation in India 4. Status of Irrigation in India as per 2011 census 5. Clouds and Precipitation: Forms of precipitation, Types of rainfall, Types of clouds.
3	**Macro Economics**	1. National income- Concept of GDP, GNP, NNP both in FC & MP, PCI 2. Tax – Concept of TAX , objective of TAX, Direct & Indirect Tax, Progressive, Regressive &Proportional tax. 3. RBI & Banking- Traditional Functions of RBI, CRR, SLR, REPO, Reverse repo, MSF, LAF market, capital market, capital market, Money market, FOREX. 4. Budget- concept of budget, components of budget, different types of deficit. 5. Keynesian outlook- IS,LM & different multipliers. 6. Inflation& Deflation- Inflation & its impact, Deflation & its impact, WPI, CPI, GDP deflator.
4	**Constitution**	1. Central State relation, Interstate relation, 2. Supreme Court-Appointment of Chief Justice, Acting Chief Justice, Qualification, Oath or Affirmation, Tenure of Judge, Removal of Judges, Salaries & allowance, Adhoc Judge, Procedure of the court, write jurisdiction, Power of Judicial review 3. High Court-Appointment of Chief Justice, Acting Chief Justice, Qualification, Oath or Affirmation, Tenure of Judge, Removal of

		Judges, Salaries & allowance, Adhoc Judge, Procedure of the court, write jurisdiction, Power of Judicial review 4. Duties& Powers of Attorney & Advocate General in Brief 5. Panchayati Raj- Three tier system, Different committees recommendation 6. Municipality, Municipal Council & Corporation, Official Languages & related Articles

References

1. History:

India's Ancient Past (Ancient History) : R.S. Sharma

History of medieval India (Medieval History): Satish Chandra History of Modern India (Modern History): Bipin Chandra

India's struggle for Independence (Modern History): Bipin Chandra

Geography:

India- Khullar Economics:

Indian Economy- TATA Mc Graw Hill/Ramesh Singh Indian Economy – Arihant

Constitution:

Indian Constitution- D.D. Basu

Our Constitution- Subhash.C. Kashyap

Name of the Course: Competitive Aptitude Training – III	
Course Code: MCA(GS)381	**Semester: 3rd**
Duration: 20 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 2	End Semester Exam: 100
Tutorial: 0.5	Continuous Assessment: 100
Credit: 1	

Aim:	
Sl. No.	
1	To develop verbal reasoning, analytical thinking, and problem-solving skills essential for competitive exams and job recruitment.
2	To enhance English language proficiency, focusing on grammar, comprehension, and professional communication.
3	To build expertise in data interpretation, improving the ability to analyze and solve quantitative problems efficiently.

Objective:	
Sl. No.	
1	To strengthen English grammar, vocabulary, and comprehension skills for competitive aptitude tests.
2	To develop logical and analytical reasoning through structured problem-solving techniques.
3	To enhance proficiency in data interpretation, including tabular and graphical analysis.
4	To improve formal communication skills, such as official letter writing, for professional settings.

Pre-Requisite:	
Sl. No.	
1.	Basic understanding of English grammar and elementary mathematical reasoning.

Module number	**Topic**	**Sub-topics**
1	**Quantitative Aptitude**	1 Simple & Compound Interest- Basic concept of SI & CI, different formulas & their applications, concept of Growth & Contraction of Business. 2 Data Interpretation- Tables, pie chart, histogram, Bar chart, solution tricks & techniques. 3 Quant Review- Miscellaneous problems from different chapters & short cuts.

		4 Indices & Surds- Basic concept, Formulae & their applications, Finding out the square roots, Elimination of Surds, Equation solve. 5 Quadratic Equation- polynomials, degree, powers, Equation & factors Solution. Progression- Concept of AP, GP & HP
2	**Objective English**	1. Error based on Noun & Pronoun. 2. Error based on Adjective & Degree of comparison. 3. Error based on Adverb & Synonym And Antonym. 4. Error Based on Verbs & Some Special Phrasal Verbs. 5. Reading Comprehension Passage.
3	**Logical Mental Ability**	1 a)Statement And Assumption, b)Statement And Conclusion, c)Statement And Course Of Action, d)Cause And Effect, e)Drawing Inference 2 Machine Input-Output a) Pattern Based I/O 3 Inequality a) Coded Inequality, b) Jumbled Inequality, c) Conditional inequality 4 Calendar And Clock a) Miscellaneous Problems
4	**Computer proficiency**	C programming, Basics of C++

List of Books Text Books:

Numerical Aptitude

1. Fastrack objective Arithmetic: Arihant
2. Quantitative aptitude for Competitive exam (4th Edition): TATA Mc Graw Hill
3. Quantitative aptitude for Competitive exam (3rd Edition): PEARSON

Verbal Ability

1. Objective English: Kiran Publication

2. General English: Arihant

LOGICAL REASONING

1. Analytical &Logical Reasoning: M.K. Pandey/B.S.C. Publication
2. A modern approach to verbal & non verbal Reasoning: R.S. Agarwal.

Course Name: Minor Project Credit: 6

Course Code: MCA391

Name of the Course: Minor Project	
Course Code: MCA391	**Semester: 3rd**
Duration: 12 Weeks.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Practical: 0	Practical Sessional Internal continuous evaluation: 100
Credit: 6	Practical Sessional external examination: 100

Aim:	
Sl. No.	
1	To enable students to apply theoretical knowledge to real-world problems through project development.
2	To enhance problem-solving, software development, and research skills.
3	To develop teamwork, project management, and documentation abilities.
Objective:	
Sl. No.	
1	To identify and define a real-world computing problem.
2	To design and implement a software solution using appropriate tools and technologies.
3	To analyze and evaluate the efficiency and effectiveness of the developed system.
4	To document and present the project findings professionally.
Pre-Requisite:	
Sl. No.	
1.	Knowledge of programming, databases, and software development lifecycle (SDLC).
Course Outcome:	
1.	Identify and define a problem statement relevant to computing applications.
2.	Develop a functional prototype or software solution using modern tools.
3.	Demonstrate analytical and technical skills in project execution.
4.	Present a well-documented project report with findings and future scope.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	2	2	1	0	1	1	1	2	2	3	2	2
CO2	3	3	3	2	3	1	0	1	1	1	2	3	3	2	3
CO3	3	3	3	2	3	1	0	1	1	2	2	3	3	2	3
CO4	2	2	2	1	2	1	0	1	1	3	2	2	2	1	2

Course Name: Industrial Training Credit: 2

Course Code: MCA381

Name of the Course: Industrial Training	
Course Code: MCA391	**Semester: 3rd**
Duration: 12 Weeks.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Practical: 0	Practical Sessional Internal continuous evaluation: 100
Credit: 2	Practical Sessional external examination: 100
Aim:	
Sl. No.	
1	To provide practical industry exposure and application of theoretical knowledge.
2	To enhance problem-solving, teamwork, and professional skills in a real-world environment.
3	To familiarize students with latest technologies, tools, and best practices in the industry.
Objective:	
Sl. No.	
1	To apply academic knowledge to real-world projects.
2	To develop technical, analytical, and professional skills through hands-on experience.
3	To understand industry workflows, ethics, and project management methodologies.
4	To improve communication, collaboration, and adaptability in a corporate setting.
Pre-Requisite:	
Sl. No.	
1.	Completion of core MCA coursework in programming, databases, and software development.
Course Outcome:	
1.	Demonstrate technical proficiency by working on industry-relevant projects.
2.	Apply problem-solving and analytical skills to real-world challenges.
3.	Exhibit teamwork, communication, and professionalism in an industrial setting.
4.	Gain exposure to emerging technologies, tools, and best industry practices.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	3	1	3	1	0	1	2	1	2	3	3	2	2
CO2	3	3	3	2	3	2	0	1	2	1	2	3	3	3	3
CO3	2	2	2	1	2	2	0	2	3	3	3	2	2	1	2
CO4	3	2	3	1	3	2	1	1	2	2	2	3	3	2	3

Syllabus for MCA Admission Batch 2021, 4th Semester

Course Name: Distributed Database Management **Credit: 3**

Course Code: MCA401A **Lecture Hours: 40**

Name of the Course: Operating Systems and Systems Software	
Course Code: MCA301	**Semester: 4th**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 3	
Aim:	
1	Develop a deep understanding of distributed database architecture and design principles.
2	Equip students with skills for optimizing distributed query processing and managing transactions.
3	Enable application of data warehousing, OLAP, and data mining techniques for real-world problem-solving.
Objective:	
1	Understand the architecture and design of distributed database systems.
2	Apply techniques for distributed query processing and optimization.
3	Master the concepts of distributed transaction processing and data warehousing.
4	Utilize data mining methods such as association analysis, classification, and clustering.
Pre-Requisite:	
1	Fundamentals of Database Management Systems, Basic Knowledge of Computer Networks, Programming Skills & Operating systems
Course Outcome:	
1	Understand and explain the architecture and design principles of distributed database systems.
2	Apply methods and techniques for distributed query processing and optimization.
3	Understand the concepts of distributed transaction processing, data warehousing, and OLAP technology.
4	Apply methods and techniques for data association analysis, classification, and clustering.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	1	1	1	1	0	0	0	2	3	1	1
CO2	2	2	1	1	2	1	0	1	0	0	0	1	2	1	1
CO3	3	2	2	2	3	1	1	1	0	0	0	2	3	2	1
CO4	3	3	2	2	2	1	1	2	1	1	1	2	3	1	2

Module number	Topic	Sub-topics
1	**Introduction to Distributed Database Management System**	Distributed DBMS features and needs. Reference architecture. Levels of distribution transparency, and replication. Distributed database design – fragmentation, allocation criteria. Storage mechanisms. Translation of global queries. / Global query optimization. Query execution and access plan. Concurrency control – 2 phases locks. Distributed deadlocks. Time-based and quorum-based protocols. Comparison. Reliability- non-blocking commitment protocols.
2	**Partitioned Networks**	Partitioned networks. Checkpoints and cold starts. Management of distributed transactions- 2-phase unit protocols. Architectural aspects. Node and link failure recoveries.
3	**Distributed Database Administration**	Distributed data dictionary management. Distributed database administration. Heterogeneous databases-federated database, reference architecture, loosely and tightly coupled. Alternative architecture. Development tasks, Operation- global task management. Client-server databases- SQL server, open database connectivity. Constructing an application.

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ ISSN/ ISBN**	**Name of the Publisher**
Stefano Ceri & Giuseppe Pelagatti	Distributed Databases: Principles and Systems	978-0070265110	McGraw Hill Education

Course Name: Image Processing Credit: 3

Course Code: MCA401B Lecture Hours: 40

Name of the Course: Image Processing	
Course Code: MCA401B	**Semester: 4th**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 3	
Aim:	
1	Equip students with a solid understanding of the core principles and techniques used in image processing.
2	Enable students to apply image processing methods to analyze, enhance, and manipulate digital images for various applications.
3	Prepare students to solve complex real-world problems related to image analysis, computer vision, and pattern recognition.
Objective:	
1	Understand the fundamental principles and techniques of image processing.
2	Apply methods to enhance and manipulate digital images.
3	Develop skills in image analysis and computer vision.
4	Solve real-world problems using image processing techniques.
Pre-Requisite:	
1	Fundamentals of Database Management Systems, Basic Knowledge of Computer Networks, Programming Skills & Operating systems
Course Outcome:	
1	To study the image fundamentals and mathematical transforms necessary for image processing.
2	To study the image enhancement techniques
3	To study image restoration procedures
4	To study the image compression procedures

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	1	1	1	1	0	0	0	2	3	1	1
CO2	2	2	1	1	2	1	0	1	0	0	0	1	2	1	1
CO3	3	2	2	2	3	1	1	1	0	0	0	2	3	2	1
CO4	3	3	2	2	2	1	1	2	1	1	1	2	3	1	2

Module number	Topic	Sub-topics
1	**Introduction and Digital Image Fundamentals** **Image enhancement in the Spatial domain**	Digital Image Fundamentals, Human visual system, Image as a 2D data, Image representation – Grayscale and Colour images, image sampling and quantization Basic grey level Transformations, Histogram Processing Techniques, Spatial Filtering, Low pass filtering, High pass filtering
2	**Filtering in the Frequency Domain** **Image Restoration and Reconstruction**	Preliminary Concepts, Extension to functions of two variables, Image Smoothing, Image Sharpening, Homomorphic filtering. Noise Models, Noise Reduction, Inverse Filtering, MMSE (Wiener) Filtering
3	**Colour Image Processing** **Image Compression**	Colour Fundamentals, Color Models, Pseudo colour image processing Fundamentals of redundancies, Basic Compression Methods: Huffman coding, Arithmetic coding, LZW coding, JPEG Compression standard
4	**Morphological Image Processing**	Erosion, dilation, opening, closing, Basic Morphological Algorithms: hole filling, connected components, thinning, , skeletons
5	**Image Segmentation** **Object Recognition and Case Studies Object Recognition**	Point, line and edge detection, Thresholding, Regions Based segmentation, Edge linking and boundary detection, Hough transform Patterns and pattern classes, recognition based on decision-theoretic methods, structural methods, case studies – image analysis Application of Image processing in process industries

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ ISSN/ ISBN**	**Name of the Publisher**
Chandra& Majumder	Digital Image Processing &Analysis	2nd Edition	PHI
Anil K. Jain	Fundamentals of Digital Image Processing	1st Edition	Pearson

Course Name: Parallel Programming Credit: 3

Course Code: MCA401C Lecture Hours: 40

Name of the Course: Parallel Programming	
Course Code: MCA401C	**Semester: 4th**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 3	
Aim:	
1	Equip students to write efficient parallel programs for faster computation.
2	Prepare students for industry applications in high-performance and big-data computing.
3	Foster critical thinking and innovation in solving computational challenges with parallel techniques.
Objective:	
1	Understand the fundamental principles and techniques of image processing.
2	Apply methods to enhance and manipulate digital images.
3	Develop skills in image analysis and computer vision.
4	Solve real-world problems using image processing techniques.
Pre-Requisite:	
1	Fundamentals of Database Management Systems, Basic Knowledge of Computer Networks, Programming Skills & Operating systems
Course Outcome:	
1	Understand the evolution of High-Performance Computing (HPC) with respect to lawsand the contemporary notion that involves mobility for data, hardware devices and software agents
2	Understand, appreciate and apply parallel and distributed algorithms in Problem Solving.
3	Evaluate the impact of network topology on parallel/distributed algorithm formulationsand traffic their performance.
4	Gain hands-on experience with agent-based and Internet-based parallel and distributed programming techniques.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	2	3	2	1	1	0	1	1	2	3	1	2
CO2	3	3	3	3	3	2	1	0	2	1	2	3	3	3	3
CO3	3	3	2	2	3	2	2	1	0	1	1	2	2	2	3
CO4	3	3	3	3	3	1	1	0	2	1	2	3	3	3	3

Module number	Topic	Sub-topics
1	**Fundamentals of Parallel Programming**	Processes and processors. Shared memory. Fork. Join constructs. Basic parallel programming techniques- loop splitting, spin locks, contention barriers and row conditions. Variations in splitting, self and indirect scheduling.
2	**Data Dependency and Scheduling Techniques**	Data dependency-forward and backward block scheduling. Linear recurrence relations. Backward dependency.
3	**Advanced Performance Tuning and Parallel Programming Techniques**	Performance tuning overhead with a number of processes, effective use of cache. Parallel programming examples: Average, mean squared deviation, curve fitting, numerical integration, travelling salesman problem, Gaussian elimination. Discrete event time simulation. Parallel Programming Constructs in HPF, FORTRAN 95. Parallel programming under Unix.

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ ISSN/ ISBN**	**Name of the Publisher**
Quinn	Parallel Computing	2nd Edition	TMH

Course Name: Cloud Computing Credit: 3

Course Code: MCA401D Lecture Hours: 40

Name of the Course: Parallel Programming	
Course Code: MCA401D	**Semester: 4th**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 3	
Aim:	
1	Analyze the Evolution and Impact of Cloud Computing
2	Evaluate Cloud Computing Service Models and Deployment Strategies
3	Investigate Security Challenges and Solutions in Cloud Computing
Objective:	
1	To understand the fundamental concepts of cloud computing.
2	To explore different cloud service models and cloud deployment models.
3	To gain practical knowledge on cloud storage, virtualization, and cloud security.
4	To comprehend the economic, organizational, and technological aspects of cloud computing and development of applications leveraging cloud-based services and APIs.
Pre-Requisite:	
1	Basic understanding of computer networks, operating systems, and internet technologies.
Course Outcome:	
1	Understand and explain the key concepts and principles of cloud computing, including its architecture, components, and models.
2	Differentiate between various cloud service models (IaaS, PaaS, SaaS) and deployment models (public, private, hybrid, community), and assess their suitability for different scenarios.
3	Apply virtualization techniques and cloud storage solutions to design and manage scalable and efficient cloud-based systems.
4	Analyse cloud security mechanisms and issues, and implement strategies to safeguard data and applications in the cloud environment.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	1	1	1	1	0	0	0	2	3	1	1
CO2	2	2	1	1	2	1	0	1	0	0	0	1	2	1	1
CO3	3	2	2	2	3	1	1	1	0	0	0	2	3	2	1
CO4	3	3	2	2	2	1	1	2	1	1	1	2	3	1	2

Module number	Topic	Sub-topics
1	**Introduction to Cloud Computing and Cloud Service Models**	Definition and Essential Characteristics of Cloud Computing, History and Evolution of Cloud Computing, Benefits and Challenges of Cloud Computing, Cloud Computing Architecture, Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), Function as a Service (FaaS)
2	**Cloud Deployment Models**	Public Cloud, Private Cloud, Hybrid Cloud, Community Cloud
3	**Virtualization**	Concepts of Virtualization, Types of Virtualization (Server, Network, Storage), Virtual Machines (VMs), Containers and Docker
4	**Cloud Storage**	Storage as a Service (STaaS), Cloud Storage Architectures, Storage Types: Block, File, and Object Storage, Examples: Amazon S3, Google Cloud Storage
5	**Cloud Security and Cloud Networking**	Security Issues in Cloud Computing, Identity and Access Management (IAM), Data Protection and Encryption, Regulatory and Compliance Issues, Networking Basics for Cloud, Software-Defined Networking (SDN), Network Function Virtualization (NFV), Cloud Load Balancing
6	**Cloud Application Development and Future Trends**	Developing Cloud-Native Applications, Microservices Architecture, DevOps and CI/CD Pipelines, Example Platforms: AWS Lambda, Google Cloud Functions Edge Computing, Serverless Computing, Quantum Cloud Computing, AI and Machine Learning in the Cloud

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ ISSN/ ISBN**	**Name of the Publisher**
Rajkumar Buyya, Christian Vecchiola, Sb Thamarai Selvi	Chapter 1: Introduction Mastering Cloud Computing	1st / 978-1259029950	Mc Graw Hill
Rajkumar Buyya, Christian Vecchiola, Sb Thamarai Selvi	Chapter 3: Virtualization Mastering Cloud Computing	1st / 978-1259029950	Mc Graw Hill
Rajkumar Buyya, Christian Vecchiola, Sb Thamarai Selvi	Chapter 4: Cloud Computing Architecture Mastering Cloud Computing	1st / 978-1259029950	Mc Graw Hill
Rajkumar Buyya, Christian Vecchiola, Sb Thamarai Selvi	Chapter 9: Cloud Platforms in Industry Mastering Cloud Computing	1st / 978-1259029950	Mc Graw Hill
Rajkumar Buyya, Christian Vecchiola, Sb Thamarai Selvi	Chapter 10: Cloud Applications Mastering Cloud Computing	1st / 978-1259029950	Mc Graw Hill
Rajkumar Buyya, Christian Vecchiola, Sb Thamarai Selvi	Chapter 5: Virtual Machines Provisioning and Migration Services Mastering Cloud Computing	1st / 978-1259029950	Mc Graw Hill
Arshdeep Bahga, Vijay Madisetti	Chapter 12: Cloud Security Cloud Computing A Hands-On Approach	1st / 9788173719233	University Press
Reference Books:			
Thomas Erl, Zaigham Mahmood, Ricardo Puttini	Cloud Computing: Concepts, Technology & Architecture	1st / 978-0133387520	Prentice Hall
Michael J. Kavis	Architecting the Cloud: Design Decisions for Cloud Computing Service Models (SaaS, PaaS, and IaaS)	1st / 978-1118617618	Wiley
Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi	Mastering Cloud Computing: Foundations and Applications Programming	1st / 978-0124114548	Morgan Kaufmann

Course Name: Compiler Design Credit: 3

Course Code: MCA402A Lecture Hours: 40

Name of the Course: Compiler Design	
Course Code: MCA402A	**Semester: 4th**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 3	
Aim:	
1	To gain Knowledge of Various aspects of a Compiler.
2	To enhance Ability to identify qualities of a good solution of NFA, DFA etc.
3	To implement NFA to DFA conversion techniques and different parsing methods to solve problems.
Objective:	
1	Provide you with the knowledge and expertise to become a proficient compiler design.
2	Demonstrate an understanding of parsing and polishing expression concepts that are vital for compiler design.
3	To produce DFA from an NFA to understand a basic compiler.
4	Critically evaluate NFA based on their design and create DFA from that.
Pre-Requisite:	
1	Proficiency in data structure, graph theory, automata theory and C programming.
Course Outcome:	
1	Understand fundamentals of compiler and identify the relationships among different phases of the compiler.
2	Understand the application of finite state machines, recursive descent, production rules, parsing, and language semantics.
3	Analyze & implement required module, which may include front-end, back-end, and a small set of middle-end optimizations.
4	Use modern tools and technologies for designing new compiler.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	-	-	-	-	-	-	-	-	-	2	-	-
CO2	2	3	2	2	-	-	-	-	-	-	-	-	2	2	-
CO3	2	3	3	2	-	-	-	-	1	1	2	1	2	3	2
CO4	2	2	2	3	3	-	-	-	1	1	2	2	-	3	3

Module number	Topic	Sub-topics
1	**Introduction to Compiling**	**Introduction to Compiling:** Compilers, Analysis of the source program, The phases of a compiler, Cousins of the compiler, The grouping of phases, Compiler-construction tools **A Simple One-Pass Compiler:** Overview, Syntax definition, Syntax-directed translation, Parsing, A translator for simple expressions, Lexical analysis, Incorporating a symbol table, Abstract stack machines, Putting the techniques together **Lexical Analysis:** The role of the lexical analyzer, Input buffering, Specification of tokens, Recognition of tokens, A language for specifying lexical analyzers, Finite automata, From a regular expression to an NFA, Design of a lexical analyzer generator, Optimization of DFA-based pattern matchers
2	**Syntax Analysis**	**Syntax Analysis:** The role of the parser, Context-free grammars, Writing a grammar, Top-down parsing, Bottom- up parsing, Operator-precedence parsing, LR parsers, Using ambiguous grammars, Parser generators **Syntax-Directed Translation:** Syntax-directed definitions, Construction of syntax trees, Bottom-up evaluation of S-attributed definitions, L-attributed definitions, Top-down translation, Bottom-up evaluation of inherited attributes, Recursive evaluators, Space for attribute values at compile time, Assigning space at compile time, Analysis of syntax-directed definitions

3	**Type Checking**	**Type Checking:** Type systems, Specification of a simple type checker, Equivalence of type expressions, Type conversions, Overloading of functions and operators, Polymorphic functions, An algorithm for unification **Run-Time Environments:** Source language issues, Storage organization, Storage-allocation strategies, Access to nonlocal names, parameter passing, Symbol tables, Language facilities for dynamic storage allocation, Dynamic storage allocation techniques
4	**Intermediate Code Generation**	**Intermediate Code Generation:** Intermediate languages, Declarations, Assignment statements, Boolean expressions, Case statements, Back Patching, Procedure calls **Code generation:** Issues in the design of a code generator, The target machine, Run-time storage management, Basic blocks and flow graphs, Next-use information, A Simple code generator, Register allocation and assignment, The dag representation of basic blocks, Peephole optimization, Generating code from dags, Dynamic programming code-generation algorithm, Code-generator generators **Code Optimization:** Introduction, The Principal sources of optimization, Optimization of basic blocks, Loops in flow graphs, Introduction to global data-flow analysis, Iterative solution of data-flow equations, Code- improving transformations, Dealing with aliases, Data-flow analysis of structured flow graphs, Efficient data-flow algorithms, A tool for data-flow analysis, Estimation of types, Symbolic debugging of optimized code.

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ ISSN/ ISBN**	**Name of the Publisher**
Aho, Lam, Sethi, Ullman	Compilers – Principles, Techniques & Tools	2nd Edition	Pearson
Holub	Compiler Design in C	2nd Edition	Prentice Hall
Mishra, Chandrasekaran	Theory of Computer Science: Automata, Languages and Computation	3rd Edition	PHI

Course Name: Mobile Computing Credit: 3

Course Code: MCA402B Lecture Hours: 40

Name of the Course: Mobile Computing	
Course Code: MCA402B	**Semester: 4th**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 3	
Aim:	
1	To understand the fundamental concepts and technologies driving mobile computing
2	To understand Mobile Networking and Connectivity
3	To address challenges in mobile security and optimization
Objective:	
1	Gain a foundational understanding of mobile communication systems, including cellular networks and their evolution.
2	Grasp the core concepts of mobile networking protocols, covering aspects like network layers and routing in unique mobile environments.
3	Explore the various mobile communication technologies and protocols.
4	Develop critical knowledge of security challenges and solutions for mobile computing devices and applications.
Pre-Requisite:	
1	Knowledge of computer fundamentals and networking concepts.
Course Outcome:	
1	Define mobile technologies in terms of hardware, software, and communications.
2	Utilize mobile computing nomenclature to describe and analyze existing mobile computing frameworks and architectures.
3	Evaluate the effectiveness of different mobile computing frameworks.
4	Describe how mobile technology functions to enable other computing technologies.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	1	1	2	2	3	2	1	0	0	0	0	0	3	2	2
CO2	2	3	2	2	3	2	1	0	0	0	0	0	3	3	2
CO3	2	3	2	2	3	2	1	0	0	0	0	0	3	3	2
CO4	2	2	2	2	3	2	1	0	0	0	0	0	3	3	2

Module number	Topic	Sub-topics
1	**Introduction:** **Wireless Transmission:** **Access Control:**	Introduction and Application of Mobile Computing Wireless Transmission: Frequency for radio transmission, Signals, Antennas, Signal propagation, Multiplexing, Modulation, Spread spectrum, Cellular systems, Medium Access Control: Motivation for a specialised MAC: Hidden and Exposed terminals. Near and Far terminals; SOMA, FOMA; TOMA: Fixed TOM, Classical Aloha, Slotted Aloha, Carrier sense multiple access, Demand assigned multiple access, PRMA packet reservation multiple access, PRMA packet reservation multiple access, reservation TOMA, Multiple access with collision avoidance, Polling, Inhibit sense multiple access
2	**CDMA:** **GSM:**	CDMA: Spread Aloha multiple access Telecommunication Systems: GSM: Mobile Services, System Architecture, radio interface, Protocols, Localization and Calling, Handover, Security, New Data Services, DECT, Systems Architecture Protocol Architecture: TETRA I, UMTS and IMT-2000, UMTS Basic Architecture, UTRA FDD mode, UTRA TDD mode
3	**Satellite Systems:** **Wireless LAN:** **IEEE 802.11:**	Satellite Systems: History, Applications, Basics: GEO, LEO, MEO, Routing, Localization. Handover Examples: Broadcast Systems: Overview, Cyclic Repetition, Digital Audio; broadcasting: Multimedia object transfer Protocol; Digital Video Broadcasting Wireless LAN: Infrared vs. Radio Transmission, Infrastructure and Ad Hoc networks, IEEE 802.11: System Architecture, Protocol Architecture, Physical Layer, Medium Access Control Layer, MAC management, Future development; HIPERLAN: Protocol architecture, Physical Layer Channel access control. Sub layer, Medium Access control sub layer, Information bases and networking;

	Bluetooth:	Bluetooth: User Scenarios, Physical Layer, MAC layer, Networking, Security, Link management. Wireless ATM: Motivation for WATM, Wireless ATM working group, WATM services, Reference model: Example configurations, Generic reference model;
4	**Handover:** **Location management:** **Mobile Network Layer:**	Handover: Handover reference model, Handover requirements, Types of handover, Handover scenarios, Backward handover, Forward handover; Location management: Requirements for location management, Procedures and Entities; Addressing, Mobile quality of service, Access point control protocol. Mobile Network Layer: Mobile IP: Goals, assumptions and requirements, Entities and Terminology, IP packet delivery, Agent advertisement and discovery, Registration,
5	**Tunneling** **Ad hoc networks** **Mobile Transport Layer**	Tunneling and Encapsulation, Optimizations, Reverse Tunnelling, Ipv6; Dynamic host configuration protocol, Ad hoc networks: Routing, Destination sequence distance vector, Dynamic source routing, Hierarchical algorithms, Alternative metrics. Mobile Transport Layer: Traditional TCP: Congestion control, Slow start, Fast retransmit/fast recovery, Implications on mobility; Indirect TCP, Snooping TCP, mobile RCP, Fast retransmit/fast recovery, Transmission/time-out freezing, Selective retransmission, Transaction oriented TCP. Support for Mobility:
6	**File systems:** **Wireless application protocol:**	File systems: Consistency, Examples; World Wide Web: Hypertext transfer protocol, Hypertext markup language, Some approaches that might help wireless access, System architectures; Wireless application protocol: Architecture, Wireless datagram protocol, Wireless transport layer security, Wireless transaction protocol, Wireless session protocol, Wireless application environment, Wireless markup language; WML script, Wireless telephony application, Examples "Stacks with WAP, Mobile databases, Mobile agents. Security and privacy aspects of Mobile

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ ISSN/ ISBN**	**Name of the Publisher**
Jochen Schiller	Mobile Communications	2nd Edition	Pearson
Reference Books:			
William Stallings	Wireless Communications and Networks		PHI
Rappaport	Wireless Communications Principals and Practices	2nd Edition	Pearson
Ashoke Talukder	Mobile Computing	2nd Edition	TMH

Course Name: Embedded Systems Credit: 3

Course Code: MCA402C Lecture Hours: 40

Name of the Course: Embedded Systems	
Course Code: MCA402C	**Semester: 4th**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 3	
Aim:	
1	To introduce the fundamentals, architecture, and applications of embedded systems.
2	To develop an understanding of real-time operating systems (RTOS), microcontrollers, and interfacing techniques.
3	To familiarize students with design methodologies, programming, and debugging of embedded systems.
Objective:	
1	To understand embedded system architecture, components, and real-time constraints.
2	To explore microcontrollers, memory management, and peripheral interfacing.
3	To learn embedded programming using C and Assembly for real-world applications.
4	To analyze real-time operating systems (RTOS), task scheduling, and power management in embedded systems.
Pre-Requisite:	
1	Basic knowledge of computer architecture, C programming, and operating systems.
Course Outcome:	
1	Explain the architecture, design, and components of embedded systems.
2	Implement microcontroller-based applications with peripheral interfacing.
3	Develop embedded software using C and Assembly programming.
4	Analyze the role of RTOS, scheduling, and power management in embedded systems.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	1	2	1	1	0	1	1	1	2	3	1	2
CO2	3	2	3	2	3	1	1	0	1	1	1	2	3	2	3
CO3	3	3	3	2	3	1	1	0	1	1	1	3	3	2	3
CO4	3	3	3	2	3	2	2	1	1	1	1	3	3	2	3

Module number	Topic	Sub-topics
1	**Introduction to Embedded Systems:**	Definition of Embedded System, Embedded Systems Vs General Computing Systems, History of Embedded Systems, Classification of Embedded Systems, Relation between Microcontroller and Embedded System, Major Application Areas, Purpose of Embedded Systems, Characteristics and Quality Attributes of Embedded Systems
	Embedded Processors:	Types of Embedded Processors, Microprocessors, Microcontrollers, DSP, Embedded Processors from Future Electronics, Applications for embedded processors, Choosing the Right Embedded Processor.
2	**Embedded Systems**	Application- and Domain-Specific: Washing Machine-Application Specific Example of Embedded System, Automotive- Domain Specific Example of Embedded System. The core of the Embedded System: General Purpose and Domain Specific Processors, ASICs, PLDs, Commercial Off-The-Shelf Components (COTS), Embedded Memories: Scratchpad Memories, Cache Memories, Flash Memories, Memory according to the type of Interface, Memory Shadowing and memory selection for Embedded Systems, Sensors and Actuators. Communication Interface: Onboard and External Communication Interfaces.
3	**Embedded Firmware:**	Reset Circuit, Brown-out Protection Circuit, Oscillator Unit, Real Time Clock, Watchdog Timer, Embedded Firmware Design Approaches and Development Languages.
	RTOS-Based Embedded System Design:	Operating System Basics, Types of Operating Systems, Tasks, Process and Threads, Multiprocessing and Multitasking, Task Scheduling.
4	**Task Communication:**	Shared Memory, Message Passing, Remote Procedure Call and Sockets

	Task Synchronization: **Trends in Embedded Industry:**	Task Communication/Synchronization Issues, Task Synchronization Techniques, Device Drivers, How to Choose an RTOS. Processor Trends in Embedded System, Embedded OS Trends, Development Language Trends

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ ISSN/ ISBN**	**Name of the Publisher**
Shibu K. V	Introduction to Embedded Systems	2nd Edition	Mc Graw Hill
Raj Kamal	Embedded Systems	4th Edition	TMH
Reference Books:			
Frank Vahid	Embedded System Design	1st Edition	John Wiley
Lyla B Das	Embedded Systems	1st Edition	Pearson
David E. Simon	An Embedded Software Primer	1st Edition	Pearson Education

Course Name: Operation Research & Optimisation Techniques

Course Code: MCA403 Credit: 3 Lecture Hours: 40

Name of the Course: Operation Research & Optimisation Techniques	
Course Code: MCA403	**Semester: 4th**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 3	
Aim:	
1	To introduce the fundamentals of Operations Research (OR) and optimization techniques for decision-making.
2	To develop the ability to formulate, analyze, and solve optimization problems in computing and business applications.
3	To apply mathematical modeling and computational techniques for real-world problem-solving.
Objective:	
1	To understand linear programming, transportation, and assignment models in OR.
2	To apply optimization techniques such as dynamic programming and network flow analysis.
3	To analyze game theory, queuing models, and inventory control techniques for decision-making.
4	To implement computational algorithms for optimization problems using programming tools.
Pre-Requisite:	
1	Basic knowledge of mathematics, probability, and programming logic.
Course Outcome:	
1	Explain the concepts, models, and techniques of Operations Research.
2	Formulate and solve optimization problems using mathematical models.
3	Apply game theory, queuing models, and inventory control for real-world scenarios.
4	Use computational tools and algorithms for solving OR and optimization problems.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	2	1	1	1	0	0	0	1	1	2	2	1	1
CO2	3	3	3	2	2	1	0	0	0	1	2	2	3	2	2
CO3	2	3	2	2	1	2	1	0	0	1	2	2	2	2	2
CO4	3	3	3	2	3	1	0	0	0	1	2	3	3	3	3

Module number	Topic	Sub-topics
1	**Linear and Integer Programming**	Linear Programming-Simplex Method, Duality Method, Assignment Problem, Transportation Problem
2	**Network Optimization and Project Scheduling**	Integer Programming-Cutting Plane, Branch & Bound Network Optimization Models- The shortest path problem, Minimum Spanning Tree Algorithm, Maximal Flow Algorithms, PERT/ CPM.
3	**Dynamic Programming and Queuing Theory**	Dynamic Programming- Characteristics, Deterministic & Probabilistic Dynamic Programming. Queuing Theory- Basic Structure, Exponential distribution, Birth-and-Death Model, M/M/I Queue.
4	**Game Theory and Sequencing**	Game Theory-Two person Zero Sum game, saddle point determination, algebraic method, graphical method etc.
5	**Inventory Control Models**	Inventory Control- Determination of EOQ, Components, Deterministic Continuous & Deterministic Periodic Review Models, Stochastic Continuous & Stochastic Periodic Review Models. Sequencing- Two men two machines, Three Men Two Machines

List of Books Text Books:

1. Operation Research, Kanti Swaroop
2. Operation Research,V.K. Kapoor
3. Operation Research,Paneer Selvam,PHI
4. Operations Research, Hillier & Lieberman, TMH
5. Operations Research, Kalavati,VIKAS
6. Operation Research,Humdy Taha,PHI
7. Statistics,Random Process & Queuing Theory, Prabha, Scitech
8. Operations Research, Vijayakumar, Scitech
9. Quantitative Techniques,Vol.1 & II ,L.C. Jhamb,EPH

Course Name: Management & Accounting Credit: 2

Course Code: MCA405 Lecture Hours: 40

Name of the Course: Management & Accounting	
Course Code: MCA405	**Semester: 4th**
Duration: 40 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 3	End Semester Exam: 70
Tutorial: 1	Continuous Assessment: 30
Credit: 2	
Aim:	
1	To gain Knowledge of basic aspects of Management
2	To enhance Ability to identify qualities of a good Management Control and Strategy
3	To implement learned Concept of Financial and Cost Accounting to solve problems
Objective:	
1	The fundamental in basic in Management
2	Basic concepts in the Management control and strategy
3	Principles of Financial Accounting
4	Significance of Cost Accounting in the Accounting field
Pre-Requisite:	
1	Proficiency in Basic of Management and Accounting
Course Outcome:	
1	On completion of this course students are expected to learn various Concept of Planning, scheduling, organizing, staffing, directing, controlling Managerial economics
2	On completion of this course students are expected to design Management Control system.
3	On completion of this course students are expected to do a comparative analysis among different Financial statement and Financial accounting used in a given scenario.
4	On completion of this course students are expected to acquire adequate knowledge and skills to solve a real-life Cost Volume Profit analysis and budgeting

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	1	2	0	0	2	1	1	0	1	3	2	2	0	0
CO2	2	3	3	1	1	2	0	1	1	2	3	2	3	1	1
CO3	2	3	3	1	1	2	0	1	0	2	3	2	3	2	1
CO4	3	3	3	1	1	2	0	1	0	1	3	3	3	2	1

Module number	Topic	Sub-topics
1	**Basics of management**	Planning, scheduling, organizing, staffing, directing, controlling Managerial economics and financial management, productivity management Human resource development and management, selection, training and role of IT
2	**Management Control Systems**	Introduction to management control systems: goals, strategies; Performance measures
3	**Strategy**	Firm and its environment, strategies and resources, industry structure and analysis, corporate strategies and its evaluation, strategies for growth and diversification, strategic planning
4	**Financial Accounting**	Financial statements and analysis Conceptual framework of cost accounting. Financial accounting computer packages.
5	**Cost Accounting**	Cost-volume profit (CVP) relationship, budgeting, cost accumulation system, variable and absorption costing system

List of Books Text Books:			
Name of Author	**Title of the Book**	**Edition/ ISSN/ ISBN**	**Name of the Publisher**
Khan & Jain	Management Accounting	8th Edition	Mc Graw Hill
Harold Koontz	Essentials of Management	11th Edition	Mc Graw Hill
Reference Books:			
Ramchandran	Accounting for Management (Management Accounting)	2nd Edition	Scitech Publications

Course Name: General Studies & Current Affairs - IV Credit: 0.5

Course Code: MCA(GS)401 Lecture Hours: 20

Name of the Course: General Studies & Current Affairs - IV	
Course Code: MCA(GS)401	**Semester: 4th**
Duration: 48 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 2	End Semester Exam: 100
Tutorial: 0	Continuous Assessment: 100
Credit: 0.5	
Aim:	
Sl. No.	
1	To enhance quantitative aptitude and logical reasoning for effective problem-solving in professional environments.
2	To develop oral, listening, and reading communication skills for workplace interactions.
3	To equip students with practical communication skills, including professional correspondence and presentations.
Objective:	
Sl. No.	
1	To strengthen quantitative and logical reasoning for analytical decision-making.
2	To improve verbal and non-verbal communication skills, including articulation and clarity.
3	To enhance listening and reading comprehension for better information processing.
4	To develop practical communication techniques, such as presentations, business communication, and public speaking.
Pre-Requisite:	
Sl. No.	
1.	Basic understanding of English language and fundamental mathematical concepts.

Module number	Topic	Sub-topics
1	**History**	Socio cultural Changes: Introduction of western Education, Ram Mohan Roy and BramhoSamaj, Young Bengal movevemnt, Arya samaj, Ramkrishna Mission, Aligarh movement, Vidyasagar Revolt of 1857: Cause, Character, cause of failure, impact Partition of Bengal: Cause, Swadeshi and Boycott, Newspaper Indian National congress
2	**Geography**	1. Natural vegetation of India 2. Minerals and multipurpose river projects of India 3. Agriculture of India • Types of Agriculture (Intensive subsistence, Extensive subsistence, Mixed farming, Jhoom cultivation) • Types of crops (Rice, Wheat, Sugarcane, Pulses, Cotton, Jute, Tobacco)
3	**Macro Economics**	1. Indian Planning & NITI Aayog 2. Indian Foreign trade and International organizations 3. Balance of Payment and Balance of Trade.
4	**Constitution**	1. Election Commission- Related Articles, Power & Function & Provision of Election 2. Emergency Provisions- Related Articles, Conditions Application, Supreme power during emergency. 3. National Commission for SC/ST/OBC, Function of the commissions, Special offer & related articles for SC/ST/OBC 4. Different amendments of Indian Constitution & the related articles 5. Formation UPSC, Related Articles, Scope & Power, Duties of CAG, Formation SPSC, Related Articles, Scope & Power.

References

1. History:

India's Ancient Past (Ancient History) : R.S. Sharma

History of medieval India (Medieval History): Satish Chandra History of Modern India (Modern History): Bipin Chandra

India's struggle for Independence (Modern History): Bipin Chandra

Geography:

India- Khullar Economics:

Indian Economy- TATA Mc Graw Hill/Ramesh Singh Indian Economy – Arihant

Constitution:

Indian Constitution- D.D. Basu

Name of the Course: **Competitive Aptitude Training – IV**	
Course Code: MCA(GS)481	**Semester: 4th**
Duration: 20 Hrs.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Theory: 2	End Semester Exam: 100
Tutorial: 0	Continuous Assessment: 100
Credit: 0.5	

Aim:	
Sl. No.	
1	To develop verbal reasoning, analytical thinking, and problem-solving skills essential for competitive exams and job recruitment.
2	To enhance English language proficiency, focusing on grammar, comprehension, and professional communication.
3	To build expertise in data interpretation, improving the ability to analyze and solve quantitative problems efficiently.

Objective:	
Sl. No.	
1	To strengthen English grammar, vocabulary, and comprehension skills for competitive aptitude tests.
2	To develop logical and analytical reasoning through structured problem-solving techniques.
3	To enhance proficiency in data interpretation, including tabular and graphical analysis.
4	To improve formal communication skills, such as official letter writing, for professional settings.

Pre-Requisite:	
Sl. No.	
1.	Basic understanding of English grammar and elementary mathematical reasoning.

Module number	Topic	Sub-topics
1	**Quantitative Aptitude**	1 Permutation & Combination. 2 Probability- basic concepts of probability , different theorems & applications, binomial, poison & normal Distributions. 3 Geometry- Concept of different shapes like triangle, quadrilateral, rectangle, square, circle etc. different theorems & their applications. 4 Mensuration- Formulae on triangles, square, Rhombus, parallelogram, sphere, circle, cone, pyramid etc, Application based problem solving. Coordinate Geometry- Locus, Straight lines, Circle etc
2	**Objective English**	1. Miscellaneous Corrections on Tense part 1. 2. Miscellaneous Corrections on Tense part 2. 3. Fill in the blanks (Single Blank) 4. Miscellaneous Vocabulary
3	**Soft Skills**	1 Communication Development. 2 Personality Development.
4	**Computer proficiency**	1. C programming, Basics of C++

List of Books Text Books:

1. Fastrack objective Arithmetic: Arihant
2. Quantitative aptitude for Competitive exam (4th Edition): TATAMc Graw Hill
3. Quantitative aptitude for Competitive exam (3rd Edition): PEARSON

Course Name: Major Project Credit: 15

Course Code: MCA491

Name of the Course: Major Project	
Course Code: MCA491	**Semester: 4th**
Duration: 12 Weeks.	**Maximum Marks: 100**
Teaching Scheme	**Examination Scheme**
Practical: 0	Practical Sessional Internal continuous evaluation: 100
Credit: 15	Practical Sessional external examination: 100

Aim:	
Sl. No.	
1	To enable students to apply theoretical knowledge to real-world problems through project development.
2	To enhance problem-solving, software development, and research skills.
3	To develop teamwork, project management, and documentation abilities.

Objective:	
Sl. No.	
1	To identify and define a real-world computing problem.
2	To design and implement a software solution using appropriate tools and technologies.
3	To analyze and evaluate the efficiency and effectiveness of the developed system.
4	To document and present the project findings professionally.

Pre-Requisite:	
Sl. No.	
1.	Knowledge of programming, databases, and software development lifecycle (SDLC).

Course Outcome:	
1.	Identify and define a problem statement relevant to computing applications.
2.	Develop a functional prototype or software solution using modern tools.
3.	Demonstrate analytical and technical skills in project execution.
4.	Present a well-documented project report with findings and future scope.

CO-PO-PSO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	2	2	1	0	1	1	1	2	2	3	2	2
CO2	3	3	3	2	3	1	0	1	1	1	2	3	3	2	3
CO3	3	3	3	2	3	1	0	1	1	2	2	3	3	2	3
CO4	2	2	2	1	2	1	0	1	1	3	2	2	2	1	2

-- END --