



INSTITUTE OF ENGINEERING & MANAGEMENT, KOLKATA



(A SCHOOL OF THE UNIVERSITY OF ENGINEERING AND MANAGEMENT, KOLKATA)

DETAILED SYLLABUS

1ST Year – MCA – 2026-2028 BATCH

DEPARTMENT OF COMPUTER APPLICATIONS
MCA STRUCTURED SYLLABUS - FIRST YEAR AND FIRST SEMESTER OF MCA2026 ADMISSION BATCH

Course Code	Course Title	Total No. of Weekly Contact Hours				Total No. of Credits
		Lecture (L)	Tutorial (T)	Practical (P)	Total Lectures	
1st Semester (Theory)						
MCA101	Computer Organisation and Architecture	4	1	0	5	3
MCA102	Computer Programming with C	4	1	0	5	4
MCA103	Data Structures with C	4	1	0	5	4
MCA105	Business English and Communication	3	1	0	4	3
MCA106	Computational Mathematics and Statistics	4	1	0	5	4
MCAGS101	Mental Maths for Professionals - I	2	1	0	3	1
Total of Theory					27	19
1st Semester (Practical)						
MCA192	C Programming Laboratory	0	0	4	4	3
MCA193	Data Structures Laboratory using C	0	0	3	3	3
Total of Practical					7	6
1st Semester (Sessional)						
MCA171	Research Methodology and IPR	2	1	0	3	2
MCA172	Python Programming Fundamentals	0	0	3	3	2
IFC	Industry and Foreign Certification	0	0	0	0	0
MAR	Mandatory Additional Requirements	0	0	0	0	0
MOOCs	Massive Open Online Course	0	0	0	0	0
Total of Sessional					6	4
Total of the Semester					40	29

DEPARTMENT OF COMPUTER APPLICATIONS
MCA STRUCTURED SYLLABUS - FIRST YEAR AND SECOND SEMESTER OF MCA2026 ADMISSION BATCH

Course Code	Course Title	Total No. of Contact Hours				Total No. of Credits
		Lecture (L)	Tutorial (T)	Practical (P)	Total Hours	
2 nd 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	Advanced-Data Structures and Algorithms with Python	3	1	0	4	4
MCA205	Artificial Intelligence and Machine Learning	3	1	0	4	4
MCA(GS)201	General Studies & Current Affairs-II	3	0	0	3	1
Total of Theory					23	20
2 nd Semester (Practical)						
MCA291	Database Management System Laboratory	0	0	3	3	3
MCA292	Object-Oriented Programming with Java Laboratory	0	0	3	3	3
MCA294	Advanced-Data Structures with Python Laboratory	0	0	3	3	3
MCA295	Artificial Intelligence and Machine Learning Laboratory	0	0	3	3	3
Total of Practical					12	12
2 nd Semester (Sessional)						
IVC272	Economics, Finance and Entrepreneurship Skills - Intermediate	1	0	0	2	0
	PPT - TECHNICAL				3	
IFC	Industry and Foreign Certification	0	0	0	0	0
MAR	Mandatory Additional Requirements	0	0	0	0	0
MOOCs	Massive Open Online Course	0	0	0	0	0
Total of Sessional					5	0
Total of the Semester					40	32

DEPARTMENT OF COMPUTER APPLICATIONS
MCA STRUCTURED SYLLABUS - SECOND YEAR AND FIRST SEMESTER OF MCA2026 ADMISSION BATCH

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	4	1	0	5	3
MCA302	Software Engineering & TQM	4	1	0	5	4
MCA303	Generative and Agentic AI	4	1	0	5	3
MCA304A/B/C/D	Professional Elective - I	4	1	0	5	3
MCA305A/B/C/D	Open Elective - I	2	0	0	2	2
MCAGS301	General Studies & Current Affairs - IV	3	0	0	3	1
Total of Theory					25	16
3rd Semester (Practical)						
MCA392	Software Project Management Laboratory	0	0	2	2	3
MCA393	Generative and Agentic AI Laboratory	0	0	2	2	3
Total of Practical					4	6
3rd Semester (Sessional)						
MCA371	Sustainability, Climate Action and Environmental Sciences	3	0	0	3	2
MCA381	Industrial Training	0	0	0	0	2
MCA382	Minor Project	0	0	8	8	6
MCA383	Seminar	0	0	0	0	1
IFC	Industry and Foreign Certification	0	0	0	0	0
MAR	Mandatory Additional Requirements	0	0	0	0	0
MOOCS	Massive Open Online Courses	0	0	0	0	0
Total of Sessional					11	11
Total of the Semester					40	33

Professional Elective - I		Open Elective - I	
Course Code	Topic	Course Code	Topic
MCA304A	Distributed Database Management	MCA305A	Organizational Behavior and Human Resources
MCA304B	Image Processing	MCA305B	Financial Accounting and Management
MCA304C	Cyber Security	MCA305C	E-Commerce and Digital Marketing
MCA304D	Cloud Computing	MCA305D	Supply Chain Management

DEPARTMENT OF COMPUTER APPLICATIONS
MCA STRUCTURED SYLLABUS - SECOND YEAR AND SECOND SEMESTER OF MCA2026 ADMISSION BATCH

Course Code	Course Title	Total No. of Contact Hours				Total No. of Credits
		Lecture (L)	Tutorial (T)	Practical (P)	Total Hours	
4 th Semester (Theory)						
MCA401	Values and Ethics	2	0	0	3	2
MCA402	Professional Elective - II	2	0	0	3	3
MCA(GS)401	General Studies & Current Affairs - IV	2	0	0	2	1
Total of Theory					8	6
4 th Semester (Practical)						
MCA491	Major Project	0	0	32	32	15
Total of Practical					32	15
4 th Semester (Sessional)						
IFC	Industry and Foreign Certification	0	0	0	0	0
MAR	Mandatory Additional Requirements	0	0	0	0	0
MOOCS	Massive Open Online Courses	0	0	0	0	0
Total of Sessional					0	0
Total of the Semester					40	21
Total of Program CREDIT					115	

Professional Elective - II	
Course Code	Topic
MCA402A	Compiler Design
MCA402B	Mobile Computing
MCA402C	Embedded Systems
MCA402D	Natural Language Processing



Institute of Engineering & Management, Kolkata
Newtown
(School of University of Engineering and Management, Kolkata)



Subject Name: Computer Organisation and Architecture

Credit: 4

Subject Code: MCA101

Lecture Hours: 40

Course Name: Computer Organisation and Architecture	
Course Code: MCA101	Semester: 1st
Duration: 40 Hrs.	Maximum Marks: 100
Teaching Scheme	Examination Scheme
Theory: 4	End Semester Exam: 100
Tutorial: 1	Continuous Assessment: 100
Credit: 4	

Aim:	
Sl. No.	
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:	
Sl. No.	
1	Understanding Logic gates, flip flops and counter.
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:	
Sl. No.	
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	Mapping with Industry and International Academia	Lecture Hours
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.	International Academia: https://web.stanford.edu/dept/registrar/bulletin_past/bulletin02-03/pdf/CompSci.pdf AICTE-prescribed syllabus: https://www.aicte-india.org/downloads/mcadegree.pdf Industry Mapping: The concepts delivered are in sync with the industry standards	4
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	International Academia: https://web.stanford.edu/dept/registrar/bulletin_past/bulletin02-03/pdf/CompSci.pdf AICTE-prescribed syllabus: https://www.aicte-india.org/downloads/mcadegree.pdf Industry Mapping: The concepts delivered are in sync with the industry standards	8
3	Register Transfer, Micro-Operations and Micro-Programmed Control	Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Micro-Operations, Logic Micro-Operations, Shift Micro-Operations, Arithmetic logic shift unit, Control Memory, Address Sequencing, Micro-Program example, Design of Control Unit.	International Academia: https://web.stanford.edu/class/cs97si/03-data-structures.pdf AICTE-prescribed syllabus: https://www.aicte-india.org/downloads/mcadegree.pdf Industry Mapping: The concepts delivered are in sync with the industry standards	8
4	Memory System:	Memory Hierarchy, Semiconductor Memories, RAM (Random Access Memory), Read Only Memory (ROM), Types of ROM, Cache	International Academia: https://web.stanford.edu/dept/registrar/bulletin_past/bulletin02-03/pdf/CompSci.pdf	7

		Memory, Performance considerations, Virtual memory, Paging, Secondary Storage, RAID.	AICTE-prescribed syllabus: https://www.aicte-india.org/downloads/mcadegree.pdf Industry Mapping: The concepts delivered are in sync with the industry standards	
5	Input-Output:	I/O interface, Programmed IO, Memory Mapped IO, Interrupt Driven IO, DMA.	International Academia: https://web.stanford.edu/dept/registrar/bulletin_past/bulletin02-03/pdf/CompSci.pdf AICTE-prescribed syllabus: https://www.aicte-india.org/downloads/mcadegree.pdf Industry Mapping: The concepts delivered are in sync with the industry standards	7
6	MULTI-PROCESSORS	Characteristics of multiprocessors, Interconnection structures, Inter Processor Arbitration, Interprocessor Communication and Synchronization, and Cache Coherence.	International Academia: https://web.stanford.edu/dept/registrar/bulletin_past/bulletin02-03/pdf/CompSci.pdf AICTE-prescribed syllabus: https://www.aicte-india.org/downloads/mcadegree.pdf Industry Mapping: The concepts delivered are in sync with the industry standards	6

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	3 rd Ed	Pearson/PHI
Reference Books: 1. Carl Hamacher, Zvonks Vranesic, Safea Zaky (2002), Computer Organization, 5th edition, McGraw Hill, New Delhi, India.			



Institute of Engineering & Management, Kolkata
Newtown
(School of University of Engineering and Management, Kolkata)



Subject Name: Computer Programming with C

Credit: 4

Subject Code: MCA102

Lecture Hours: 48

Course Name: Computer Programming with C & C Programming Laboratory	
Course Code: MCA102 & MCA192	Semester: 1st
Duration: 40 Hrs.	Maximum Marks: 100
Teaching Scheme	Examination Scheme
Theory: 4	End Semester Exam: 100
Tutorial: 1	Continuous Assessment: 100
Practical: 4	Practical Sessional Internal continuous evaluation: 100
Credit: 4+3	Practical Sessional external examination: 100
Aim:	
Sl. No.	
1	To develop a rigorous understanding of algorithmic problem-solving and translate abstract logic into structured C programs.
2	To cultivate the ability to write modular, efficient, and memory-safe code suited for postgraduate-level software development.
3	To establish a strong foundation in core programming constructs, memory management, and data handling as a stepping stone for advanced systems programming.

Objective:	
Sl. No.	
1	A foundational understanding of computer operations, mathematical logic, and an aptitude for computational problem-solving.
2	To impart an in-depth understanding of memory management, specifically through pointer arithmetic and dynamic memory allocation.
3	To demonstrate the practical implementation of user-defined data types (structures and unions) and secure string manipulations.
4	To foster critical computational thinking necessary for optimising code, debugging errors, and implementing file handling for data persistence.
Pre-Requisite:	
Sl. No.	
1.	Proficiency in one high-level programming language
Course Outcome:	
Upon successful completion of this course, students will be able to:	
C01	Apply algorithmic thinking and fundamental C programming constructs (control structures, operators) to translate mathematical and logical problems into executable code.
C02	Analyse computational problems to select and implement appropriate linear data structures (arrays, strings) for efficient data organisation and manipulation.
C03	Evaluate memory management and program modularity by leveraging pointers, dynamic memory allocation, and user-defined functions to optimise software performance.
C04	Design and develop comprehensive software solutions utilizing heterogeneous data types (structures/unions) and persistent file I/O operations for real-world data processing scenarios.

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	2	1	-	1	1	1	1	2	3	1	2
CO2	2	3	3	2	2	1	-	1	1	1	1	2	3	2	2
CO3	2	2	3	2	3	1	1	1	1	1	2	2	3	2	2
CO4	2	2	3	2	3	2	1	1	2	1	2	2	3	3	2

Module Number	Topic	Sub-topics	Mapping with Industry and International Academia	Lecture Hours	Corresponding Lab Assignment
1	Fundamentals & C Basics	Algorithmic Thinking & Compilation Process Tokens, Data Types, and Memory Footprints Formatted & Unformatted I/O Operators, Expressions, and Type Casting	International Academia: Massachusetts Institute of Technology (MIT): 6.087 (Practical Programming in C) Harvard University: CS50 (Introduction to Computer Science) AICTE-prescribed syllabus (3 Years): AICTE-prescribed syllabus (2 Years): Industry Mapping: The concepts delivered are in sync with the industry standard	6	Week 1: Fundamentals, I/O, and Operators

2	Control Structures	if, else, and switch statements for, while, and do-while loops Nested loops and optimization Jump statements (break, continue, goto)	International Academia: <u>Massachusetts Institute of Technology (MIT): 6.087 (Practical Programming in C)</u> <u>Harvard University: CS50 (Introduction to Computer Science)</u> AICTE-prescribed syllabus (3 Years): AICTE-prescribed syllabus (2 Years): Industry Mapping: The concepts delivered are in sync with the industry standards	6	Week 2: Control Structures
3	Functions & Storage Classes	Modular design, Declaration, and Call Stack Parameter passing and Return values Recursion mechanics and examples Storage Classes and Scope rules	International Academia: <u>Massachusetts Institute of Technology (MIT): 6.087 (Practical Programming in C)</u> <u>Harvard University: CS50 (Introduction to Computer Science)</u> AICTE-prescribed syllabus (3 Years): AICTE-prescribed syllabus (2 Years): Industry Mapping: The concepts delivered are in sync with the industry standards	7	Week 3: Functions & Recursion
4	Arrays and Strings	1D Arrays and passing arrays to functions 2D Arrays and Matrix operations String basics and <string.h> library Secure coding (preventing buffer overflows)	International Academia: <u>Massachusetts Institute of Technology (MIT): 6.087 (Practical Programming in C)</u> <u>Harvard University: CS50 (Introduction to Computer Science)</u> AICTE-prescribed syllabus (3 Years): AICTE-prescribed syllabus (2 Years): Industry Mapping: The concepts delivered are in sync with the industry standards	7	Week 4: 1-D and 2-D Arrays Week 5: Strings & Secure Coding
5	Pointers & Dynamic Memory Allocation	Pointer fundamentals and Address mechanics Pointer arithmetic and advanced types (void, null)	International Academia: <u>Massachusetts Institute of Technology (MIT): 6.087 (Practical Programming in C)</u>	9	Week 6: Pointers Fundamentals

		Pointers with Arrays and Strings Pass by Reference Dynamic Memory (malloc, calloc, free, leaks)	• <u>Harvard University: CS50 (Introduction to Computer Science)</u> <u>AICTE-prescribed syllabus (3 Years):</u> <u>AICTE-prescribed syllabus (2 Years):</u> <i>Industry Mapping:</i> The concepts delivered are in sync with the industry standards		Week 7: Dynamic Memory Allocation (DMA)
6	User-Defined Data Types	Structures (Basics, Array of Structures, Nested) Passing Structures and the -> operator Memory alignment and padding Unions, enum, and typedef	<i>International Academia:</i> • <u>Massachusetts Institute of Technology (MIT): 6.087 (Practical Programming in C)</u> • <u>Harvard University: CS50 (Introduction to Computer Science)</u> <u>AICTE-prescribed syllabus (3 Years):</u> <u>AICTE-prescribed syllabus (2 Years):</u> <i>Industry Mapping:</i> The concepts delivered are in sync with the industry standards	5	Week 8: User-Defined Data Types
7	File Handling & Advanced Concepts	File operations (Open, Read, Write, Close modes) Random Access (fseek, ftell, rewind) Command Line Arguments Preprocessor Directives	<i>International Academia:</i> • <u>Massachusetts Institute of Technology (MIT): 6.087 (Practical Programming in C)</u> • <u>Harvard University: CS50 (Introduction to Computer Science)</u> <u>AICTE-prescribed syllabus (3 Years):</u> <u>AICTE-prescribed syllabus (2 Years):</u> <i>Industry Mapping:</i> The concepts delivered are in sync with the industry standards	5	Week 9: File Handling & Command Line Arguments Weeks 10 & 11: Capstone Projects

List of Books Textbooks:			
Name of Author	Title of the Book	Edition/ISSN/ISBN	Name of the Publisher
E. Balagurusamy	Programming in ANSI C	978-9389949186	McGraw-Hill Education
Yashavant Kanetkar	Let us C	978-9365899689	BPB Publications
K R Venugopal; Sudeep R Prasad	Mastering C	978-9332901278	McGraw-Hill Education
Reference Books:			
Dennis Ritchie; Brian W. Kernighan	C PROGRAMMING LANGUAGE, 2ND EDN	978-9332549449	Pearson Education India
K. N. King	C Programming, A Modern Approach, 2 nd Edition	978-0-393-97950-3	W. W. Norton & Company

Annexure – I

Course Code: MCA102

Course Name: C Programming Lab

Week 1: Fundamentals, I/O, and Operators

Direct Questions:

1. Write a program using the `sizeof()` operator to print the memory footprint of `char`, `int`, `float`, and `double` on your current compiler architecture.
2. Write a program to swap the values of two variables without using a third temporary variable.
3. Write a program that takes an integer input from the user and uses bitwise operators to determine if the number is odd or even.

Story-Based Question: A new music streaming startup needs a basic bandwidth calculator to estimate server load. Write a program that asks the user for the number of active listeners, the average audio bitrate (in kbps), and the average song duration (in minutes). Calculate and display the total data transferred in Megabytes (MB).

Week 2: Control Structures

Direct Questions:

1. Write a menu-driven program using a `switch-case` statement to simulate a basic calculator. Handle the divide-by-zero edge case.
2. Write a program to print a right-angled pyramid pattern of numbers.
3. Write a program to check if a user-inputted number is an Armstrong number.

Story-Based Question: A local research team is analyzing epidemiological data to track a recent outbreak. Write a program that takes a patient's data as input: Age, Body Temperature, and Symptom Duration. Use an `else-if` ladder to classify their risk level as "Low," "Moderate," or "High" based on threshold conditions you define.

Week 3: Functions & Recursion

Direct Questions:

1. Write a function `isPrime(int n)` that returns 1 if prime and 0 otherwise. Print all prime numbers between 1 and 100.
2. Write a recursive function to calculate the Factorial of a given number.
3. Write a user-defined function `power(int base, int exp)` that calculates base exponentiation without using `<math.h>`.

Story-Based Question: For an upcoming university CyberX workshop, participants need a secure, dynamically generated passcode. Write a recursive function that takes a 4-digit ID and recursively sums the digits until a single digit is obtained. Multiply that digit by a security constant to generate the passcode.

Week 4: 1-D and 2-D Arrays

Direct Questions:

1. Write a program to find the second largest and second smallest elements in a 1D array of integers.
2. Write a program to perform Matrix Multiplication on two 3x3 matrices.
3. Write a program to reverse the elements of a 1D array *in-place*.

Story-Based Question: You are organizing an online Capture the Flag (CTF) programming competition. You have two separate, sequentially sorted arrays of competitor scores. Write a program to seamlessly interleave and merge these two arrays into a single sorted array to generate the final leaderboard.

Week 5: Strings & Secure Coding

Direct Questions:

1. Write a program to find the length of a string and reverse it without using standard library functions.
2. Write a program that takes a string input and counts the total number of vowels, consonants, digits, and whitespaces.
3. Write a program to check if two strings inputted by the user are anagrams.

Story-Based Question: You are formatting an academic research paper on XAI, and the publisher requires specific proprietary frameworks to be redacted. Write a program that reads a text using `fgets()`, searches for a target word (like "SHAP" or "LIME"), and replaces all its occurrences with asterisks (****).

Week 6: Pointers Fundamentals

Direct Questions:

1. Write a program to swap two variables using pass-by-reference.
2. Traverse a 1D integer array and print its elements using *only* pointer arithmetic.
3. Write a function that returns both the maximum and minimum values of an array via pointer parameters.

Story-Based Question: A smart wearable device tracks `heart_rate` and `step_count`. Write a function taking these variables by reference. If the `heart_rate` exceeds 150 BPM, reset the `step_count` to zero and double the `heart_rate` to trigger a hardware alert state.

Week 7: Dynamic Memory Allocation (DMA)

Direct Questions:

1. Use `malloc()` to allocate memory for an array of N integers. Populate, calculate the sum, and `free()` the memory.
2. Demonstrate `calloc()` by allocating memory for an array of floats and verifying initialization to zero.
3. Use `realloc()` to expand a dynamically allocated array of 5 integers to hold 10 integers.

Story-Based Question: An epidemiological prognosis system needs to load daily patient records. Write a program that asks for the current day's patient count, dynamically allocates the memory to store their ages, calculates the moving average age, and strictly ensures the memory is freed before termination.

Week 8: User-Defined Data Types

Direct Questions:

1. Define a `struct Employee (emp_id, name, salary)`. Read and print data for one employee.
2. Create an array of `struct Student`. Read data for 5 students and print the details of the one with the highest marks.
3. Write a program demonstrating a `union`, showing how memory is shared and overwritten.

Story-Based Question: An online music learning platform manages its catalog using structures. Define a `Course` structure containing an inner `Pricing` structure. Create an array of 3 courses, calculate the final price after discounts for each, and display the course with the lowest final cost.

Week 9: File Handling & Command Line Arguments

Direct Questions:

1. Write a program using command-line arguments to copy the contents of one file into another.
2. Write a program to open a text file and count the number of lines, words, and characters.
3. Accept an integer via command-line and append its multiplication table to `table.txt`.

Story-Based Question: A cybersecurity digital footprint analyzer monitors network requests. Write a program that reads a mock log file. Extract any line containing the keyword "UNAUTHORIZED" and append it to a new file called `threat_report.txt`.

Weeks 10 & 11: Capstone Projects

Students must select **ONE** of the following 12 capstone projects to complete. These projects require the integration of multiple core concepts including Pointers, Structures, Dynamic Memory, and File Handling.

Project 1: The Epidemiological Prognosis Tracker

An epidemiological research lab needs a system to process daily patient metrics. Write a C program that reads a dataset of patients (Patient ID, Age, Body Temperature, Symptom Duration) from a file. Using structures and pointer manipulation, apply a rules-based threshold to identify high-risk cases. The program must automatically extract these flagged records and generate a formatted "Prognosis_Report.txt" for the clinical team.

Project 2: Music Startup Ecosystem Manager

A new startup focusing on musical education and talent discovery needs a backend inventory engine. The platform manages three distinct entities: "Learning Courses", "Instrument Access", and "Talent Profiles". Using arrays of structures and dynamic memory allocation (`malloc/realloc`), build a menu-driven program that allows administrators to add new courses, update instrument availability, and display a sorted list of registered musical talents.

Project 3: CyberX Digital Footprint Analyzer

Following the successful conclusion of a CyberX cybersecurity workshop, the server generated a massive log file of network access requests. Write a command-line application that parses this "digital footprint" text file line by line. Using advanced string manipulation, the program must extract unique IP addresses, count their frequency, and quarantine any log entry containing the word "UNAUTHORIZED" into a separate secure text file.

Project 4: Capture The Flag (CTF) Auto-Scorer

You are tasked with building the backend scoring engine for an upcoming university Hackerrank-style CTF competition. Competitor data (Team Name, Challenges Solved, Time Penalties) is continuously updated. Use pointers to structures to manage the team data. Write a dynamic sorting algorithm that instantly updates the leaderboard based on net scores and displays the top 3 teams in real-time on the console.

Project 5: Explainable AI (XAI) Feature Importance Matrix

In machine learning, algorithms like SHAP and LIME output feature importance scores. Write a C program that simulates this by generating a 2D matrix of floating-point numbers representing "feature weights" for a dataset. Calculate the row-wise and column-wise averages. Apply a user-defined threshold to identify the most critical data features, extracting their indices into a 1D dynamically allocated array using pointers.

Project 6: University Event Poster Layout Generator

The Department of Computer Applications frequently hosts workshops and competitions. Write a text-based layout generator that takes user input for an event (Title, Date, Time, Venue, Speaker Name). Using string formatting, loop constructs, and standard I/O padding, the program must generate a visually appealing ASCII-art "poster" layout and save it as a text file to be distributed to the MCA students.

Project 7: Supermarket Inventory & Billing System

Build a robust point-of-sale system for a local supermarket. Define a structure for `Product` (ID, Name, Price, Stock). The program should allow a cashier to input purchased item IDs and quantities. It must check stock availability, calculate the total bill including a 5% GST, update the remaining inventory using pass-by-reference, and print an itemized receipt to the console.

Project 8: Advanced Cryptographic Secure Messaging

Design a two-way secure messaging utility. The program takes a string message from the user and a numeric encryption key. Implement a custom encryption algorithm that applies a Caesar Cipher shift followed by a bitwise XOR operation on every character. The program must be able to output the encrypted ciphertext, and conversely, take ciphertext and the key to decrypt it back to the original message.

Project 9: The Hospital Blood Bank Matcher

A regional hospital requires a fast blood-matching registry. Store donor records (Name, Blood Group, Age, Contact) in a binary file using `fwrite()`. When an emergency trauma request comes in, the user inputs the required blood type and units needed. The program must rapidly search the binary file using `fread()`, identify eligible donors, and output their contact information, ensuring all file pointers are securely closed.

Project 10: Dynamic Library Book Allocator

A university library's new acquisitions arrive in unpredictable batch sizes. Write a program that initially allocates memory for an array of 5 `Book` structures (Title, Author, ISBN). As the user inputs new books, if the array becomes full, use `realloc()` to dynamically double the array size without losing previous entries. Implement a function to safely `free()` the entire catalog when the program terminates.

Project 11: Railway Route & Seat Manager

Design a train ticketing system for a specific route. Define a structure for passenger tickets. Manage a 1D array representing 50 available seats. The menu-driven application must allow users to view available seats, book a specific seat number, cancel a ticket, and display the final passenger manifest. Prevent double-booking using logical condition checks.

Project 12: MCA Gradebook & SGPA Calculator

Create a comprehensive gradebook application for the 1st-semester MCA cohort. The program reads a file containing student Roll Numbers and marks in 5 subjects. Using user-defined functions, calculate the total marks, percentage, and SGPA for each student based on the university's grading scale. The application must generate individual text-file "report cards" named dynamically (e.g., "Report_Roll101.txt").



University of Engineering and Management

Institute of Engineering & Management, New Town Campus



Subject Name: Data Structure with C

Credit: 4

Code: MCA103

Lecture Hours: 40

Name of the Course: Data Structure with C & Data Structures Laboratory using C	
Course Code: MCA103 & MCA193	Semester: 1st
Duration: 40 Hrs.	Maximum Marks: 100
Teaching Scheme	Examination Scheme
Theory: 4	End Semester Exam: 100
Tutorial: 1	Continuous Assessment: 100
Practical: 3	Practical Sessional Internal continuous evaluation: 100
Credit: 4+3	Practical Sessional external examination: 100

Aim:	
Sl. No.	
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:	
Sl. No.	
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:	
Sl. No.	
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 and limitations.
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 structures and 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 life software 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	Mapping with Industry and International Academia	Lecture Hours	Corresponding Lab Assignment
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	International Academia: https://web.stanford.edu/class/cs97si/03-data-structures.pdf AICTE-prescribed syllabus: https://makautexam.net/aicte_details/Syllabus/MCA/sem221.pdf Industry Mapping: The concepts delivered are in sync with the industry standards	4	
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	International Academia: https://web.stanford.edu/class/cs97si/03-data-structures.pdf AICTE-prescribed syllabus: https://makautexam.net/aicte_details/Syllabus/MCA/sem221.pdf Industry Mapping: The concepts delivered are in sync with the industry standards	8	1. Write a C program to print an array. 2. Write a C program to check whether a given string is Palindrome or not. 3. Write a C program to convert temperature from degree Centigrade to Fahrenheit. 4. Write a C program to sort an array. 5. Write a C program to print the largest and second largest element of the array. 6. Write a C program to display Fibonacci series. 7. Write a program that reads two 2D metrics from the console, verifies if metrics multiplication is possible or not. Then multiplies the metrics and prints the 3rd metrics. 8. Write a program that reads a 2D metrics and checks if the metrics is a symmetric metrics or not.

				9. Write a C program to print reverse array 10. Write a C program to check the sum of all elements of an array 11. Write a C program to check duplicate number in an array. 12. Write a C program to read a 2D array (with most of the elements as 0s) and then represent the same array as Sparse Metrics. 13. Write a C program to pass an array to a function using Call by Value, update the array values in the function, print the array elements both in the function and in the calling function. 14. Write a C program to pass an array to a function using Call by Reference, update the array values in the function, print the array elements both in the function and in the calling function. 15. Write a program to display n number of elements. Memory should be allocated dynamically using malloc(). 16. Write a program to display n number of elements. Memory should be allocated dynamically using calloc(). 17. Write a program to allocate memory using malloc() and then reallocate the previously allocated memory using realloc(). Display the elements which have been taken after reallocation. 18. Write a program to allocate memory using calloc() and then reallocate the previously allocated memory using realloc(). Display the elements which have been taken after reallocation. 19. Write a C program to search an element in an
--	--	--	--	---

					Array using dynamic memory allocation
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	International Academia: https://web.stanford.edu/class/cs97si/03-data-structures.pdf AICTE-prescribed syllabus: https://makauteam.net/aicte_details/Syllabus/MCA/se m221.pdf Industry Mapping: The concepts delivered are in sync with the industry standards	8	1. Write a Menu driven C program to accomplish the following functionalities in single linked list. a) Create a single linked list. b) Display the elements of a single linked list. c) Insert a node at the beginning of a single linked list. d) Insert a node at the end of a single linked list. e) Insert a node before a given node of a single linked list. f) Insert a node after a given node of a single linked list. g) Delete a node from the beginning of a single linked list. h) Delete a node from the end of a single linked list. i) Delete a node after a given node of a single linked list. j) Delete the entire single linked list. 2. Write a Menu driven C program to accomplish the following functionalities in circular linked list. a) Create a circular linked list. b) Display the elements of a circular linked list. c) Insert a node at the beginning of a circular linked list. d) Insert a node at the end of a circular linked list.

					e) Delete a node from the beginning of a circular linked list. f) Delete a node from the end of a circular linked list. g) Delete a node after a given node of a circular linked list. h) Delete the entire circular linked list. 3. Write a Menu driven C program to accomplish the following functionalities in doubly linked list. a) Create a doubly linked list. b) Display the elements of a doubly linked list. c) Insert a node at the beginning of a doubly linked list. d) Insert a node at the end of a doubly linked list. e) Insert a node before a given node of a doubly linked list. f) Insert a node after a given node of a doubly linked list. g) Delete a node from the beginning of a doubly linked list. h) Delete a node from the end of a doubly linked list. i) Delete a node after a given node of a doubly linked list. j) Delete the entire doubly linked list. 4. Write a Menu driven C program to accomplish the following functionalities in circular doubly linked list. a) Create a circular doubly linked list. b) Display the elements of a circular doubly linked
--	--	--	--	--	---

					list. c) Insert a node at the beginning of a circular doubly linked list. d) Insert a node at the end of a circular doubly linked list. e) Delete a node from the beginning of a circular doubly linked list. f) Delete a node from the end of a circular doubly linked list. g) Delete a node after a given node of a circular doubly linked list. h) Delete the entire circular doubly linked list.
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	International Academia: https://web.stanford.edu/class/cs97si/03-data-structures.pdf AICTE-prescribed syllabus: https://makauteexam.net/aicte_details/Syllabus/MCA/sem221.pdf Industry Mapping: The concepts delivered are in sync with the industry standards	7	1. Write a Menu driven C program to accomplish the following functionalities in Stack using an Array: - Insert an element into the stack using an array (Push Operation). - Delete an element from the stack using an array (Pop Operation). - Return the value of the topmost element of the stack (without deleting it from the stack) using an array. - Display the elements of a stack using an array. 2. Write a Menu driven C program to accomplish the following functionalities in Stack using Linked List: - Insert an element into the stack using a Linked List (Push Operation). - Delete an element from the stack using a Linked List (Pop Operation). - Return the value of the topmost element of the stack (without deleting it from the stack) using a Linked List. - Display the elements of the stack using a Linked List. 3. Write a program to convert an infix expression

					into its equivalent postfix notation. - Write a program to convert an infix expression into its equivalent prefix notation. - Write a program to evaluate a postfix expression. - Write a program to evaluate a prefix expression. - Write a program to print the Fibonacci series using recursion. - Write a program to solve the tower of Hanoi problem using recursion
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	International Academia: https://web.stanford.edu/class/cs97si/03-data-structures.pdf AICTE-prescribed syllabus: https://makautexam.net/aicte_details/Syllabus/MCA/sem221.pdf Industry Mapping: The concepts delivered are in sync with the industry standards	7	- Write a Menu driven C program to accomplish the following functionalities in Queue using an Array: - Insert an element into the queue using an array (Enqueue Operation). - Delete an element from the queue using an array (Dequeue Operation). - Return the value of the FRONT element of the queue (without deleting it from the queue) using an array (Peep operation) - Display the elements of a queue using an array. - Write a Menu driven C program to accomplish the following functionalities in Queue using Linked List: - Insert an element into the queue using a Linked List (Enqueue Operation). - Delete an element from the queue using a Linked List (Dequeue Operation). - Return the value of the FRONT element of the queue (without deleting it from the queue) using a Linked List (Peep operation). - Display the elements of a queue using a

					Linked List. 3. Write a Menu driven C program to accomplish the following functionalities in Circular Queue using Array: - Insert an element into the circular queue. - Delete an element from the circular queue. - Return the value of the FRONT element of the circular queue (without deleting it from the queue). Display the elements of a circular queue using the circular 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)	International Academia: https://web.stanford.edu/class/cs97si/03-data-structures.pdf AICTE-prescribed syllabus: https://makautexam.net/aicte_details/Syllabus/MCA/sem221.pdf Industry Mapping: The concepts delivered are in sync with the industry standards	6	- Write a C program to implement the concept of Bubble sort. - Write a C program to implement the concept of Selection sort. - Write a C program to implement the concept of Insertion sort. - Write a C program to implement the concept of Quick sort. - Write a C program to implement the concept of Merge sort. - Write a C program to show that Quick sort is better than Bubble sort. - Write a C program to show that merge sort is more effective than quick sort. - Write a C program to search an element in an array using linear search. - Write a C program to search an element in an array using binary search. - Write a C program to search an element in an array using interpolation search.

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



University of Engineering and Management
Institute of Engineering & Management, New Town Campus



Subject Name: Business English and Communication

Credit: 3

Subject Code: MCA105

Lecture Hours: 33

Name of the Course: Business English and Communication	
Course Code: MCA105	Semester: 1
Duration: 33	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	Mapping with Industry and International Academia	Lecture Hours
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.	International academia: https://www.coursera.org/learn/understanding-corporate-communications	4
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.	International Academia: https://ocw.mit.edu/courses/21g-222-expository-writing-for-bilingual-students-fall-2002/	8
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.	International Academia: https://ocw.mit.edu/courses/21g-222-expository-writing-for-bilingual-students-fall-2002/ Stanford Courses Online: https://online.stanford.edu/courses/csp-xcom88-high-impact-communication-advance-your-technology-career https://online.stanford.edu/courses/gsb-x0011-sharpen-your-communication-skills	8

			<i>Industry Mapping:</i> Campus Interviews and Recruitment Drives. Software: Orell Talk https://orelltalk.com/	
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.	International Academia: https://ocw.mit.edu/courses/21g-222-expository-writing-for-bilingual-students-fall-2002/ Stanford Courses Online: https://online.stanford.edu/courses/csp-xcom88-high-impact-communication-advance-your-technology-career https://online.stanford.edu/courses/gsb-x0011-sharpen-your-communication-skills <i>Industry Mapping:</i> • Campus Interviews and recruitment drives. • Software: Orell Talk https://orelltalk.com/	
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.	International Academia: https://ocw.mit.edu/courses/21g-222-expository-writing-for-bilingual-students-fall-2002/ <i>Industry Mapping:</i>	7

			Email writing and writing other relevant corporate documents.	
6	Practical Communication Skill Development	- Interactive sessions on negotiation and persuasion. - Group discussions and teamwork exercises	Software: Orell Talk https://orelltalk.com/ International academia: https://ocw.mit.edu/courses/15-280-communication-for-managers-fall-2016/ <i>Industry Mapping:</i> Campus Interviews and recruitment drives.	6

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 (5 th 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 (19 th ed)	Everest Publishing House



University of Engineering and Management
Institute of Engineering & Management, New Town Campus



Subject Name: Computational Mathematics and Statistics

Credit: 4

Lecture Hours: 48

Subject Code: MCA106

Course Name: Computational Mathematics and Statistics	
Course Code: MCA106	Semester: 1 st
Duration: 48 Hrs.	Maximum Marks: 100
Teaching Scheme	Examination Scheme
Theory: 4	End Semester Exam: 100
Tutorial: 1	Continuous Assessment: 100
Credit: 4	

Aim:	
Sl. No.	
1	To provide a rigorous foundation in discrete mathematical structures, including set theory and propositional logic, which are essential for software development and formal methods.
2	To develop analytical and statistical problem-solving skills necessary for data-driven computing, focusing on probability distributions, parameter estimation, and statistical hypothesis testing.
3	To introduce advanced mathematical concepts like fuzzy sets and generating functions that underpin modern computational fields such as artificial intelligence, soft computing, and operations research.

Objective:	
Sl. No.	
1	Introduce the fundamental concepts of relations, functions, and propositional logic (including CNF and DNF) to formulate formal computational models and logic circuits.
	Equip students with combinatorial tools, including the inclusion-exclusion principle, and mathematical induction techniques necessary for analyzing algorithm design and cryptography.
3	Teach foundational statistical measures, probability distributions (Binomial, Poisson, Normal), and hypothesis testing methods (z-test, t-test, chi-squared test, ANOVA) required for machine learning and numerical analysis.
4	Explore advanced mathematical frameworks, specifically generating functions for solving recurrences and fuzzy set operations for applications in financial modeling and soft computing.
Pre-Requisite:	
Sl. No.	
1.	A fundamental understanding of high-school-level algebra and introductory algorithmic thinking to effectively grasp abstract logic, recurrences, and numerical computations.
Course Outcome:	
Upon successful completion of this course, students will be able to:	
1.	Apply the principles of set theory, mathematical relations, and propositional logic to solve foundational discrete computational problems and model logic circuits.
2.	Analyze combinatorial problems and algorithm constraints using permutations, combinations, and the principle of mathematical induction.
3.	Evaluate datasets and statistical models by selecting and applying appropriate probability distributions and hypothesis testing methods (large and small sample tests) to assess data consistency and error.
4.	Formulate computational solutions for complex systems using advanced mathematical models, such as generating functions for recurrences and fuzzy sets for artificial intelligence applications.

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
CO2	3	3	2	1	-	-	-	-	-	-	-	1	2	-	1
CO3	3	3	1	3	2	-	-	-	-	-	-	1	1	3	2
CO4	3	3	2	2	1	-	-	-	-	-	-	2	1	2	3

Module number	Topic	Sub-topics	Mapping with Industry and International Academia	Lecture Hours
1	Set Theory and Functions	Set Theory: Definitions and operations (union, intersection, complement), Power sets and Cartesian products. Functions: Definitions and types (injective, surjective, bijective), Composition of functions and inverse functions.	Functions: Definitions and types (injective, surjective, bijective), Composition of functions and inverse functions.	4
2	Relations and Propositional Logic	Relations: Definitions and properties (reflexive, symmetric, transitive), Equivalence relations and partitions, Partial orders and Hasse diagrams. Propositional Logic: Propositions and logical connectives, Truth tables and logical equivalences, Normal forms (CNF, DNF).	Industry: Software development, logic circuits, artificial intelligence. Academia: Formal methods, logic in computer science	8
3	Combinatorics and Mathematical Inductions	Combinatorics: Permutations and combinations, Binomial theorem and Pascal's triangle, Inclusion-exclusion principle. Mathematical Inductions: Principle of mathematical induction,	Industry: Cryptography, algorithm design, network security. Academia: Discrete mathematics, theoretical computer science	8

4	Probability Distributions & Statistics	Statistics - measure of central tendency, dispersion (Moments, Skewness & Kurtosis). Least square curve fitting Introduction to mass function, density function, distribution function (Binomial, Poisson, Normal), estimation of parameters (unbiasedness-concept of noise/error, consistency). Testing of Hypothesis, Large Sample, Small Sample (z test, t test, chi-squared test, Anova)	Industry Mapping: https://www.sagemath.org/ , MATLAB International Academia: https://ocw.mit.edu/courses/18-440-probability-and-random-variables-spring-2014/pages/lecture-notes/ , https://ocw.mit.edu/courses/2-993j-introduction-to-numerical-analysis-for-engineering-13-002j-spring-2005/pages/lecture-notes/	6 6 8
5	Advanced Topics	Generating Functions: Definitions and basic properties, Applications in counting and solving recurrences. Fuzzy Sets: Definitions and basic properties, Operations on fuzzy sets (union, intersection, complement), Applications and example.	Industry: Financial modelling, operations research. Academia: Discrete mathematics, combinatorial analysis, soft computing, artificial intelligence	6

University of Engineering and Management

Institute of Engineering & Management, New Town Campus

Subject Name: Mental Maths for Professionals

Credit: 1

Subject Code: MCAGS101

Lecture Hours: 48

Module number	Topic	Sub-topics	Mapping with International/National/ State Level Exams	Lecture Hours	Corresponding Assignment
1	Quantitative Aptitude	Textbook: Quantitative Aptitude, Author: R.S Aggarwal, Publisher: S.Chand A. Quant Foundation 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	International Exams 1. <i>GRE</i> https://www.ets.org/gre/test-takers/general-test/prepare/content/verbal-reasoning.html#accordion-9f58105fc6-item-88093eca37 National Exams: 1. <i>UPSC Civil Services Exam</i> https://upsc.gov.in/sites/default/files/Notif-CSP-23-english-010223.pdf), pg 25-26 2. <i>UPSC Combined Defence Services</i> https://upsc.gov.in/sites/default/files/Notif-CDS-I-Exam-2023-	24	1. Assignment on Numerical Problem Solving using Vedic Mathematics principle. 2. Assignment on Numerical Problem-Solving using percentage to fraction relation.

		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.	Engl-211222.pdf), pg 20-21 3. Combined Graduate Level conducted by SSC (https://ssc.nic.in/SSCFileServer/PortalManagement/UploadFiles/notice_CGLE_03042023.pdf) pg. 20-22 4. Intelligence Bureau ACIO (https://www.pw.live/exams/wp-content/uploads/2023/11/IB-ACIO-Recruitment-2023-Notification-Emp-News.pdf) State Level Exams: 1. Civil Services Executive Exam (WBCS) (https://wbpsc.gov.in/Download?param1=20230225142430_Syllabus.pdf&param2=advertisement, pg 1 2. Miscellaneous Services Recruitment Examination (file:///C:/Users/UEMK/Downloads/2707970_2019.pdf) pg 1		
2	Logical Reasoning	Textbook: Modern Approach to Verbal and Non-Verbal Reasoning, Author Dr. R.S Aggarwal, Publisher: S.Chand 1. Coding and Decoding (Chapter 4)	International Exams 1. GRE (https://www.ets.org/gre/test-takers/general-test/prepare/content/verbal-reasoning.html#accordion-9f58105fc6-item-88093eca37)	24	1. Assignment on Letter Coding, Number Coding, Conditional Coding and Chinese Pattern. 2. Assignment on Directions and Distance 3. Assignment on Indicating based Blood Relation, Coding based Blood

		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.	National Exams: 1. UPSC Civil Services Exam (https://upsc.gov.in/sites/default/files/Notif-CSP-23-engl-010223.pdf), pg 25-26 2. UPSC Combined Defence Services (https://upsc.gov.in/sites/default/files/Notif-CDS-I-Exam-2023-Engl-211222.pdf), pg 20-21 3. Combined Graduate Level conducted by SSC (https://ssc.nic.in/SSCFileServer/PortalManagement/UploadedFiles/notice_CGLE_03042023.pdf) pg. 20-22 4. Intelligence Bureau ACIO (https://www.pw.live/exams/wp-content/uploads/2023/11/IB-ACIO-Recruitment-2023-Notification-Emp-News.pdf) State Level Exams: 1. Civil Services Executive Exam (WBCS) (https://wbpsc.gov.in/Download?param1=20230225142430_Syllabus.pdf&param2=advertisement,		Relation and Family Tree based Blood Relation
--	--	--	--	--	--

			<i>pg 1</i> <i>Miscellaneous Services Recruitment Examination</i> (file:///C:/Users/UEMK/Downloads/2707970_2019.pdf) <i>pg 1</i>		
--	--	--	--	--	--

Learning Resources: Text

Book

1. Quantitative Aptitude- R.S Agarwal
2. Verbal & non-verbal reasoning- R.S Agarwal



University of Engineering and Management
Institute of Engineering & Management, New Town Campus



Syllabus for MCA Admission Batch 2025, 1st Semester

Subject Name: RESEARCH METHODOLOGY AND IPR

Credit: 02

Subject Code: MCA171

Lecture Hours: 36 Hrs.

Name of the Course: RESEARCH METHODOLOGY AND IPR	
Course Code: MCA171	Semester: FIRST
Duration: 36	Maximum Marks: 100
Teaching Scheme	Examination Scheme
Theory: 2	End Semester Exam: 100
Tutorial: 1	Continuous Assessment: 100
Credit: 2	

Aim:	
Sl. No.	
1	To develop comprehensive understanding of research methodologies and systematic approaches to academic inquiry
2	To foster awareness of intellectual property rights and their application in research and development
3	To equip students with practical skills for conducting ethical and legally compliant research

Objective:	
Sl. No.	
1	To enable students to formulate well-defined research problems and design appropriate research frameworks
2	To provide knowledge of various data collection methods and their appropriate application in different research contexts
3	To develop competency in data analysis techniques and interpretation of research findings
4	To instil understanding of IPR principles, patent processes, and research ethics in academic and industrial settings
Pre-Requisite: Basic understanding of statistics and data interpretation. Familiarity with computer applications and software tools. Foundational knowledge of research concepts and academic writing	
Course Outcome:	
CO1	Formulate and design a research problem.
CO2	Understand and Comprehend the Data Collection Methods
CO3	Perform Data analysis and acquire Insights.
CO4	Understand IPR and follow research ethics.

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

Module number	Topic	Sub-topics	Lecture Hours
1	Research Design	Overview of research process and design, Use of Secondary and exploratory data to answer the research question, Qualitative research, Observation studies, Experiments and Surveys.	6
2	Data Collection and Sources	Measurements, Measurement Scales, Questionnaires and Instruments, Sampling and methods. Data - Preparing, Exploring, examining and displaying.	6
3	Data Analysis and Reporting	Overview of Multivariate analysis, Hypotheses testing and Measures of Association. Presenting Insights and findings using written reports and oral presentations.	6
4	Intellectual Property Rights	Intellectual Property – The concept of IPR, Evolution and development of concept of IPR, IPR development process, Trade secrets, utility Models, IPR & Biodiversity, Role of WIPO and WTO in IPR establishments, Right of Property, Common rules of IPR practices, Types and Features of IPR Agreement, Trademark, Functions of UNESCO in IPR maintenance.	6
5	Patents	Patents – objectives and benefits of patent, Concept, features of patent, Inventive step, Specification, Types of patent application, process E-filing, Examination of patent, Grant of patent, Revocation, Equitable Assignments, Licences, Licensing of related patents, patent agents, Registration of patent agents.	6

List of Books Text Books:

Name of Author	Title of the Book	Edition/ISSN/ISBN	Name of the Publisher
1. Cooper Donald R, Schindler Pamela S and Sharma JK, “Business Research Methods”, Tata McGraw Hill Education, 11e (2012).			

Reference Books:

1. Catherine J. Holland, “Intellectual property: Patents, Trademarks, Copyrights, Trade Secrets”, Entrepreneur Press, 2007.
2. David Hunt, Long Nguyen, Matthew Rodgers, “Patent searching: tools & techniques”, Wiley, 2007.
3. The Institute of Company Secretaries of India, Statutory body under an Act of parliament, “Professional Programme Intellectual Property Rights, Law and practice”, September 2013.



University of Engineering and Management
Institute of Engineering & Management, New Town Campus
Syllabus for MCA Admission Batch 2026, 1st Semester



Subject Name: Basic Python Programming

Credit: 3

Subject Code: MCA172

Lecture Hours: 40

Name of the Course: Basic Python Programming	
Course Code: MCA172	Semester: 2nd
Duration: 40 Hrs.	Maximum Marks: 100
Teaching Scheme	Examination Scheme
Sessional: 2	End Semester Exam: 100
Practical: 3	Credit: 2

Aim:	
Sl. No.	
1	To provide a strong foundation in Python programming language and computational problem-solving.
2	To introduce procedural, modular, and object-oriented programming constructs available in Python.
3	To develop practical skills in building real-world applications using Python's standard library, file I/O, and database connectivity.

Objective:	
Sl. No.	
1	Understanding the basic syntax, variables, expressions, data types, and operators of Python.
2	Clear understanding of control flow constructs (conditionals, loops) and user-defined functions.
3	Proficient use of Python's core data structures — lists, tuples, dictionaries, and sets.
4	Understanding of modules, packages, file I/O, and exception handling in Python.
5	Develop a base for Object-Oriented Programming and modern application development using Python.

Pre-Requisite:	
Sl. No.	
1.	Basic knowledge of computer fundamentals.
2.	Familiarity with any introductory programming language (preferably C).
Course Outcome:	
1.	Understand Python fundamentals including data types, operators, and control-flow constructs.
2.	Develop modular Python programs using functions, scope rules, recursion, and standard library modules.
3.	Apply Python's core data structures (lists, tuples, dictionaries, sets) along with file I/O and exception handling to solve real-world problems.

4.	Design object-oriented solutions using classes, objects, inheritance, polymorphism, and constructors.
----	---

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	3	-	-	-	-	2	-	2	3	-	-
CO2	3	3	2	2	3	-	-	-	-	2	-	2	3	2	-
CO3	3	3	3	3	3	-	-	-	-	2	-	2	3	2	-
CO4	3	3	3	3	3	-	-	-	-	2	-	2	3	3	2

Module-wise Syllabus Distribution

Module number	Topic	Sub-topics	Mapping with Industry and International Academia	Lecture Hours
1	Fundamentals & Basics	Introduction to Python: history, features, interpreter, interactive and script mode. Variables, expressions, statements, identifiers, keywords, comments. Standard data types: Numbers (int, float, complex), Booleans, Strings — basic string operations and slicing. Operators: arithmetic, relational, logical, assignment, bitwise, and membership operators. Input/output functions and type conversion.	International Academia: https://web.stanford.edu/class/cs41/ AICTE-prescribed syllabus: https://www.aicte-india.org/downloads/mcadegree.pdf Industry Mapping: The concepts delivered are in sync with industry standards for software development, data science, and automation.	7
2	Control Flow & Functions	Conditionals: if, else, elif, nested conditional statements. Loops: while and for loops, use of range(), break, continue,	International Academia: https://web.stanford.edu/class/cs41/ AICTE-prescribed syllabus:	8

Module number	Topic	Sub-topics	Mapping with Industry and International Academia	Lecture Hours
		and pass statements. Functions: defining and calling functions, parameters and arguments (positional, keyword, default), return statement, local vs. global scope, lambda functions, recursion.	https://www.aicte-india.org/downloads/mcadegree.pdf Industry Mapping: The concepts delivered are in sync with industry standards for software development, data science, and automation.	
3	Core Data Structures (Collections)	Lists: creation, indexing, slicing, built-in methods, mutability, nested lists, list comprehension. Tuples: immutability, packing/unpacking, tuple methods. Dictionaries: key-value pairs, accessing and modifying elements, dictionary methods, dictionary comprehension. Sets: creation, set operations (union, intersection, difference), set methods.	International Academia: https://web.stanford.edu/class/cs41/ AICTE-prescribed syllabus: https://www.aicte-india.org/downloads/mcadegree.pdf Industry Mapping: The concepts delivered are in sync with industry standards for software development, data science, and automation.	9
4	Modules, Exception Handling & File I/O	Functions (advanced): keyword and default arguments, global/local scope, recursion (revisited). Modules: importing modules, from-import, packages, introduction to the Python Standard Library (math, random, os, datetime). Exception Handling: try/except/finally, handling common errors, raise statement, built-in exceptions, user-defined exceptions. File I/O: opening, reading, writing, and closing text and binary files; working with file pointers.	International Academia: https://web.stanford.edu/class/cs41/ AICTE-prescribed syllabus: https://www.aicte-india.org/downloads/mcadegree.pdf Industry Mapping: The concepts delivered are in sync with industry standards for software development, data science, and automation.	9
5	Advanced Programming (OOP)	Object-Oriented Programming in Python: classes and objects, attributes and methods, constructors (__init__), self parameter. Inheritance (single, multiple, multilevel), method overriding, polymorphism, encapsulation, abstraction. Introduction to modern tools: CRUD operations with MongoDB / building a simple web page with Django (project-level exposure).	International Academia: https://web.stanford.edu/class/cs41/ AICTE-prescribed syllabus: https://www.aicte-india.org/downloads/mcadegree.pdf Industry Mapping: The concepts delivered are in sync with industry standards for software development, data science, and automation.	7

List of Books			
Text Books:			
Name of Author	Title of the Book	Edition/ISSN/ISBN	Name of the Publisher
John V. Guttag	Introduction to Computation and Programming Using Python	3rd Edition	MIT Press / PHI
Sheetal Taneja	Python Programming: A Modular Approach	1st Edition	Pearson Education
Reference Books:			
Robert Sedgewick, Kevin Wayne, Robert Dondero	Introduction to Programming in Python: An Interdisciplinary Approach	1st Edition	Pearson / Addison-Wesley
Mark Lutz	Learning Python	5th Edition	O'Reilly Media
Allen B. Downey	Think Python: How to Think Like a Computer Scientist	2nd Edition	O'Reilly Media

Lab Assignments (MCA291)

Basics & Control Flow

1. Temperature Conversion: Write a program to convert Celsius to Fahrenheit and vice versa.
2. Prime Numbers: Check if a number is prime or print all primes in a given range.
3. Fibonacci Series: Generate the series up to n terms using loops or recursion.
4. Pattern Printing: Use nested loops to print star or number patterns.

Data Structures

5. List Operations: Create a list and perform operations like append, remove, sort, and find the largest/smallest element.
6. String Manipulation: Implement programs for vowel counting, string slicing, and palindrome checking.
7. Dictionary Application: Develop a student record system using names as keys and grades as values.

Advanced Concepts

8. Search & Sort: Implement Linear Search, Binary Search, and Selection or Bubble Sort.
9. OOP: Create a “Bank Account” or “Library Management” system using classes, objects, and inheritance.
10. File Handling: Read from and write to a text file (e.g., counting word frequency or reversing file contents).

Project / Modern Tools

11. Basic CRUD operations using MongoDB with Python.
12. Build a simple web page using the Django framework.

Story-based Assessment Questions (Unit 1 – Unit 5)

Unit 1: Fundamentals — The Online Store

Story: You are building a checkout system for a digital store. A customer buys an item with a price P and a tax rate T (percentage). Write a program to calculate the final price after adding tax and then applying a flat discount of ₹50.

Concepts: *Variables, Arithmetic Operators, Type Conversion.*

- Test Case 1: Price = 500, Tax = 10% → Expected Output: ₹500.0 (calculated as $500 + 50 - 50$)
- Test Case 2: Price = 100, Tax = 5% → Expected Output: ₹55.0 (calculated as $100 + 5 - 50$)

Unit 2: Control Flow & Functions — The Smart Fitness Tracker

Story: Your fitness tracker needs to give you a message based on your daily steps. Write a function `check_goal(steps)` that returns “Goal Reached” if steps are 10,000 or more, “Keep Walking” if between 5,000 and 9,999, and “Sedentary” if below 5,000.

Concepts: *Functions, if-elif-else, Comparison Operators.*

- Test Case 1: steps = 12000 → Expected Output: “Goal Reached”.
- Test Case 2: steps = 4500 → Expected Output: “Sedentary”.

Unit 3: Core Data Structures — The Movie Theater

Story: You have a list of available seats in a theater: `seats = [10, 12, 15, 20, 22]`. A user wants to book a specific seat number. Write a program that checks if the seat is available. If it is, remove it from the list and print “Booked”; otherwise, print “Unavailable”.

Concepts: *Lists, List Methods (remove, in operator).*

- Test Case 1: User Choice = 15 → Expected Output: “Booked”, Updated List: [10, 12, 20, 22].
- Test Case 2: User Choice = 11 → Expected Output: “Unavailable”.

Unit 4: Files & Exceptions — The Grade Reporter

Story: You are reading a file grades.txt that contains a list of numbers. Some lines might accidentally contain text instead of numbers. Write a program that reads the file, converts each line to an integer, and calculates the total sum. Use exception handling to skip any line that is not a valid number.

Concepts: *File I/O (open, readlines), try-except, ValueError.*

- Test Case 1: File contents 80\n90\nABC\n70 → Expected Output: Total Sum: 240 (skips ‘ABC’).
- Test Case 2: File is missing → Expected Output: “Error: File not found”.

Unit 5: Advanced Programming — The Virtual Bank

Story: Create a class BankAccount with an initial balance. It should have a method withdraw(amount) that reduces the balance but only if there are sufficient funds. If funds are insufficient, print “Insufficient Balance”.

Concepts: *OOP (Classes, Objects, Methods, self).*

- Test Case 1: Initial Balance = 1000, Withdraw = 400 → Expected Output: Balance: 600.
- Test Case 2: Initial Balance = 500, Withdraw = 600 → Expected Output: “Insufficient Balance”.

University of Engineering and Management

Institute of Engineering & Management, New Town Campus

Syllabus for MCA Admission Batch 2026, 2nd Semester

Subject Name: Database Management Systems

Credit: 4

Subject Code: MCA201

Lecture Hours: 40

Name of the Course: Database Management Systems	
Course Code: MCA201& MCA291	Semester: 2 nd
Duration: 40 Hrs.	Maximum Marks: 100
Teaching Scheme	Examination Scheme
Theory: 4	End Semester Exam: 100
Tutorial: 1	Continuous Assessment: 100
Practical: 4	Practical Sessional Internal continuous evaluation: 100
Credit: 4+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:	Understand the basic concepts and the applications of database systems.
CO2:	Master the basics of SQL and construct queries using SQL.
CO3:	Understand the relational database design principles.
CO4:	Familiar with the basic issues of transaction processing and concurrency control.
Relevant Links:	
DBMS Study Material DBMS NPTEL Link DBMS Coursera Link DBMS Linkedin Learning Link	

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PO1 2	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	Mapping with Industry & Academia	Lecture Hours	Corresponding Lab Assignment
1	Introduction: Database System Applications	Database System Applications, Purpose of Database Systems, View of Data, Database Languages – 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 Database design: Database Design and ER 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.	International Academia: https://ocw.mit.edu/course/s/1-264j-database-internet-and-systems-integration-technologies-fall-2013/pages/syllabus/ AICTE-prescribed syllabus: https://www.aicte-india.org/sites/default/files/Model_Curriculum_flipbook/CSE(UG)/index.html#p=123 Industry Mapping and Gap Analysis: Basic DBMS was previously available in the MCA syllabus and is in sync with the syllabus of AICTE and the University of Berkeley. All the basic concepts are as per the industry standards.	3	Annexure – I (SQL Query based Lab – Assignments) Assignment – 1: Design E-R Diagrams for Different case studies
	Relational Model	Introduction to the Relational Model, Integrity Constraints over Relations, Enforcing Integrity constraints, Querying relational data, Logical database Design: ER to Relational, Introduction to Views, Destroying / Altering Tables and Views.		5	Assignment – 2: Case Studies using basic SQL Relational Algebra Operations
2	Relational Algebra and Calculus	Preliminaries, Relational Algebra, Relational calculus – Tuple relational Calculus, Domain relational calculus, Expressive Power of Algebra and calculus. SQL: Queries, Constraints, Triggers: Form of Basic SQL Query, UNION, INTERSECT, and EXCEPT, Nested Queries, Aggregate Operators, NULL values Complex Integrity All JNTU World Constraints in SQL, Triggers and Active Databases, Designing Active Databases.	International Academia: https://ocw.mit.edu/course/s/1-264j-database-internet-and-systems-integration-technologies-fall-2013/pages/syllabus/ AICTE-prescribed syllabus: https://www.aicte-india.org/sites/default/files/Model_Curriculum_flipbook/CSE(UG)/index.html#p=123	10	Assignment – 3: Case Studies using basic SQL Relational Algebra Operations

			Industry Mapping and Gap Analysis: Basic Java programming was previously available in the MCA syllabus and is in sync with the syllabus of AICTE and the University of Berkeley. All the basic concepts are as per the industry standards.		
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.	International Academia: https://ocw.mit.edu/course/s/1-264j-database-internet-and-systems-integration-technologies-fall-2013/pages/syllabus/ AICTE-prescribed syllabus: https://www.aicte-india.org/sites/default/files/Model_Curriculum/flipbook/CSE(UG)/index.html#p=123 Industry Mapping and Gap Analysis: Collection Class is incorporated as per international standards.	6	Assignment – 4: SQL based assignment on different normalforms
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.	International Academia: https://ocw.mit.edu/course/s/1-264j-database-internet-and-systems-integration-technologies-fall-2013/pages/syllabus/ AICTE-prescribed syllabus: https://www.aicte-india.org/sites/default/files/Model_Curriculum/flipbook/CSE(UG)/index.html#p	8	Assignment – 5: SQL-based assignment on Transaction Management
	Concurrency Control	Lock-Based Protocols, Multiple Granularity, Timestamp-Based Protocols, Validation-Based Protocols, Multiversion Schemes. Recovery System-Failure Classification,			Assignment – 6: SQL-based assignment on Transaction Management

		Storage, Recovery and Atomicity, Recovery Algorithm, Buffer Management, Failure with loss of nonvolatile storage, Early Lock Release and Logical Undo Operations, Remote Backup systems.	=123 Industry Mapping and Gap Analysis: Java applets were deprecated by Java 9 in 2017. Thus, only the basics of the applet are included and Swings, AWT and event handling are taught in detail. Oracle Announces End Of Java Applet Support		
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)	<i>International Academia & AICTE-prescribed syllabus:</i> https://ocw.mit.edu/course/s/1-264j-database-internet-and-systems-integration-technologies-fall-2013/pages/syllabus/ Industry Mapping and Gap Analysis: e-Brochure (cdac.in) Industry requirement for full stack development, previously missing which is partially incorporated.	8	
	B+ Trees	A Dynamic Index Structure, Search, Insert, Delete. Hash- Based Indexing: Static Hashing, Extendible hashing, Linear Hashing, Extendible vs. Linear Hashing.			
	Introduction to Unstructured Data and NoSQL	Characteristics of unstructured data, limitations of relational databases for unstructured storage, NoSQL data models (Key-Value, Document, Column-Family, Graph), and industry examples handling unstructured data (MongoDB, Cassandra, Neo4j, Redis).	<i>International Academia & AICTE-prescribed syllabus:</i> https://efaidnbmnnnibpcajpcglclefindmkaj/https://www.aicte.gov.in/sites/default/files/Updated-AICTE%20-%20UG%20CSE.pdf Industry Mapping and Gap Analysis https://ocw.mit.edu/course		
					Assignment – 6: Implement B+ tree in Python Assignment – 7: 1. Inserting JSON Documents 2. Querying JSON Documents 3. Updating JSON Documents 4. Deleting Documents 5. Delete one document 6. Delete multiple documents 7. Drop a collection or

			<i>s/6-830-database-systems-fall-2010/pages/readings/lec19 /</i>		database
--	--	--	--	--	----------

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 (McGraw-Hill International Editions: Computer Science Series)	ISE	Dreamtech Press

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_Outrate 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.

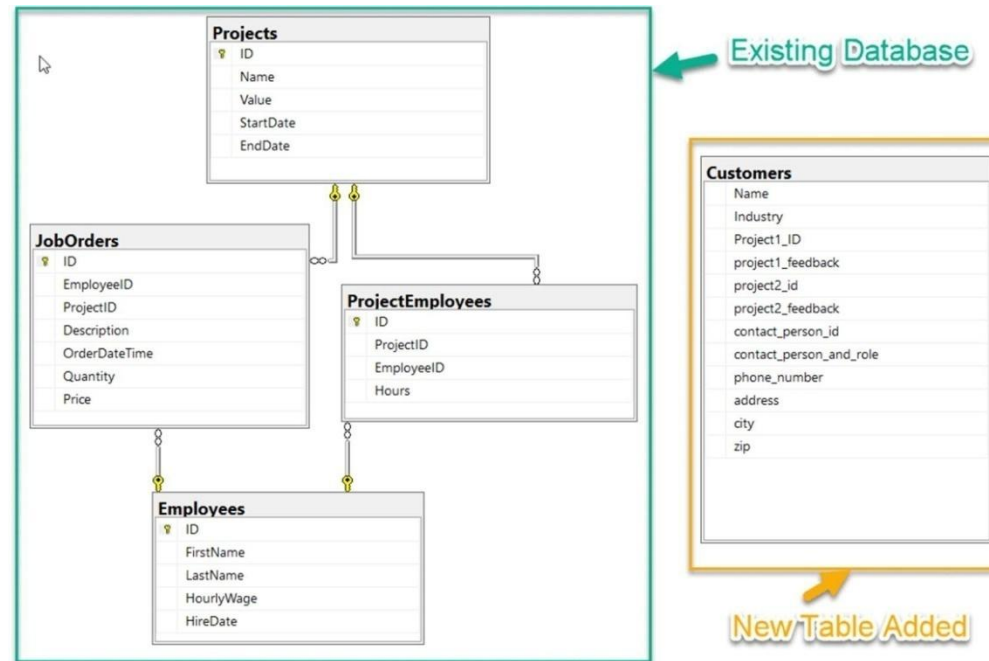
Question (F): Develop a SQL query to list the details of customers who have performed three transaction on a day.

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.

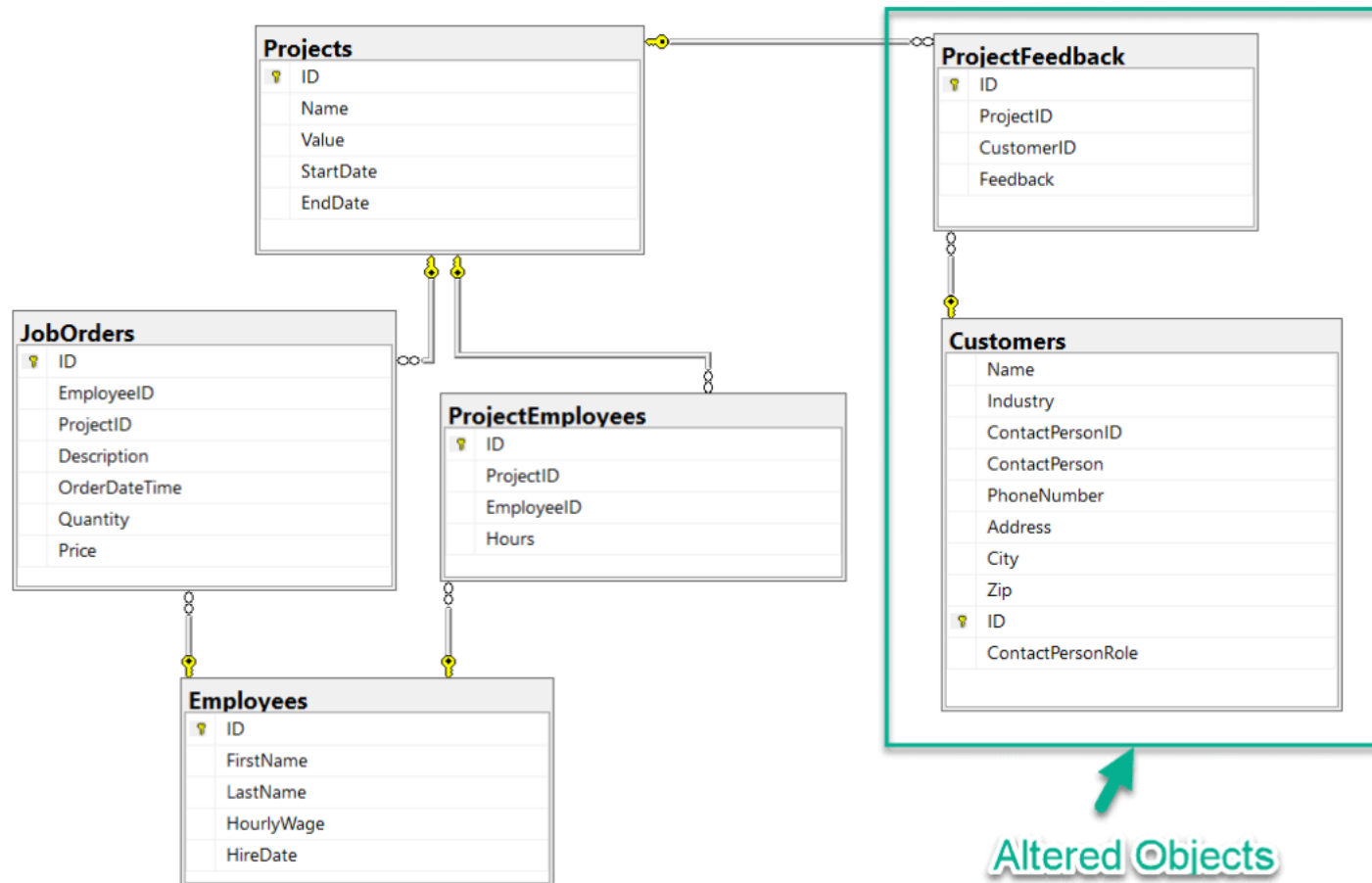
Also, we can see there are two repeating groups of columns in this table - (Project1_ID, Project1_FeedBack) and (Project2_ID, Project2_Feedback). We need to get these removed from this table.

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

- 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.
- split the column *ContactPersonAndRole* into two individual columns. This can be done in two steps as follows:
 - Rename the original column from *ContactPersonAndRole* to *ContactPerson*.
 - Add a new column for *ContactPersonRole*.
- 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

Syllabus for MCA Admission Batch 2026, 2nd Semester

Subject Name: Object-Oriented Programming with Java

Credit: 4

Subject Code: MCA202

Lecture Hours: 40

Name of the Course: Object-Oriented Programming with Java & Object-Oriented Programming with Java Laboratory	
Course Code: MCA202 and MCA292	Semester: 2nd
Duration: 40 Hrs.	Maximum Marks: 50
Teaching Scheme	Examination Scheme
Theory: 3	End Semester Exam: 100
Tutorial: 1	Continuous Assessment: 100
Practical: 2	Practical Sessional Internal continuous evaluation: 40
Credit: 4+2	Practical Sessional external examination: 60
AIM:	
Sl. No.	
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.

Course Objectives:	
Sl. No.	
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.
Pre-Requisite:	
Sl. No.	
1.	Basics of programming language.
2.	Logic building skills.
Course Outcome:	
Sl. No.	
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 using JDBC, JSP, Servlets and Databases and make small projects based on them.
Relevant Links:	
JAVA Course Material JAVA NPTEL Link JAVA Coursera Link JAVA Linkedin Learning	

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	Mapping with Industry and Academia	Lecture Hours	Corresponding Lab Assignment
1	OOPs Concept	Object, Class, Data abstraction, Data encapsulation, Inheritance, Polymorphism, Dynamic binding	International Academia: First Course in Java – EL ENG X429.9 UC Berkeley Extension AICTE-prescribed syllabus: MAKAUT – MCA 2ND SEM SYLLABUS Industry Mapping and Gap Analysis: Basic Java programming was previously available in the MCA syllabus and is in sync with the syllabus of AICTE and the University of Berkeley. All the basic concepts are as per the industry standards.	1	Annexure – I (Programming based Lab – Assignments)
	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.		1	Assignment – 1: Basic Programming and Command Line Arguments
	Data Concept	Data Types, Variables, Arrays and constants Tokens in Java (Identifiers, Literals, Keywords, Operator)		2	
	Control Statements	Simple if statement, if-else statement, Nesting of if-else statement, switch statement		2	Assignment – 2: Constructors & Inheritance
	Iteration Statement	for loop, while loop, do-while loop		1	Assignment – 3: Flow Control
	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		3	
2	String and String Buffer	Use of different functions	International Academia: First Course in Java – EL ENG X429.9 UC Berkeley Extension AICTE-prescribed syllabus: MAKAUT – MCA 2ND SEM SYLLABUS Industry Mapping and Gap Analysis: Basic Java programming was previously available in the MCA syllabus and is in sync with the syllabus of AICTE and the University of Berkeley. All the basic	2	Assignment – 4: Inheritance and Dynamic Polymorphism
	Inheritance	Basic concepts, types of inheritance, use of super keyword, overriding methods.		2	
	Packages, Interfaces	User-defined package, standard packages, import package, Class path, how to create interface, use and extend interface		3	Assignment – 5: Abstract class & Interface in Java.
	Multithreaded Programming	Overview, Thread Life cycle, Advantages of multithreading over multitasking, Thread Creation, Synchronized threads, Synchronized Methods		3	Assignment – 6: Threads, Multithreading & Thread Synchronization

			concepts are as per the industry standards.		
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.	International Academia: CS – 108 – Object Oriented System Design Stanford University AICTE-prescribed syllabus: MAKAUT – MCA 2ND SEM SYLLABUS Industry Mapping and Gap Analysis: Collection Class is incorporated as per international standards.	3	Assignment – 7: Exception Handling & Collections
	Collections	Collections, Iteration, Set and SortedSet, List, Map and SortedMap, Legacy Collection Types		3	
4	Stream	Byte Streams, Input Stream, Output Stream Character Streams (Reader, Writer), How Files and Streams Work, Working with Reader classes (InputStreamReader, BufferedReader)	International Academia: CS – 108 – Object Oriented System Design Stanford University AICTE-prescribed syllabus: MAKAUT – MCA 2ND SEM SYLLABUS Industry Mapping and Gap Analysis: Java applets were deprecated by Java 9 in 2017. Thus, only the basics of the applet are included and Swings, AWT and event handling are taught in detail. Oracle Announces End Of Java Applet Support	3	Assignment – 8: Keyboard input and string handling in Java
	Applets	Applet vs. Application, Applet class, Advantages of Applet, Applet		2	
	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		3	Any one web-based project as per Annexure-II.
5	JDBC and Web Application Development	Generic Servlet, HTTP Servlet, Server-Side Include, Overview of JSP, JSP Components, Bean, Session Tracking, Accessing Database with JDBC, Basics, Manipulating Databases with JDBC	International Academia & AICTE-prescribed syllabus: CSE IIT KGP Object Oriented Programming with JAVA	6	

			Industry Mapping and Gap Analysis: e-Brochure (cdac.in) Industry requirement for full stack development, previously missing which is partially incorporated.		
--	--	--	--	--	--

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

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 the message “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 and dual_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.
Staff (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.

$$e = 1/0! + 1/1! + 1/2! + 1/3! + \dots + 1/n!$$
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 and avgMarks. 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:

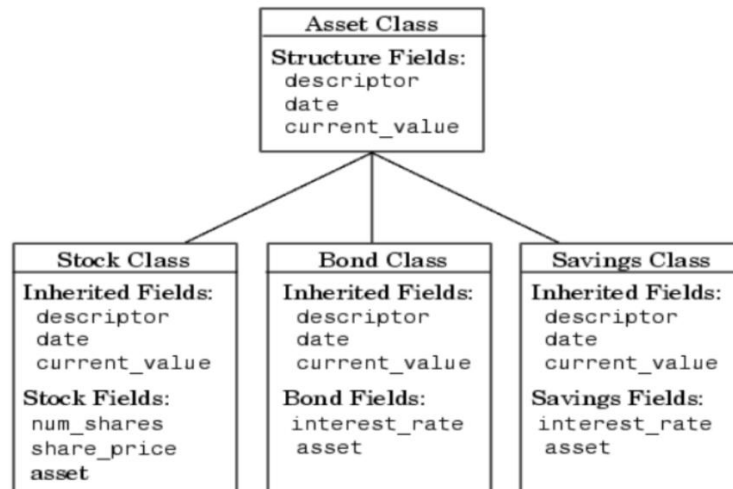
Data Members:

(a) rateOfInterest

Methods:

(a) calculateAount()

5. Implement the below Diagram.



Here, Asset class is an abstract class containing an abstract method `displayDetails()` 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 to its 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.

Annexure – II (Java based minor project – As per Software Development Life Cycle)

1. E-commerce project in java using JDBC and MYSQL

2. Online Pharmacy Management System using JDBC and MYSQL
3. Home Improvement System Using JDBC and MYSQL
4. Jewellery Shop Management Using JDBC and MYSQL
5. Student Management System Using JDBC and MYSQL
6. Hotel Management Project Using JDBC and MYSQL
7. Hospital Management System Using JDBC and MYSQL
8. Online Bike Service Booking Using JDBC and MYSQL
9. Online Food Delivery Using JDBC and MYSQL
10. Online Movie Ticket Booking Using JDBC and MYSQL
11. Bank management system Using JDBC and MYSQL

Syllabus for MCA Admission Batch 2026, 2nd Semester

Subject Name: Data Communication & Computer Networks

Credit: 4

Subject Code: MCA203

Lecture Hours: 40

Name of the Course: Data Communication & Computer Networks	
Course Code: MCA203	Semester: 2 nd
Duration: 40 Hrs.	Maximum Marks: 100
Teaching Scheme	Examination Scheme
Theory: 3	End Semester Exam: 100
Tutorial: 1	Continuous Assessment: 100
Credit: 4	

Aim:	
Sl. No.	
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:	
Sl. No.	
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:	
Sl. No.	
1.	Knowledge of basic data communication & network security.

Course Outcome:	
CO1	Identify the different components in a Communication System and their respective roles.
CO2	Describe the technical issues related to the Networks
CO3	Defining the standard model and protocols of networking
CO4	Understand the basics of data communication, networking, internet and their importance.
Relevant Links:	
DCCN Study Material DCCN NPTEL LINK DCCN Coursera Link DCCN LinkedIn Learning Link	

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	Mapping with Industry and International Academia	Lecture Hours
1	Introduction to Networks & Network Model	Introduction to communication systems, components, Transmission Impairments, Performance criteria of a communication system. Goals of computer Network, Networks: Classification, Components and Topology, categories of network [LAN, MAN, WAN]; Internet: brief history, internet today; Protocols and standards; OSI and TCP/IP model	International Academia: https://online.stanford.edu/courses/cs144-introduction-computer-networking AICTE-prescribed syllabus: https://makautexam.net/aicte_details/Syllabus/MCA/se m221.pdf Industry Mapping and Gap Analysis: Basic data communication & network system was previously available in the MCA syllabus and is in sync with the syllabus of AICTE and the Stanford University. All the basic concepts are as per the industry standards.	7
2	Physical Layer	Data, signal and Transmission: Analog and Digital, Transmission modes, Overview of data [analog & digital], signal [analog & digital], transmission [analog & digital] & transmission media [guided & unguided]; Circuit switching; time division & space division switch, TDM bus; Telephone Network.	International Academia: https://online.stanford.edu/courses/cs144-introduction-computer-networking AICTE-prescribed syllabus: https://makautexam.net/aicte_details/Syllabus/MCA/se m221.pdf Industry Mapping & Gap Analysis: Practical knowledge on digital electronics circuit related to signal oscillation and transmission as well as various types of transmission media should be incorporated as per industry standard.	6
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 sub layer: Point to point protocol, FDDI, token bus, token ring; Reservation, polling, concentration; Multiple access protocols: ALOHA, CSMA, FDMA, TDMA, CDMA; Ethernet	International Academia: https://online.stanford.edu/courses/cs144-introduction-computer-networking AICTE-prescribed syllabus: https://makautexam.net/aicte_details/Syllabus/MCA/se m221.pdf Industry Mapping and Gap Analysis: Needs to add practical exposure to design and	6

			implement various error detection and correction methods using any programming paradigm.	
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.	International Academia: https://online.stanford.edu/courses/cs144-introduction-computer-networking AICTE-prescribed syllabus: https://makautexam.net/aicte_details/Syllabus/MCA/se m221.pdf Industry Mapping and Gap Analysis: Needs to incorporate hands-on training on network hardware devices and should have some programming knowledge on routing strategies as per industry requirement.	6
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]	International Academia: https://online.stanford.edu/courses/cs144-introduction-computer-networking AICTE-prescribed syllabus: https://makautexam.net/aicte_details/Syllabus/MCA/se m221.pdf Industry Mapping and Gap Analysis: Needs to add practical exposure to design and implement various transport layer protocols using any programming paradigm which is the key requirement of IT industry.	6
6	Application Layer	Details of Application Layer protocols/services such as HTTP, FTP, Telnet, SMTP & WWW and other	International Academia: https://online.stanford.edu/courses/cs144-introduction-computer-networking AICTE-prescribed syllabus: https://makautexam.net/aicte_details/Syllabus/MCA/se m221.pdf Industry Mapping and Gap Analysis:	5

			Needs to introduce various application layer protocols using any programming paradigm which is the key requirement of IT industry as well as academia.	
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.	International Academia: https://online.stanford.edu/courses/soe-y0001-cryptography-i AICTE-prescribed syllabus: https://makautexam.net/aicte_details/Syllabus/MCA/se m221.pdf Industry Mapping and Gap Analysis: Needs to incorporate in-depth details of cryptography and related algorithms and should facilitate such programming environment to implement these algorithms as per industry requirements.	4

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	4 th Ed	TMH
Andrew S. Tannenbaum	Computer Networks	6 th Ed	PHI
Reference Books:			
William Stallings	Data & Computer Communications	10 th Ed	PHI
Douglas E. Comer	Computer Networks and Internets with Internet Applications	4 th Ed	Pearson
Jean Warland	Communication Networks: A First Course	2 nd Ed	TMH
Ed Title	Schaum's Outline of Computer Networking	2 nd Ed	TMH

Syllabus for MCA Admission Batch 2026, 2nd Semester

Subject Name: Advanced Data Structure using Python

Credit: 4

Subject Code: MCA204

Lecture Hours: 36

Name of the Course: Advanced Data Structures using Python	
Course Code: MCA204 & MCA294	Semester: 2 nd
Duration: 40 Hrs.	Maximum Marks: 100
Teaching Scheme	Examination Scheme
Theory: 3	End Semester Exam: 100
Tutorial: 1	Continuous Assessment: 100
Practical: 2	Practical Sessional Internal continuous evaluation: 100
Credit: 4+3	Practical Sessional external examination: 100
Aim:	
Sl. No.	
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:	
Sl. No.	
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:	
Sl. No.	
1.	Proficiency in one high level programming language
Course Outcome:	
CO1	On completion of this course students are expected to learn various data structures, their usages, merits and limitations.
CO2	On completion of this course students are expected to design and analyze various algorithms.
CO3	On completion of this course students are expected to do a comparative analysis among different data structures and decide on the appropriate data structure to be used in a given scenario.
CO4	On completion of this course students are expected to acquire adequate knowledge and skills to solve a real life software problem.
Relevant Links:	
Adv. DS Study Material Adv. DS NPTEL LINK Adv. DS Coursera Link Adv. DS Linkedin Learning Link	

CO-PO 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	Mapping with Industry and International Academia	Lecture Hours	Corresponding Lab Assignment
1	Non-Linear Data Structure-Tree	Basic Tree Terminologies. Different types of Trees: General Trees, Forests, Binary Tree, Binary Search Tree, Expression Tree. Tree operations on each of the trees and their algorithms with complexity analysis. Tree traversal algorithms (pre-order, post-order, in-order, level-order). Constructing a Binary Tree from traversal results. Huffman's Tree Applications of Tree	International Academia: https://web.stanford.edu/class/cs166/ AICTE-prescribed syllabus: https://makautexam.net/aicte_details/Syllabus/MCA/sem221.pdf Industry Mapping: Unstructured Data [Trie (Prefix Tree), Splay Trees, Suffix Trees]	8	1. Implementation of Binary Tree and its various operations in Python 2. Traversing a Tree using pre-order, post-order, in-order traversal 3. Implementing Huffman tree in Python
2	Non-Linear Data Structure-Efficient Binary Tree	Binary Search Tree (BST). Operations on BST, Threaded Binary Tree, AVL Tree, B Tree, B+Tree, Trie	International Academia: https://web.stanford.edu/class/cs97si/03-data-structures.pdf AICTE-prescribed syllabus: https://makautexam.net/aicte_details/Syllabus/MCA/sem221.pdf Industry Mapping: Unstructured Data [Trie (Prefix Tree), Splay Trees, Suffix Trees]	6	1. Implementation of Binary Search Tree and its various operations in Python 2. Implementation of AL Tree in Python 3. Implementation of B Tree/B+ Tree in Python
3	Non-Linear Data Structure-Graph	Graph Terminology, Types of Graphs (Undirected, Directed), Representation of graphs (Adjacency Matrix, Adjacency List), Graph Traversal Algorithms (BFS-Breadth First Search, DFS-Depth First Search), Application of Graphs	International Academia: https://web.stanford.edu/class/cs166/ AICTE-prescribed syllabus: https://makautexam.net/aicte_details/Syllabus/MCA/sem221.pdf	8	1. For a given graph, to identify all simple paths from one vertex to another vertex 2. Calculate the degree of a given node in a graph 3. From a given Adjacency matrix, to

			Industry Mapping: Shortest Path Algorithms (Dijkstra's, Bellman-Ford)		construct the weighted or unweighted graph (as the case may be) 4. From a given Adjacency list, to construct the weighted or unweighted graph (as the case may be) 5. To write program to implement BFS- Breadth First Search 6. To write program to implement DFS- Depth First Search
4	Shortest path Algorithms	Shortest Path Algorithms-Minimum Spanning Tree (MSP), Kruskal's Algorithm, Prim's Algorithm, Dijkstra's Algorithm, Warshall's Algorithm, Modified Warshall's Algorithm	International Academia: https://web.stanford.edu/class/cs166/ AICTE-prescribed syllabus: https://makautexam.net/aicte_details/Syllabus/MCA/sem221.pdf Industry Mapping: Shortest Path Algorithms (Dijkstra's, Bellman-Ford)	8	1. To write program to implement Minimum Spanning Tree (MSP) 2. To determine shortest path using Dijkstra's Algorithm
5	Hashing	Hashing- definition, Hash Tables, Different Hash Functions (Division, Multiplication, Mid-Square, Folding), Collision, Collision Resolution Techniques in Hashing by Open Addressing (Linear Probing, Quadratic Probing, Double Hashing, Rehashing), Chaining, Application of Hashing	International Academia: https://web.stanford.edu/class/cs166/ AICTE-prescribed syllabus: https://makautexam.net/aicte_details/Syllabus/MCA/sem221.pdf Industry Mapping: Universal Hashing, Cuckoo Hashing	6	1. To calculate Hash Functions using various techniques Functions (Division, Multiplication, Mid-Square, Folding) 2. To resolve Collision using various Open Addressing techniques (Linear Probing, Quadratic Probing, Double Hashing, Rehashing),

List of Books Text Books:			
Name of Author	Title of the Book	Edition/ISSN/ISBN	Name of the Publisher
Tenenbaum	Data Structure Using C & C++	2 nd Ed	PEI
Reference Books:			
Kruse, Tondo & Leung	Data Structures & Program Design in C	2 nd Ed	PHI
Loudan	Mastering Algorithms With C		SPD/O'REILLY
Radhaganesan	C and Data Structures		Scitech Publications
Reema Thareja	Data Structure Using C	s2 nd Ed	Oxford

Syllabus for MCA Admission Batch 2026, 2nd Semester

Subject Name: Artificial Intelligence and Machine Learning

Credit: 04

Subject Code: MCA205 & MCA295

Lecture Hours: 40 Hrs.

Name of the Course: Artificial Intelligence and Machine Learning	
Course Code: MCA205 & MCA295	Semester: 2nd
Duration: 40 Hrs.	Maximum Marks: 100
Teaching Scheme	Examination Scheme
Theory: 4	End Semester Exam:100
Tutorial: 1	Continuous Assessment:100
Practical: 2	Practical Sessional internal continuous evaluation:100
Credit:4+3	Practical Sessional external examination:100

Aim:	
Sl. No.	
1	To gain knowledge of various aspects of artificial intelligence
2	To enhance the ability to identify qualities of a good solution of AI, ML etc.
3	To implement learned analytical techniques and AI & ML to solve problems.
Objective:	
Sl. No.	
1	Provide you with the knowledge and expertise to become a machine learning expert.
2	Demonstrate an understanding of intelligence and machine learning concepts that are vital for real-life problems.
3	Produce Python code to analyze different problems.
4	Critically evaluate intelligent problems based on their design, development and use in real-time.
Pre-Requisite:	

Sl. No.	
1.	Proficiency in Mathematics, Algorithms and Programming related to AI, ML etc.

Course Outcome:	
CO1.	Explain how knowledge is collected, created and managed for intelligence.
CO2.	Understand the key concepts in AI including their real-world applications and the toolkit used by intelligent systems.
CO3.	Implement AI & ML techniques using knowledge and toolkits.
CO4.	To enable systems to solve complex problems and automate decision-making processes.

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

Module number	Topic	Sub-topics	Mapping with Industry and International Academia	Lecture Hours	Corresponding Lab Assignment
1	Module I: Introduction to AI(2L)	Overview of Artificial Intelligence – Introduction–History of AI– Applications of AI– Objectives of AI- features of AI	International Academia: https://ocw.mit.edu/course/res-str-002-data-management-spring-2016/ AICTE-prescribed syllabus: https://makautexam.net/aicte_details/CourseStructure/MCA21.pdf	4	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

			Industry Mapping: The concepts delivered are in sync with the industry standards		reverse order. - Implement python script to accept line of text and find the number of characters, number of vowels and number of blank spaces in it. - 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	Module II: Symbolic Logic(6L)	Normal Forms in Propositional Logic – Logical Consequences – Resolution Principal – Predicate Calculus – Well Formed Formulas – Clausal Form – Rules of Inference – Unification – Resolution	International Academia: https://ocw.mit.edu/course/s/15-062-data-mining-spring-2003/ AICTE-prescribed syllabus: https://makauteam.net/aicte_details/CourseStructure/MCA21.pdf Industry Mapping: The concepts delivered are in sync with the industry standards	4	- 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'). - Write Python script to copy file contents from one file to another. - Implement a python script to check the element is in the list or not by using Linear search & Binary search. - Implement a python script to arrange the elements in sorted order using Bubble, Selection, Insertion and Merge sorting techniques. - Write a python program by using exception handling mechanism. - Write a python program to perform various database operations (create, insert, delete, update).

3	Module III: Search Techniques (6L)	Different types of search techniques and comparison among them	International Academia: https://ocw.mit.edu/course/s/15-071-the-analytics-edge-spring-2017/pages/visualization/ AICTE-prescribed syllabus: https://makautexam.net/aicte_details/CourseStructure/MCA21.pdf Industry Mapping: The concepts delivered are in sync with the industry standards	6	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.
4	Module IV: Knowledge Representation (8L)	Procedural versus declarative knowledge, forward versus backward reasoning, Structured Knowledge: Graphs, Frames, and Related Structures, Object-Oriented Representations, Representing knowledge in an uncertain domain, the semantics of Bayesian networks, Dempster-Shafer theory, Fuzzy sets & fuzzy logics	International Academia: https://prolearn.mit.edu/data-science-and-big-data-analytics-making-data-driven-decisions AICTE-prescribed syllabus: https://makautexam.net/aicte_details/CourseStructure/MCA21.pdf Industry Mapping: The concepts delivered are in sync with the industry standards	4	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.
5	Module V: Expert systems(2L)	Characteristic features of expert systems Applications, importance of expert systems Rule based system architectures (the knowledge base, the inference process, explaining how or why, building a knowledge base, the I/O interface)	International Academia: AICTE-prescribed syllabus: https://makautexam.net/aicte_details/CourseStructure/MCA21.pdf Industry Mapping: The concepts delivered are in	4	

			sync with the industry standards		
6	Module VI: Machine Learning (10L)	Computational learning tasks for predictions, learning as function approximation, generalization concept. - Linear models and Nearest-Neighbors (learning algorithms and properties, regularization). - Neural Networks (MLP and deep models, SOM). - Probabilistic graphical models. - Principles of learning processes: elements of statistical learning theory, model validation. - Support Vector Machines and kernel-based models. To implement linear regression, Data classification, Data clustering – To learn how to create segments based on similarities using K-Means and Hierarchical clustering	International Academia: https://professionalprograms.mit.edu/online-program-internet-of-things/ AICTE-prescribed syllabus: https://makautexam.net/aicte_details/CourseStructure/MCA21.pdf Industry Mapping: The concepts delivered are in sync with the industry standards	4	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. 3. Write a program to implement k-Nearest Neighbor algorithm to classify the iris data set. 4. Write a program to implement k-Means clustering algorithm to cluster the set of data stored in .CSV file 5.
7.	Module VII: Introduction to Robotics(4L)	Definitions, illustration of application domains -Mechanics and kinematics of the robot- Sensors for robotics -Robot Control-Architectures for controlling behavior in robots- Robotic Navigation-Tactile Perception in humans and robots- Vision in humans and robots.	International Academia: https://ocw.mit.edu/course/s/6-034-artificial-intelligence-fall-2010/ AICTE-prescribed syllabus: https://makautexam.net/aicte_details/CourseStructure/MCA21.pdf Industry Mapping: The concepts delivered are in sync with the industry standards	4	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.

List of Books Text Book			
Name of Author	Title of the Book	Edition	Name of the Publisher
Rich, Knight	Artificial Intelligence	3 rd edn	PHI
Patterson	Introduction to Artificial Intelligence and Expert Systems	2 nd edn	Pearson
Akerkar	Introduction to Artificial Intelligence	2 nd edn	PHI
Alpaydin	Introduction to Machine Learning	3 rd edn	PHI

University of Engineering and Management

Institute of Engineering & Management, New Town Campus

Syllabus for MCA Admission Batch 2026, 2nd Semester

Subject Name: General Studies & Current Affairs – II

Credit: 1

Subject Code: MCA(GS)201

Lecture Hours: 48

Module number	Topic	Sub-topics	Mapping with International/ National/ State Level Exams	Lecture Hours	Corresponding Assignment
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. Basic economics -Types of Economy, Feature of Indian Economy (BECC-101, Block-1, Unit-1, Unit-2, Unit-3) - HDI (BECC111, Block-2 http://egyankosh.ac.in/handle/123456789/81256) 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) - Liberalisation, Privatisation and Globalisation (LPG)(IGNOU,	International Exams GRE (https://www.ets.org/pdfs/gre/gre-math-review.pdf) GMAT (https://downloads.mba.com/downloads/gmat-handbook.pdf) National Exams: UPSC Civil Services Exam (https://upsc.gov.in/sites/default/files/Notif-CSP-23-engl-010223.pdf), pg. 25-26 UPSC Combined Defense Services (https://upsc.gov.in/sites/default/files/Notif-CDS-I-Exam-2023-Engl-211222.pdf), pg 20-21 RBI Grade B (https://rbidocs.rbi.org.in/rdocs/Content/PDFs/DADVTGRB09052023FA65E4FB1C2C)	48	Basic economics: Compose an assignment on “Impact of COVID-19 Pandemic on Human Development in India”- analyze how the pandemic has affected healthcare, education, employment and socioeconomic disparities in the country. Sectors of the economy and their analysis: Group Discussion on “ Impact of LPG Reforms and FDI on Indian Industrial Sector: Economic Transformation, Challenges and Growth Prospects RBI & Its Function: Group discussion on “Analyzing India’s Monetary Policy: Implication, Challenges and Economic Stability

		BECC-114, Block- 6) http://egyankosh.ac.in/handle/123456789/90547 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/123456789/89589 6. Budget (Union, Railway), Concept of revenue, expenditure & different types of deficit. (BECC-109, Block- 3, Unit-9) http://egyankosh.ac.in/handle/123456789/76561	F473396B4FD7E5F69CDD E.PDF), pg 22-23 4. IBPS Probationary officer(https://www.ibps.in/wp-content/uploads/Detailed-Advt.-CRP-PO-XII.pdf) , Pg 7. 5. Combined Graduate Level conducted by SSC (https://ssc.nic.in/SSCFileServer/PortalManagement/UploadedFiles/notice_CGLE_03042023.pdf) pg. 20-22 6. Intelligence Bureau ACIO (https://www.pw.live/exams/wp-content/uploads/2023/11/TB-ACIO-Recruitment-2023-Notification-Emp-News.pdf) 7. XAT (https://xat.org.in/xat-syllabus/) 8. GATE (https://gate2026.iisc.ac.in/papers-and-syllabus/) 9. CAT https://iimcat.ac.in/per/g01/pub/756/ASM/WebPortal/1/index.html?756@@1@@1 State Level Exams: 1. Civil Services Executive Exam (WBCS) (https://wbpsc.gov.in/Download)	4. Budget: An assignment on the “Understanding India’s Union Budget: A Comprehensive Analysis ** All the assignments are in line with entrance exams for premier B-Schools and GS Paper-I of UPSC CSE.
--	--	--	--	---

References

1. Indian Economy-Ramesh Singh

The detailed syllabus of the MCA 2nd Year will be published soon.