ASSAM SCIENCE AND TECHNOLOGY UNIVERSITY

Guwahati

Course Structure and Syllabus
(From Academic Session 2020-21 onwards)

MASTER OF COMPUTER APPLICATIONS (MCA)

1st Semester

ASSAM SCIENCE AND TECHNOLOGY UNIVERSITY

Guwahati

Course Structure
(From Academic Session 2020-21 onwards)

MASTER OF COMPUTER APPLICATIONS (MCA)
1st Semester

Sl. No.	Sub-Code	Subject	Hours per Week			Credit	Marks	
			L	T	P	C	CE	ESE
Theory								
1	MCA202101	Computer Organization and Architecture	3	0	0	3	30	70
2	MCA202102	Design and Analysis of Algorithms	3	1	0	4	30	70
3	MCA202103	Programming Techniques using Python and Java	3	1	0	4	30	70
4	MCA202104	Advanced Database Systems	3	0	0	3	30	70
5	MCA202105	Operating Systems	3	1	0	4	30	70
Practical								
1	MCA202116	Laboratory- Java and Assembly Language Programming	0	0	8	4	30	70
Bridge Courses [For Non-Computer Students]								
Theory								
1	MCA202B106	Introductory Programming and Data Structures using C	3	1	0	0	-	100
2	MCA202B107	Mathematical Foundation of Computer Science	3	1	0	0	-	100
3	MCA202B108	Fundamentals of Computer Systems and Networking	3	1	0	0	-	100
Practical								
1	MCA202B116	Laboratory- C and Data Structures	0	0	8	0	-	100
TOTAL			**15/24**	**3/6**	**8/16**	**22**	**180**	**420/820**
Total Contact Hours per week: 26/46								
Total Credits: 22								

Detail Syllabus:

Course Code	Course Title	Hours per week L-T-P	Credit C
MCA202101	**Computer Organization and Architecture**	**3-0-0**	**3**

MODULE 1: CPU Architecture (15 Lectures)

Instruction format - operand addressing formats – three, two one and zero address instructions; Instruction set selection, Instruction types: data transfer, data manipulation and program control; Addressing modes - direct, indirect, immediate, relative, indexed etc. Instruction execution process - fetch and execution cycles; data path organization – single and two bus; control structure: hardwired and micro-programmed; control steps in different instruction execution, Reduced instruction set computer (RISC), CISC and RISC characteristics, block diagram and pin diagram of 8085, use of registers in assembly language programs, assembly language programming.

MODULE 2: Computer Arithmetic (7 Lectures)

Review of addition and subtraction with signed magnitude and 2's complement data, hardware implementation, Multiplication algorithm, Hardware implementation, hardware algorithm, Booths multiplication algorithm, Array multiplier, Division basic, Floating point arithmetic.

MODULE 3: I/O Architecture (15 Lectures)

Characteristics of simple I/O devices, their controllers; I/O interface – addressing: memory mapped and isolated I/O, data transfer: Synchronous and Asynchronous data transfer, types of asynchronous data transfer: strobe control, handshaking. Modes of data transfer: program controlled, interrupt initiated and DMA data transfer; polled and interrupt controlled synchronization; Interrupt mechanism - device identification - polling, vectored; priority schemes - daisy chaining, interrupt masking; Concept of DMA - cycle stealing and burst mode, DMA interface bus arbitration mechanism; Concept of I/O channels and peripheral processors.

MODULE 4: Advanced Memory Concepts (8 Lectures)

Memory hierarchies, Cache memory- Mapping techniques, Virtual memory- address space, memory space, address mapping using pages.

Textbooks:

1. Mano M.M: Computer system Architecture, PHI (EEE)
2. Hamacher, Vranesic and Zaky: Computer Organization, TMGH

References Books:

1. William Stallings, Computer Organization and architecture, Pearson
2. Hennessey: Computer Architecture, Elsevier
3. Stallings: Computer Organization & Architecture, PE
4. Hayes : Computer Architecture & Organization, MGH

Course Code	Course Title	Hours per week L-T-P	Credit C
MCA202102	Design and Analysis of Algorithms	3-1-0	4

MODULE 1: Introduction (8 Lectures)

Algorithms, Analysis of Algorithms, Design of Algorithms and complexity of Algorithms, Asymptotic Notations, Growth of Function, Recurrences Sorting in Polynomial Time: Insertion Sort, Merge Sort

MODULE 2: Design and Analysis Techniques (10 Lectures)

Divide and Conquer, merge sort, finding closest pair of points. Augmented Data Structure- Red-Black tree, OS-Tree, Interval Tree, B-Tree, AVL tree

MODULE 3: Greedy Algorithms (10 Lectures)

Coin charging, Kruskal's, Prim and Dijkstra's algorithm, Knapsack problem.

Dynamic Programming:

Coin charging problem, matrix multiplication, longest common subsequence, Floyd and War shall algorithm. Application of Sorting and Searching

MODULE 4: Graph Algorithms (6 Lectures)

Topological sort, minimum spanning trees, shortest paths, maximum-flow – Flow networks, Ford-Fulkerson method, Maximum bipartite matching.

MODULE 5: Concept of different Problem classes and Introduction to Approximation Algorithms (6 Lectures)

P, NP, NP-Complete, Easy vs Hard, Polynomial time, non-deterministic algorithms, reducibility.

Approximation Algorithms:

Traveling salesman problem, Parallel and Distributed algorithms.

Text Books/ References Books:

1. Introduction to Algorithm by Thomas Corman , CE Leiserson Roland L Rivest, C Stein
2. Fundamentals of Algorithms - Ellis Horowitz, S Salini S. Rajasekaran

Course Code	Course Title	Hours per week L-T-P	Credit C
MCA202103	Programming Techniques using Python and Java	3-1-0	4

PART 1

MODULE 1: Introduction to Java (7 Lectures)

Basics of Java programming, Data types, Variables, Operators, Control structures including selection, Looping, Java methods, Overloading, Math class, Arrays in java.

Objects and Classes : Basics of objects and classes in java, Constructors, Finalizer, Visibility modifiers, Methods and objects, Inbuilt classes like String, Character, this reference.

MODULE 2: Inheritance and Polymorphism (8 Lectures)

Inheritance in java, Super and sub class, Overriding, Object class, Polymorphism, Dynamic binding, Generic programming, Casting objects, Instance of operator, Abstract class, Interface in java, Package in java, UTIL package.

MODULE 3: Event and GUI programming (10 Lectures)

Event handling in java, Event types, Mouse and key events, GUI Basics, Panels, Frames, Layout Managers: Flow Layout, Border Layout, Grid Layout, GUI components like Buttons, Check Boxes, Radio Buttons, Labels, Text Fields, Text Areas, Combo Boxes, Lists, Scroll Bars, Sliders, Windows, Menus, Dialog Box, Applet and its life cycle, Introduction to swing.

PART II

MODULE 4: Introduction to Python Programming Language (2 Lectures)

Built in Functions, Conversions, Numeric Data Types, String Operators, String Slices, String Operations, String Values, Naming Conventions, IDLE, Dynamic Types, Strengths and Weaknesses, Introduction to Python Language.

MODULE 5: (3 Lectures)

The while Loop, break and continue, Bit Wise Operators, True or False, Operators, Logical, Relational Operators, the if Statement, Indenting, Control Flow and Syntax, Introduction, Data Collections and Language Component: Copying Collections. Sorting Dictionaries, Dictionaries, Sets, Tuples, the for Loop, Lists

MODULE 6: (7 Lectures)

Classes in Python, Object and Classes: Principles of Object Orientation, File Organization, Instance Methods, Creating Classes, 12 25 Custom Exception Classes, Type Identification, Polymorphism, Inheritance, Class Variables, Special Methods

MODULE 7: (8 Lectures)

The dir Function, Standard Modules – time, Standard Modules – math, Standard Modules – sys, Modules, Lambda, Mapping Functions in a Dictionary, Passing Functions to a Function, Functions - "First Class Citizens", Scope, Variable Number of Arguments, Passing Collections to a Function Keyword and Optional Parameters, Function Documentation, Parameters, Defining Your Own Functions, Introduction, Functions and Modules

Text Books/ References Books:

1. Dive into Python, Mike
2. Learning Python, 4th Edition by Mark Lutz
3. Programming Python, 4th Edition by Mark Lutz

4. Introduction to Java Programming (Comprehensive Version), Daniel Liang, Seventh Edition, Pearson.
5. Programming in Java, Sachin Malhotra & Saurabh Chaudhary, Oxford University Press.
6. Murach's Beginning Java 2, Doug Lowe, Joel Murach and Andrea Steelman, SPD.
7. Core Java Volume-I Fundamentals, Eight Edition, Horstmann & Cornell, Pearson Education.
8. The Complete Reference, Java 2 (Fourth Edition), Herbert Schild, TMH. 6 Java Programming, D. S. Malik, Cengage Learning

Course Code	Course Title	Hours per week L-T-P	Credit C
MCA202104	**Advanced Database Systems**	**3-0-0**	**3**

MODULE 1: Introduction to Relational Database Design (3 Lectures)
What is DBMS, advantage of using DBMS, data models (object based logical models, record based logical models), DBMS users, overall system structure Hierarchical and network data models and related data structures

MODULE 2: Entity Relationship Models and Relational Database Design (10 Lectures)
ER diagrams, generalization, specialization, aggregation. Database models - Network model, Hierarchical model, and Relational model.
Relational database design: Underlying concepts, structure, study of relational languages (relational algebra, SQL, QBE). Integrity constraints (domain constraints, referential, assertions, triggers, functional dependencies), Normalization (using FDs, multivalued dependencies, join dependencies), Domain -key normal form

MODULE 3: Transactions (6 Lectures)
Concept, state, ACID properties, serializability and recoverability, testing for serializability. **Concurrency Control:** Lock - based, protocols, timestamp based protocols, validation based protocols, multi version schemes, deadlock handling

MODULE 4: Recovery System (3 Lectures)
Log based recovery (deferred and immediate database modification), checkpoints, shadow paging, recovery with concurrent with transactions, buffer managements in recovery, recovery from loss of non - volatile storage, logical undo logging, transaction rollback, restart recovery, fuzzy checkpointing

MODULE 5: Query Processing (4 Lectures)
Storage and file structure, file organization: disk storage systems, blocking factor, Access path: searching, indexing and hashing techniques, external sorting, transformation of relational expressions, breaking of queries into sub queries to optimize execution plan, Select, Project and Join Operations, set operations, aggregation Cost based query optimization. measurement of cost of a query considering different access path, evaluation of expressions.
Heuristic query optimizations: query tree query graph and representation of queries in query tree, Steps for heuristic query optimization, semantic query optimization.

MODULE 6: Parallel Databases (4 Lectures)
Parallel database architecture, scale up and speedup using parallel database architecture, Introductory concepts, partitioning techniques, interoperation parallelism and intra operation parallelism - Parallel Sort algorithms (range partitioning sort, parallel external sort-merge), Parallel Join Algorithms (partitioned join, fragment-and-replicate join, parallel hash join), interoperation parallelism (pipelined, independent). Example of parallel databases

MODULE 7: Distributed Databases (6 Lectures)
Replication and fragmentation, network transparency, join processing, distributed transaction processing, two-phase and three-phase commit protocols, handling failure, coordinator selection, concurrency control (locking, timestamping), deadlock handling (centralized, fully distributed), multi-database systems.

MODULE 8: Security and Integrity (4 Lectures)

Violations, authorization, views, privileges, granting privileges, discretionary and mandatory access control mechanism, Bell La-podulas Security Access Control Mechanism, designing databases using mandatory access control mechanisms, security specification in SQL.

Text Books/ References Books:

1. Elmasari and Navathe, Fundamentals Of Database System, Narosa Publishing Company, 1989.
2. J.D. Ullman, principles of Database Systems, Galgotia Publishing Private Limited.
3. Silberschats, Kroth and Sudershan, Principles of Database Systems, McGraw Hill Publication.
4. 4. C.J. Date, An Introduction to Database Systems, Vol-I and Vol-II, Addison-Wesley Publishing Company.

Course Code	Course Title	Hours per week L-T-P	Credit C
MCA202105	Operating Systems	3-1-0	4

MODULE 1: Introduction and Historical Perspective (3 Lectures)

Operating System concept and it's role, Batch processing, multiprogramming, time-sharing, distributed system, Functions, components and structure of an operating system

MODULE 2: Process Management (18 Lectures)

Structure of a process, PCB, operations on processes, Support for concurrent processes-Trade-off sequential and concurrent processing with examples in implementation Shared data, Critical sections, Mutual exclusion, busy form of waiting, lock and unlock primitives, semaphore, block and wakeup, Producer-consumer problem, multiple producer and consumer and synchronization, Dinning Philosopher's problem, monitors, Starvation problem in scheduling and Priority inversion.

Inter process communication mechanisms and primitives. Communication mechanisms in Client-Server system.

Threads:

Multithreading Models, Threading issues, kernel and user thread implementations Pthreads, thread affinity.

Scheduling:

Process states, context switching, schedulers, scheduling criteria, types of scheduling algorithms, Implementation of concurrency Primitives.

System deadlock: Causes of deadlock and deadlock handling strategies: prevention, detection and avoidance

MODULE 3: I/O Systems (3 Lectures)

I/O Management device controller, Device drivers, I/O Software goals and structure, Interrupt and handling mechanisms, Application of I/O Interface, Terminal handling, Block and character devices. Vectored I/O

MODULE 4: Memory Management (7 Lectures)

Address space management: address binding, logical vs physical address space, Static and dynamic memory management protection and sharing, Contiguous and non - contiguous memory allocation, fragmentation and solution; Swapping, Paging and Segmentation, page table; page replacement and space allocation policies. Combined paging and segmentation, Virtual memory, Demand Paging and performance assessment; Page replacement policies, Thrashing, Working set model, Kernel Memory Allocation.

MODULE 5: File System and Disk Management (6 Lectures)

File concept and organization, File management strategies, File system structure and access methods, tradeoffs, Directory structures, Allocation Methods: contiguous, linked, indexed, FAT and I-node structures, File system protection, Security, Integrity, Device independence. Protection domain and protection matrix. File Recovery, backup and restore.

Mass storage device structure, Disk management and disk scheduling algorithms.

MODULE 6: Case study of Operating Systems (4 Lectures)

Concepts of Distributed and Network Operating System. Fundamental concepts on Embedded/Real time operating systems. Eg Rt LINUX

Text Books:

1. Silbersehatz A., Galvin P.B. and Gagne G: "Operating System Concepts", Wiley, 9th Ed.
2. Milenkovic M., "Operating System - Concepts and Design", McGraw Hill.
3. Tanenbaum A.S.: "Operating System - Design and Implementation", PHI **(**EEE).
4. Bach, M.: "Design of the UNIX operating system", PHI(EEE).

Reference Books:

1. Stalling, William, "Operating Systems", Maxwell McMillan International Editions, 1992.
2. Dietel, R.N., "An Introduction to Operating Systems", Addison Wesley
3. Walia Ekta, "Operating System Concepts", Khanna Book Publishing and Co.

Course Code	Course Title	Hours per week L-T-P	Credit C
MCA202116	Laboratory- Java and Assembly Language Programming	0-0-8	4

SL. No.	PART-1: STATEMENT OF THE PROBLEM (40%)
1	Write a Java Program to define a class, describe its constructor, overload the Constructors and instantiate its object
2	Write a Java Program to define a class, define instance methods and overload them and use them for dynamic method invocation
3	Write a Java Program to demonstrate use of sub class
4	Write a Java Program to demonstrate use of nested class
5	Write a Java Program to implement inheritance and demonstrate use of method overriding
6	Write a Java Program to implement multilevel inheritance by applying various access controls to its data members and methods
7	Write a program to demonstrate use of implementing interfaces
8	Write a program to demonstrate use of extending interfaces
9	Write a Java program to implement the concept of importing classes from user defined package and creating packages
10	Write a program to implement the concept of threading by extending Thread Class
11	Write a program to implement the concept of threading by implementing Runnable Interface
12	Write a program to implement the concept of Exception Handling using predefined exception
13	Write a program to implement the concept of Exception Handling by creating user defined exceptions
14	Write a program using Applet to display a message in the Applet
15	Write a Java Program to demonstrate Keyboard event
16	Write a Java Program to demonstrate Mouse events
17	Write programs for using Graphics class • to display basic shapes and fill them • draw different items using basic shapes • set background and foreground colors
18	Write a Menu Driven Program in Java which performs sorting of a group of integer values using bubble sort method
19	Write a class called “Student” with a) Instance variable roll_no, name, semester and branch b) A method called “setdata ()” which takes 4 parameters and returns void. This method must contain statements to set the values of instance variables of each object of” Student” class c) A method called “calculate ()” which does not have any input parameter and returns a double value. This method must contain statements to take input for marks of 5 subjects (Physics, Chemistry, Mathematics, English and Hindi) and calculate the percentage of marks using formula: Percentage marks= (Total marks/500) *100 The method must also contain a return statement to return the calculated percentage value d) A method called “showdata ()” which does not take any parameter and returns void. This method must contain statement to display the values of roll no, name, semester, branch and percentage of mark of each object of “Student” class Write another class “StudentRecord” which contains the main () method. Inside the main() method write statements to create 3 different object of “Student” class and display the values of their instance variable with the calculated percentage of marks for each of them using Menu
20	Write a class called “Employee” with a) Instance variables Employee code, Employee name and Basic salary

	b) A parameterized constructor which contains statements to set the values of instance variables of each object of "Employee" class c) A method called "calculate ()" which does not have input parameter and it returns a double value. This method contains statements to calculate the gross salary of an Employee object using the formula HRA= 60% of Basic salary DA= 98% Basic salary Gross Salary= Basic Salary + HRA + DA This method must contain a return statement to return the calculated value of Gross salary d) A method called "showdata ()" which does not take any parameter and return void. This method must contain statements to display the values of instance variables and calculated gross salary of each object of "Employee" class Write another class "EmployeeRecord" which contains the main () method. Inside the main() method write statements to create 3 different object of "Employee" class and display the values of their instance variable with the calculated gross salary for each of them using Menu
PART-2: PYTHON PROGRAMMING (30%)	
1	Write a Program in Python to convert Celsius to Fahrenheit
2	Write a Program in Python to Check Leap Year
3	Write a Program in Python to Find the Factorial of a Number
4	Write a Program in Python to Print the Fibonacci sequence
5	Write a Program in Python to Convert Decimal number to a) Binary b) Octal c) Hexadecimal
6	Write a Program in Python to Sort Words in Alphabetic Order
7	Write a Program in Python to print the largest and smallest element in an array
8	Write a Program in Python to Count Even and Odd Numbers in a List
9	Write a Program in Python to check if a given key exists in a Dictionary
10	Write a Program in Python Program to Merge Two Dictionaries Example
11	Write a Program in Python Program to Print Floyd's Triangle using For Loop
12	Write a Program in Python to Check Whether a String is Palindrome or Not
13	Write a Program to Create a Class and Object in Python
14	Write a Program in Python to demonstrate single level inheritance in Python
15	Write a Program in Python to demonstrate multiple inheritance in Python
16	Write a Program in Python to demonstrate Operator Overloading in Python
17	Write a Program in Python to calculate the factorial of a number using recursion
18	Write a Program in Python to Multiply all numbers in the list (4 different ways)
19	Write a Program in Python to Join Tuples if similar initial element
20	Write a Program in Python to perform Insertion sort
PART-3: ASSEMBLY LANGUAGE PROGRAMMIMG (30%)	
1	Write a program to find the sum of two BCD numbers stored in memory
2	Write a program, which will read two decimal numbers, then multiply them together, and finally print out the result (in decimal)
3	Write a program to convert the ASCII code to its BCD equivalent
4	Write a program, which will read in two decimal inputs and print out their sum, in decimal
5	Write a program, which will read in two decimal inputs and print out the smaller of the two, in decimal

6	Write a program to calculate the average of three given numbers stored in memory
7	Write a program in 8086 assembly language to find the volume of sphere using following formula: V = 4/3π r3
8	Write a program to evaluates 3 * (x^3) + 4x + 5 if flag = = 1 or evaluates 7x + 8 if flag== 0. Assume x is a 16-bit unsigned integer
9	Write a program to convert Centigrade (Celsius) to Fahrenheit temperature measuring scales. Using formula: Celsius = (Fahrenheit - 32) * 5 / 9
10	Write a Program which adds the sales tax in the Price list of items and replace the Price list with a new list
11	Write a program to find the factorial of decimal number given by user
12	Write a program to find n Cr for a given n and r
13	Write a program to arrange given N numbers in descending order
14	Write a program, which will read in decimal inputs repeatedly until a zero value is read; at this point, it should print out the sum of the numbers read in so far
15	Develop and execute an assembly language program to find the LCM of two 16-bit unsigned integers
16	Develop and execute an assembly language program to find the HCF of two unsigned 16-bit numbers
17	Write a program for finding the largest number in an array of 10 elements
18	Develop and execute a program to sort a given set of 8-bit unsigned integers into ascending order
19	Develop and execute an assembly language program to sort a given set of 16- bit unsigned integers into descending order
20	Write a Program which adds the sales tax in the Price list of items and replace the Price list with calculated values

Bridge Courses [For Non-Computer Students]

Course Code	Course Title	Hours per week L-T-P	Credit C
MCA202B106	**Introductory Programming and Data Structures using C**	**3-1-0**	**0**

MODULE 1: Introduction to Computers (6 Lectures)
Introducing and Interacting with Computers, Computer, organization, Number System & Computer codes, Computer Arithmetic, Boolean Algebra and IO Devices

MODULE 2: Introduction to Memory and Languages (6 Lectures)
Processor and Memory, Types of Storage Devices, Computer Software and types, Basics of Programming, Programming Languages. Language Elements, Algorithms and Flowcharts

MODULE 3: Problem Solving with C Programming (10 Lectures)
History, Execution of C Program, Constants, Variables and Keywords, Data types, Expressions, constants, variables, Operators, Operator Precedence and associativity, data input and output, Formatted Console I/O Functions, Conversion Specifications, assignment statements, conditional statements, Looping Statements, Storage Classes

MODULE 4: Array and Modular Programming (6 Lectures)
Introduction to Function, Functions with Simple Output Parameters, Passing Values between Functions, Multiple Calls to a Function, Parameter Passing by Value v/s Parameter Passing by Reference, Recursion **Arrays**: Declaring and Referencing Arrays, Array Subscripts, Using for Loops for Sequential Access, Multidimensional Arrays, Passing arrays as arguments

MODULE 5: Structures, Unions, Strings and Pointers (6 Lectures)
Structures & Unions- definition, Processing structures – Passing structures to a function. Pointers: Operations on Pointers – Pointers to Functions, Functions Returning Pointers, Arrays of pointers. String handling

MODULE 6: Sorting and Searching (6 Lectures)
Different sorting techniques: Bubble sort, quick sort, insertion sort
Different sorting techniques: Linear search, Binary Search, hashing

MODULE 7: Linear Data Structure (6 Lectures)
Stack, Queue, Linked List various operations and application**, Nonlinear Data Structure:** Binary Tree, Binary Search Tree, AVL Tree, B Tree, Graph, Depth First Search, Breadth First Search

Text Books:

1. Peter Norton," Introduction to Computers", 6th Edition,2009.
2. Yashvant Kanetkar,"Let Us C", BPB Publications, 13th edition, 2012.
3. S Prasad, K.R Venugopal, "Mastering C", Tata McGraw Hill, 2006.
4. E.Balaguruswamy, "Programming in ANSI C", Tata McGraw Hill, 6th edition, 2012

Reference Books:

1. Pradeep K Sinha, PritiSinha, "Computer Fundamentals", 6th Edition, 2003.
2. Bayron Gottfried, "Schaum's Outline of Programming with C", 4th Edition, 2018 (Paper Back).
3. Kernighan and Ritchie, "The C Programming Language", Prentice Hall, 2015 (Paper Back).

Course Code	Course Title	Hours per week L-T-P	Credit C
MCA202B107	Mathematical Foundation of Computer Science	3-1-0	0

MODULE 1: Sets, Functions and Relations (7 Lectures)

Set: Set, subset, power set, properties of set, operations on sets, products on set, representation of set in computer partitions.

Functions and relation: Binary relation, types of relation, equivalence relation and equivalence class, closure of relation, function, different types of functions.

MODULE 2: Matrices and Operations (5 Lectures)

Matrices and types, Matrix arithmetic, Matrix transpose, Determinants, Difference between matrix and determinant, Determinant of a matrix, Matrix Inversion

MODULE 3: Mathematical Logic and Mathematical Induction (4 Lectures)

Truth tables, algebra of propositions, conditional proposition, logical arguments, principle of mathematical induction

MODULE 4: Introductory Concept on Formal logic (7 Lectures)

Revision of propositional and first orders predicate logic, Normal forms: conjunctive normal form, clausal form. Premises, arguments and hypothesis, testing validity, logic of proof and inference rules, predicate calculus, Logic programming: Introduction to Prolog

MODULE 5: (6 Lectures)

Combinatorics: Fundamental principle, Factorial notation, permutation and combination

Principle of counting: Principle of inclusion and exclusion, pigeonhole principle

Group Theory: Binary operation and laws, Algebraic structure, Group and properties, Order of an element in group

Recurrence Relations and Generating Function: recurrence relation, solving recurrence relation by substitution and generating functions, Characteristics roots solution of homogeneous recurrence relation

MODULE 6: Probability and Statistical Concepts (14 Lectures)

Probability Theory: Sample spaces; Events and probability; Discrete probability; Union, intersection and compliment of events; conditional probability; Bayes Theorem, Random Variables and Distribution: Random variables, Discrete Probability Distribution - Binomial, Poisson, Density functions and Distributions functions; Continuous probability distribution - Normal, Student's t & x
(chi - square) statistic, Large sample tests for mean and proportion, Moments and Moment generating functions: Linear correlation coefficient: Multiple correlation.

Mathematical Expectations: Expectations, variance and co - variance, Addition and Multiplication theorem of Expectation

Text Books:

1. Trembly,Manohar, "Discrete Mathematical Structures"
2. J.E.Hopcroft & J.D. Ullman, "Introduction to Automata Theory, Languages & Computations", Narosa Publishing House 1999
3. Dr. S.K Sarkar, "A Text Book of Discrete Mathematics", S Chand and Company.
4. Mishra, K.L.P, Chandrasekaran, N., "Theory of Computer Science", PHI.

References:

1. S Santha, "Discrete Mathematics with Combinatorics and Graph Theory", Cengage Learning
2. Liu, C.L and Mahapatra, D.P, "Elements of Discrete Mathematics", Tata McGraw Hill.
3. C.K Nagpal, "Formal Language and Automata Theory", Oxford University Press.
4. Anuradha, A Puntambekar: Theory of Computation, Technical Publication.

Course Code	Course Title	Hours per week L-T-P	Credit C
MCA202B108	Fundamentals of Computer Systems and Networking	3-1-0	0

MODULE 1: Representation of Information (7 Lectures)

Number System: Binary, octal, hexadecimal. Positive and negative numbers; fixed point and floating point numbers. **Arithmetic Operations:** Addition, subtraction, etc. Character codes: ASCII and EBCDIC. Redundant coding for error detection and correction: Concept Hamming distance, parity codes, and Hamming code, block codes, cyclic redundancy code

MODULE 2: Digital Logic and Gates (15 Lectures)

Logic Design: Boolean algebra, Boolean variables and functions - canonical and standard forms, truth table, minimization of Boolean functions - Karnaugh map.

Logic Gates: AND, OR, NOT, NAND, NOR, XOR, EXOR/Equivalence

Combinational and Sequential Circuits:

Combinational Circuits: Block Diagram, Implementation of Boolean functions using logic gates; Adder – Half Adder, Full Adder, Subtractor - Half Subtractor and Full Subtractor, decoders, encoders, Multiplexers, Combinational Logic Circuit Design, simple arithmetic and logic circuits.

Sequential Circuits: Block Diagram, flip flop: RS, D, JK, T, Master Slave, Sequential Circuit Design, Shift Registers and Counters – synchronous and asynchronous

MODULE 3: Introduction to Computer Networks (8 Lectures)

Overview: Goals of networking, types, application, topologies, Standards, performance issues. Basics of digital communication, signal, noise, LAN, MAN, WAN. Networking and internetworking devices.

Network Architecture: ISO-OSI reference model, TCP/IP model, design philosophy, layer, protocol, interface, and service concepts. Layer-wise functionality

MODULE 4: Physical Layer (10 Lectures)

Basic functions and services, Concepts of data transmission, Analog and digital Transmission, Asynchronous and Synchronous transmission, bandwidth, data rate of a Channel, modulation and multiplexing methods: PCM, FDM, TDM, switching techniques (Circuit, Packet switching and message switching), modem, encoding methods, communication media.

Introduction to data link and network layers.

Text Books/References:

1. Digital Logic and Computer Design – PHI (EEE) Mano, M.M.
2. Computer Organisation and Architecture – William Stallings.
3. Tanenbaum A.S., Computer Network, PHI (EEE)
4. Forouzan, Data communication and networking, 4th Edn, TMGH

Course Code	Course Title	Hours per week L-T-P	Credit C
MCA202B116	Laboratory- C and Data Structures	0-0-8	0

Exp. No.	STATEMENT OF THE EXPERIMENT
1	Write a program to compute the GCD of three numbers.
2	Write a program to compute the LCM of three numbers.
3	Write a program to write a program which read a decimal number and convert that number to either • Binary. • Octal • Hexadecimal
4	Write a program to find the largest and smallest element in an array
5	Write a menu driven program to create an array of integers and perform the following operations • Insert an element in a specific position of the array • Delete an element from a specific position of the array • Search an element in the array • Reverse the array elements. • Sort the elements in the array • Remove Duplicate Elements from the array • Frequency of each element
6	An array A contains 5 elements whereas another array B contains 10 elements. write a function to create an array C that contain only those elements that are common to both A and B
7	Write a program to merge two sorted array, merged array is also in sorted order
8	Write a program that accepts a string and count the number of vowel , consonant , blank , digits , special characters
9	Write a menu driven program to create a two dimensional array of integers (matrix)and perform the following operations • Addition of two matrices • Multiplication of two matrices • Transpose of the matrix
10	Write a menu driven program to create a two dimensional array of integers (matrix)and perform the following operations • Row Sum • Column Sum • Sum of Diagonal Elements (for two possibilities) • Sum of Upper Triangular Element • Sum of Lower Triangular Elements
11	Write a program to find the sum of digits of a given numbers using recursion
12	Write a program to generate first n Fibonacci terms using recursion
13	Write a program using pointer to check a string is palindrome or not
14	Write a program that takes the following information of n students and print the name and roll numbers according to their performance Roll No ii) Name iii) Marks of 3 subjects hints: struct student { introllno; char name[20]; int mark[3]; }std[10];
15	Write a program that accept two times in hh:mm:ss format and added up
16	Write a program to that take roll no, name, marks of N students as input and display the information of a particular student whose name is specified by the user

17	Write a program to display the largest elements among three elements entered by the user using macro
18	Write a program that accept a string in upper case and convert the string to lower case using command line arguments
19	Enter some data in a file, place the character and integers in two different file
20	Write a program in to create a singly linked list of integers and perform the following operations • Add a node at the end of a linked list • Add a node at the beginning of a linked list • Add a node after a specified position • Count the number of nodes present in the linked list • Delete a specified node from the linked list • Reverse the linked list
21	Write a program in to create a doubly linked list of integers and perform the following operations • Add a node at the end of a linked list • Add a node at the beginning of a linked list • Add a node after a specified position • Count the number of nodes present in the linked list • Delete a specified node from the linked list • Reverse the linked list
22	Write a program in to create a circular linked list and perform the following operations • Insert a node • Delete a node
23	Write a program to read the name ,age and salary of 5 persons and maintains them in a linked list sorted by name
24	Write a program to merge two singly linked list
25	Write a program to merge two doubly linked list
26	Write a program to implement stack with two dimensional array
27	Write a program to convert an infix expression into its postfix notation
28	Write a program to evaluate a postfix expression
29	Write a program to copy the contents of one stack to another
30	Write a program to implement circular queue as an array
31	Write a program to implement circular queue as a linked list
32	Write a program to implement the followings: • Input restricted deque as an array • Output restricted deque as an array
33	Write a program to implement priority queue as a linked list
34	Write a program that uses both recursive and non recursive functions to perform the following searching operations for a Key value in a given list of integers: • Linear search • Binary search
35	Write a program to perform the following operations: • Insert an element into a binary search tree • Delete an element from a binary search tree • c) Search for a key element in a binary search tree
36	Write a program to implement the tree traversal methods
37	Write a program to perform the following operations on AVL tree: • Insert an element • Delete an element • Search an element

****