

Computer Maintenance Syllabus

Common for BA/B.Com/B.Sc Courses (Computer Maintenance as core subject)

Semester	Paper	Subject	Hrs.	Credits
SEMESTER I	I	Programming Methodology	4	4
		Programming Methodology Lab	2	2
SEMESTER II	II	Data Structures	4	4
		Data Structures using c/c++ Lab	2	2
SEMESTER III	III	Operating System	4	4
		Operating System Lab	2	2
SEMESTER IV	IV	Database Management System	4	4
		Database Management System Lab	2	2
	V	Computer Networks / Computer Organisation & Architecture	4	4
		Computer Networks Lab/ Computer Organisation & Architecture Lab	2	2

PROGRAMMING METHODOLOGY

Common for BA/B.Com/B.Sc Courses (Computer Maintenance as core subject)

Lectures: Each Unit: 12h

Total: 60h

Credits :4

LEARNING OBJECTIVES:

- To learn and develop simple algorithms and flowcharts to solve problems
- To develop problem solving skills coupled with structured programming principles..
- To provide C and C++ programming exposure.
- To introduce foundational concepts of computer programming to students of different branches of B.sc (Computer Science), B.Com (Computers), B.A (Computers)

UNIT 1. Introduction to Computer Problem Solving: Introduction to procedural, structured and object oriented programming languages, Introduction to programming and Programs concepts, Implementation of algorithms, Efficiency of Algorithms, Analysis of Algorithms, Design the Flowcharts, Requirements for solving problems by computer, Top-down Design, Program verification.

Introduction to C++ Programming, Structure of a C++ program, identifiers and keywords, constants: Numeric, Character and String constants, C++ operators: Arithmetic , Relational, Equality, Logical, Unary, Conditional, Assignment, Bitwise, Sizeof and Comma operator. Constants, Variables, Data Types

UNIT II. Decision control and Looping Statements: Decision Making and Branching: Simple if , if-else, nested if statement. Looping Statements: While , do-while, for loop, nested for loops, break and continue statements, go to label statement.

UNIT III. Functions and Arrays : Defining Function, function declaration, Types of functions, return statement, Actual and Formal arguments, Recursive function, Call- by- value, call- by- reference, Function overloading.

Arrays: Array declaration, Initialization, Accessing elements of the Array, Storing values in Array, One dimensional arrays, Two dimensional arrays.

UNIT IV. Strings and Structures: String definition, Reading and Writing strings, String operations: String length, String appending, String reverse, String comparison, String concatenation using string Library functions.

Structures: Structures declaration, Initialization of structures, Accessing members of a structures, Array of structure: Passing individual members, Passing the entire structure, structures and functions.

UNIT V. Files: Introduction to Files, opening a file, closing a file, Reading data from files: fscanf(), fgets(),fgetc(), fread(). Writing Data to Files: fprintf(), fputs(), fputc(), fwrite().

Storage class: advantages of storage class, auto, extern, register, static.

COURSE OUTCOMES:

Upon Completion of the course, the students will be able to:

- Write programs for simple applications
- Formulate algorithm for simple problems
- Analyze different data types and arrays

- Use programming language to solve problems.

TEXT BOOKS:

1. ReemaThareja : Computer Fundamental and Programming in C (Oxford Higher Education)
2. R.G. Dromey: *How To Solve It By Computer*. (PHI).
3. Object Oriented Programming with C++*Programming in ANSI C: 7:E.Balagurusamy* (Tata McGraw Hill,)
4. Yashwant Kanetkar : Let Us C++ (PHI)
5. Brian W.Kemighan& Dennis Ritchie "C Programming Language", PHI

PROGRAMMING METHODOLOGY LAB

Exercises

Credits : 2

No of hours/week : 2

1. Write a C++ program for biggest three numbers.
2. Write a C++ program to print the reverse number using while loop.
3. Write a C++ program to check whether the given number is Armstrong or not using do-while loop.
4. Write a C++ program to check whether the given number is prime number or not.
5. Write a C++ program to print any multiplication table using for loop.
6. Write a C++ program for swapping of two numbers using call-by-value
7. Write a C++ program for swapping of two numbers using call-by-reference
8. Write a C++ program to print Fibonacci series using recursion.
9. Write a C++ program for Matrix addition
10. Write a C++ program to perform various string operations .

DATA STRUCTURES

Common for BA/B.Com/B.Sc Courses (Computer Maintenance as core subject)

Lectures: Each Unit: 12h

Total: 60h

Credits :4

LEARNING OBJECTIVES:

- To introduce the fundamental concepts of Data Structures and to emphasize the importance of Data Structures in developing and implementing algorithms.
- To be familiar with the description of algorithms in both functional and procedural styles.
- To have a knowledge of complexity of basic operations like insert, delete, search and sort on these data structures.
- Ability to choose a data structure to suitably model any data used in computer applications.
- Ability to assess efficiency tradeoffs among different data structure implementations.

UNIT I. Basic concepts- Algorithm Specification-Introduction, Recursive algorithms, Data Abstraction Performance analysis, Linear and Non Linear data structures, Singly Linked Lists-Operations, Concatenating, circularly linked lists-Operations for Circularly linked lists, Doubly Linked Lists-Operations. Representation of single, two dimensional arrays, sparse matrices-array and linked representations.

UNIT II. Stack- Operations, Array and Linked Implementations, Applications- Infix to Postfix Conversion, Postfix Expression Evaluation, Recursion Implementation, Queue- Definition and Operations, Array and Linked Implementations, Circular Queues - Insertion and Deletion Operations, Dequeue (Double Ended Queue).

UNIT III. Trees, Representation of Trees, Binary tree, Properties of Binary Trees, Binary Tree Representations- Array and Linked Representations, Binary Tree Traversals, Threaded Binary Trees, Priority Queue- Implementation, Heap- Definition, Insertion, Deletion.

UNIT IV. Graph and its Representation, Graph Traversals, Connected Components, Basic Searching Techniques, Minimal Spanning Trees.

UNIT V. Sorting Methods, Comparison of Sorting Methods, Search Trees- Binary Search Trees, AVL Trees- Definition and Examples.

Course Outcomes

After completing this course satisfactorily, a student will be able to:

1. Describe how arrays, records, linked structures, stacks, queues, trees, and graphs are represented in memory and used by algorithms
2. Describe common applications for arrays, records, linked structures, stacks, queues, trees, and graphs.
3. Write programs that use arrays, records, linked structures, stacks, queues, trees, and graphs
4. Compare and contrast the benefits of dynamic and static data structures implementations
5. Describe the concept of recursion, give examples of its use, describe how it can be implemented using a stack .

TEXTBOOKS

- Fundamentals of Data structures in C, 2nd Edition, E. Horowitz, S. Sahni and Susan Anderson-Freed, Universities Press.
- Data structures and Algorithm Analysis in C, 2nd edition, M. A. Weiss, Pearson.
- Lipschutz: Schaum's outline series Data structures Tata McGraw-Hill

- D S Malik, Data Structures Using C++, Thomson, India Edition 2006.
- SamantaD, Classic Data Structures, Prentice-Hall of India, 2001.

DATA STRUCTURES LAB

Exercises

Credits : 2

No of hours/week : 2

Students are required to write and practically execute programs to solve problem using various data structures. The teacher can suitably device problems which help students experiment using the suitable datastructures and operations. Some of the problems are indicated below.

1. Write program that uses functions to perform the following:
 - a) Creation of list of elements where the size of the list, elements to be inserted and deleted are dynamically given as input.
 - b) Implement the operations, insertion, deletion at a given position in the list and search for an element in the list
 - c) To display the elements in forward / reverse order
2. Write a program that demonstrates the application of stack operations (Eg: infix expression to postfix conversion)
3. Write a program to implement queue data structure and basic operations on it (Insertion, deletion, find length) and code atleast one application using queues.
4. Write a program that uses well defined functions to Create a binary tree of elements and Traverse the a Binary tree in preorder, inorder and postorder,
5. Write program that implements linear and binary search methods of searching for an elements in a list
6. . Write and trace programs to understand the various phases of sorting elements using the methods
 - a) Insertion Sort b) Quicksort c) Bubble sort

OPERATING SYSTEM

Common for BA/B.Com/B.Sc Courses (Computer Maintenance as core subject)

Lectures: Each Unit: 12h

Total: 60h

Credits :4

LEARNING OBJECTIVES:

1. Describe the important computer system resources and the role of operating system in their management policies and algorithms.
2. To understand various functions, structures and history of operating systems and should be able to specify objectives of modern operating systems and describe how operating systems have evolved over time.
3. Understanding of design issues associated with operating systems.
4. Understand various process management concepts including scheduling, synchronization, and deadlocks.
5. To have a basic knowledge about multithreading.
6. To understand concepts of memory management including virtual memory.

UNIT I. (Introduction to Operating System) What is Operating System? History and Evolution of OS, Basic OS functions, Resource Abstraction, Types of Operating Systems– Multiprogramming Systems, Batch Systems, Time Sharing Systems; Operating Systems for Personal Computers, Workstations and Hand-held Devices, Process Control & Real time Systems.

UNIT II. (Operating System Organization and Process Characterization) Processor and User Modes, Kernels, System Calls and System Programs, System View of the Process and Resources, Process Abstraction, Process Hierarchy, Threads, Threading Issues, Thread Libraries; Process Scheduling, Non-Pre-emptive and Pre-emptive Scheduling Algorithms.

UNIT III. Process Management (Deadlock) Deadlock, Deadlock Characterization, Necessary and Sufficient Conditions for Deadlock, Deadlock Handling Approaches: Deadlock Prevention, Deadlock Avoidance and Deadlock Detection and Recovery.

UNIT IV. (Inter Process Communication and Synchronization) Concurrent and Dependent Processes, Critical Section, Semaphores, Methods for Inter-process Communication; Process Synchronization, Classical Process Synchronization Problems: Producer-Consumer, Reader-Writer.

UNIT V. (Memory Management) Physical and Virtual Address Space; Memory Allocation Strategies– Fixed and -Variable Partitions, Paging, Segmentation, Virtual Memory.

COURSE OUTCOMES:

Upon Completion of the course, the students will be able to:

- Students will be able to understand the various functions of an operating system
- Students will be able to understand process scheduling, process synchronization, deadlock handling mechanism.
- Students will be able to understand inter process communication, process synchronization.
- They will be able to understand how the OS manages memory allocation in case of multiprocessing.

TEXT BOOKS:

- A Silberschatz, P.B. Galvin, G. Gagne, Operating Systems Concepts, 8th Edition, John

Wiley Publications 2008.

- A.S. Tanenbaum, Modern Operating Systems, 3rd Edition, Pearson Education 2007.
- G. Nutt, Operating Systems: A Modern Perspective, 2nd Edition Pearson Education 1997.
- W. Stallings, Operating Systems, Internals & Design Principles 2008 5th Edition, Prentice Hall of India.
- M. Milenkovic, Operating Systems- Concepts and design, Tata McGraw Hill 1992.

OPERATING SYSTEMS LAB

Exercises

Credits : 2

No of hours/week : 2

1. Connecting & disconnecting computer peripherals and components & driver installation
2. OS installation like Windows XP, Windows 7, 8, 10
3. OS installation like FAT, NTFS
4. Basic trouble shoots using beep Sound
5. Dual OS installation
6. Assigning and identifying valid IP Addresses.
7. Configure network computers using switch
8. Installation of Network Interface Card (NIC).

DATABASE MANAGEMENT SYSTEMS

Common for BA/B.Com/B.Sc Courses (Computer Maintenance as core subject)

Lectures: Each Unit: 12h

Total: 60h

Credits :4

LEARNING OBJECTIVES:

1. Explain the basics of database systems and database management system software.
2. Ability to model data in applications using conceptual modeling tools such as ER Diagrams and design database schemas based on the model.
3. Formulate queries, using SQL, to a broad range of query and data update problems.
4. Demonstrate the need of normalization theory and apply such knowledge to the normalization of a database.
5. Be acquainted with the basics of transactions, concurrency control and recovery.

UNIT I. Overview of Database Management System: Introduction, Data and information, Database, Database management System, Objectives of DBMS, Evaluation of Database management System, Classification of Database Management System, file-based system, Drawbacks of file-Based System, advantages of DBMS, Data models, Database Architecture.
Relational Model: Introduction, CODD's Rules, relational data model,

UNIT II. Entity-Relationship Model: Introduction, the building blocks of an entity relationship diagram, classification of entity sets, attribute classification, relationship degree, relationship classification, reducing ER diagram to tables, enhanced entity-relationship model (EER model), generalization and specialization, **IS A** relationship and attribute inheritance, multiple inheritance, advantages of ER modeling.

UNIT III. SQL Data Definition: Specifying Tables, Data Types, Constraints; Simple SELECT, INSERT, UPDATE, DELETE Statements; Complex SELECT Queries, including Joins and Nested Queries; Views; Altering Schemas.

UNIT IV. Relational Algebra: Definition of Algebra; Relations as Sets; Operations: SELECT, PROJECT, JOIN, etc. Normalization Theory and Functional Dependencies, 2NF, 3NF, BCNF, 4NF, 5NF.

UNIT V. Indexing: Files, Blocks, and Records, Hashing; RAID; Replication; Single-Level and Multi-Level Indexes. Query Processing Translation of SQL into Query Plans; Basics of Transactions, Concurrency and Recovery.

COURSE OUTCOMES:

Upon Completion of the course, the students will be able to:

- Understand the basics of Database Management system, how the relational model evolved from the conventional file system.
- Able to analyse an organizational data storage and retrieval needs, using E-R Model and EER model.

- Able to create queries for simple to complex problems, simple to complex data.
- Understand the importance of normalization, apply the normalization techniques while designing the database.
- Realize the query processing, transaction processing, concurrency control and recovery procedures adapted by the database management system software

TEXTBOOKS

- A. Silberschatz, H.F. Korth, S. Sudarshan, Database System Concepts 6th Edition, McGrawHill, 2010.4. R. Elmasri, S.B.
- Elmasri's and Navathe's Fundamentals of Database Systems. Addison-Wesley
- Database Management Systems, Raghu Ramakrishnan, Johannes Gehrke, McGraw Hill Education

DATABASE MANAGEMENT SYSTEMS LAB

EXERCISES

Credits :2

No of hours/week :2

Students are required to practice the concepts learnt in the theory by designing and querying a database for a chosen organization (Like railway services, hospital administration Library, Payroll etc). The teacher may devise appropriate weekly lab assignments to help students practice the designing , querying a database in the context of example database. Some indicative list of experiments is given below.

1: E-R Model

Analyze the organization and identify the entities , attributes and relationships in it. Identify the primary keys for all the entities. Identify the other keys like candidate keys, partial keys, if any.

2. Concept design with E-R Model. Relate the entities appropriately. Apply cardinalities for each relationship. Identify strong entities and weak entities (if any).
3. Draw ER diagrams for train services in a railway station
- 4: Draw ER diagram for hospital administration
5. Creation of college database and establish relationships between tables
6. Write a view to extract details from two or more tables
7. Write a program to demonstrate nested queries
8. Write a program to demonstrate Joins
9. Write a program to demonstrate Aggregate functions
10. Write a program to implement views

COMPUTER NETWORKS

Common for BA/B.Com/B.Sc Courses (Computer Maintenance as core subject)

Lectures: Each Unit: 12h

Total: 60h

Credits :4

LEARNING OBJECTIVES:

1. Understand the structure of Data Communications System and its components. Be familiarize with different network terminologies.
2. Familiarize with contemporary issues in network technologies.
3. Know the layered model approach explained in OSI and TCP/IP network models
4. Identify different types of network devices and their functions within a network.
5. Learn basic routing mechanisms, IP addressing scheme and internetworking concepts.
6. Familiarize with IP and TCP Internet protocols.

UNIT I. Introduction to Computer Networks and Networking Elements: Network Definition, Network Topologies, Network Classifications, Network Protocol, Layered Network Architecture, Overview of OSI Reference Model, Overview of TCP/IP Protocol Suite, Hub, Switch (Managed and Unmanaged), Routers

UNIT II. Data Communication Fundamentals and Techniques: Analog and Digital Signal, Data-Rate Limits, Digital to Digital Line Encoding Schemes, Pulse Code Modulation, Parallel and Serial Transmission, Digital to Analog Modulation - Multiplexing Techniques- FDM, TDM, Transmission Media.

UNIT III. Networks Switching Techniques and Access Mechanisms: Circuit Switching, Packet Switching- Connectionless Datagram Switching, Connection- Oriented Virtual Circuit Switching; Dial-Up Modems, Digital Subscriber Line, Cable TV for Data Transfer.

UNIT IV. Data Link Layer Functions and Protocol: Error Detection and Error Correction Techniques, Data-Link Control- Framing and Flow Control, Error Recovery Protocols-Stop and Wait ARQ, Go-Back-N ARQ, Point to Point Protocol on Internet.

UNIT V. Multiple Access Protocol and Network Layer: CSMA/CD Protocols, Ethernet LANS; Connecting LAN and Back-Bone Networks- Repeaters, Hubs, Switches, Bridges, Router and Gateways, Networks Layer Functions and Protocols Routing, Routing Algorithms, Network Layer Protocol of Internet - IP Protocol, Internet Control Protocols.

COURSE OUTCOMES:

Upon Completion of the course, the students will be able to:

- Able to understand the fundamentals of computer networks, TCP/IP protocol.
- Able to understand the data communication techniques and multiplexing techniques.
- They will be able to understand the network switching techniques and various access mechanisms.
- Able to understand CSMA/CD protocols, routing algorithms.

TEXTBOOKS

- B. A. Forouzan: Data Communications and Networking, Fourth edition, THM Publishing Company Ltd 2007.
- A. S. Tanenbaum: Computer Networks, Fourth edition, PHI Pvt. Ltd 2002

EXERCISES

Credits :2

No of hours/week :2

1. Study of internet access technologies.
2. Study of installing network operating system (NOS).
3. Study of creating user accounts and groups in network.
4. Study of assigning permissions example printer.
5. Study of installing SUS.
6. Study of antivirus software installations.
7. Study of antispymware software installation.

Computer Organization & Architecture

Common for BA/B.Com/B.Sc Courses (Computer Maintenance as core subject)

Lectures: Each Unit: 12h

Total: 60h

Credits :4

LEARNING OBJECTIVES

1. To make students understand the basic structure, operation and characteristics of digital computer.
2. To familiarize the students with arithmetic and logic unit as well as the concept of pipelining.
3. To familiarize the students with hierarchical memory system including cache memories and virtual memory.
4. To make students know the different ways of communicating with I/O devices and standard I/O interfaces.

UNIT I: Fundamentals of Digital Electronics: Number systems, Complements, Fixed-Point Representation, Floating-Point Representation, Logic Gates- OR,AND,NOT, Combinational Circuits NAND, NOR, Exclusive-OR , Half-adder and Full adder.

UNIT II: Sequential Logic Circuits: Flip-Flops-Basic RS Flip-Flop, SR Flip-Flop, JK Flip-Flop, Registers, Counters, Multiplexer, Demultiplexer, Decoder, Encoder.

UNIT III: Basic Computer Organization: Instruction codes, Computer Registers, Computer Instructions, Input - Output & Interrupts - Instructions set of 8086 Microprocessor – Assembly language programming.

UNIT IV: Processor and Control Unit: Hardwired vs. Micro programmed Control Unit, General Register Organization, Stack Organization, Instruction Format, Data Transfer & Manipulation, Program Control, RISC, CISC processors.

UNIT V: Memory and I/O Systems: Peripheral Devices, I/O Interface, Data Transfer Schemes- Program Control, Interrupt, DMA Transfer. Memory Hierarchy- Processor vs. Memory Speed, High-Speed Memories, Cache Memory, Associative Memory, Interleave, Virtual Memory

COURSE OUTCOMES:

Upon Completion of the course, the students will be able to:

- Able to understand the basics of digital electronics, and get familiar with the electronics fundamentals.
- Understand the various types of flip-Flops, multiplexer and demultiplexer, decoder and encoder.
- Able to write instructions using assembly language.
- Learn the concepts related to memory and I/O systems.

TEXT BOOKS

1. Computer System Architecture, M. Morris Mano, 3rd Edition, Prentice Hall.
2. Computer Organization and Design, David A. Patterson and John L. Hennessey, Fifth edition, Morgan Kauffman / Elsevier, 2014.
3. Computer Organization and Architecture, William Stallings, Prentice Hall.
4. IBM PC Assembly Language Programming. Peter Abel
5. *Computer Organization*, Hamacher C, Vranesic Z & Zaky S, 5th edition, Mc Graw-Hill, 2002

Computer Organization & Architecture LAB

EXERCISES

Credits :2

No of hours/week :2

Students are required to write and practically execute programs to solve problems using MASM/ TASM. The teachers can suitably device problems which help students. Some of the programs are given below.

1. Moving of data from one block to another
2. To find sum of the given numbers
3. To find largest among n given numbers
4. To find factorial of the given number
5. To find the given number is prime or nor
6. To generate Fibonacci series
7. To convert binary number to decimal number
8. To convert decimal number to binary number