Department of IT & Computer Application
M.M.M. University of Technology
Gorakhpur

Credit structure for Two-Year MCA Programme
(For newly admitted students from session 2020-21)

Category	I	II	III	IV	Total
Programme Core (PC)	20	21	21	4	**66**
Programme Electives (PE)	-	-	4	4	**8**
Project (P)	-	-	-	10	**10**
Audit	-	-	-	-	-
Total	**20**	**21**	**25**	**18**	**84**

Junior Year, Semester-I

Sr.	Category	Paper Code	Subject Name	L	T	P	Cr.
1	PC	MCA-111	Object Oriented Programming with C^{++}	3	1	2	**5**
2	PC	MCA-112	Database Management Systems	3	1	2	**5**
3	PC	MCA-113	Computer Organization & Architecture	3	1	-	**4**
4	PC	MCA-114	Software Engineering	3	1	-	**4**
5	PC	MCA-115	Software Lab-1	-	1	2	**2**
6	AC	Audit Subject	-	-	-	-	-
Total				**12**	**5**	**6**	**20**

Junior Year, Semester-II

Sr.	Category	Paper Code	Subject Name	L	T	P	Cr.
1	PC	MCA-201	JAVA Programming	3	1	2	**5**
2	PC	MCA-202	Data Structures & Applications	3	1	2	**5**
3	PC	MCA-203	Computer Network	3	1	2	**5**
4	PC	MCA-204	Operating System Concepts	3	1	-	**4**
5	PC	MCA-205	Software Lab-2	-	1	2	**2**
6	AC	Audit Subject	-	-	-	-	-
Total				**12**	**5**	**8**	**21**

Senior Year, Semester-III

Sr.	Category	Paper Code	Subject Name	L	T	P	Cr.
1	PC	MCA-301	Artificial Intelligence	3	1	-	**4**
2	PC	MCA-302	Web Technologies	3	1	2	**5**
3	PC	MCA-303	Design & Analysis of Algorithms	3	1	2	**5**
4	PC	MCA-304	Cloud Computing	3	1	2	**5**
5	PE1	MCA-***	Elective-I	3	1/0	0/2	**4**
6	PC	MCA-305	Mini Project	-	-	4	**2**
7	AC	MCA-Ind	Industrial/Practical Training	-	-	2	-
Total				**15**	**5/4**	**10/12**	**25**

Senior Year, Semester-IV

Sr.	Category	Paper Code	Subject Name	L	T	P	Cr.
1	PC	MCA-401	Internet of Things	3	1	-	**4**
2	PE2	MCA-***	Elective-II	3	1/0	0/2	**4**
3	AC	MCA-Sem	Seminar	-	-	6	-
4	P	MCA-Prj	Project	-	-	20	**10**
Total				**6**	**2/1**	**22/24**	**18**

Programme Core (PC) for MCA

Sr.	Category	Paper Code	Subject Name	L	T	P	Cr.
1.	PC	MCA-111	Object Oriented Programming with c++	3	1	2	**5**
2.	PC	MCA-112	Database Management Systems	3	1	2	**5**
3.	PC	MCA-113	Computer Organization & Architecture	3	1	-	**4**
4.	PC	MCA-114	Software Engineering	3	1	-	**4**
5.	PC	MCA-115	Software Lab-1	-	1	2	**2**

6.	PC	MCA-201	JAVA Programming	3	1	2	**5**
7.	PC	MCA-202	Data Structures & Applications	3	1	2	**5**
8.	PC	MCA-203	Computer Network	3	1	2	**5**
9.	PC	MCA-204	Operating System Concepts	3	1	-	**4**
10.	PC	MCA-205	Software Lab-2	-	1	2	**2**
11.	PC	MCA-301	Artificial Intelligence	3	1	-	**4**
12.	PC	MCA-302	Web Technologies	3	1	2	**5**
13.	PC	MCA-303	Design & Analysis of Algorithms	3	1	2	**5**
14.	PC	MCA-304	Cloud Computing	3	1	2	**5**
15.	PC	MCA-305	Mini Project	-	-	4	**2**
16.	PC	MCA-401	Internet of Things	3	1	-	**4**
Total				**39**	**15**	**24**	**66**

Programme Electives (PE-I)

Sr.	Paper Code	Subject Name	L	T	P	Cr.
1	MCA-351	Automata Theory	3	1	-	**4**
2	MCA-352	Python Programming	2	1	2	**4**
3	MCA-353	Information Security & Cyber Laws	3	1	-	**4**
4	MCA-354	Android Programming	2	1	2	**4**
5	MCA-355	Distributed Database Systems	3	1	-	**4**
6	MCA-356	Computer Graphics	3	1	-	**4**
7	MCA-357	Distribution Systems	3	1	-	**4**
8	MCA-358	Data Mining & Warehousing	3	1	-	**4**

Programme Electives (PE-II)

Sr.	Paper Code	Subject Name	L	T	P	Cr.
1	MCA-451	Compiler Design	2	1	2	**4**
2	MCA-452	Machine Learning	2	1	2	**4**
3	MCA-453	Wireless Sensor Networks	3	1	-	**4**
4	MCA-454	Mobile Computing	3	1	-	**4**
5	MCA-455	Big Data Technologies	3	1	-	**4**
6	MCA-456	Data Sciences & Analysis	3	1	-	**4**
7	MCA-457	Natural Language Processing	2	1	2	**4**
8	MCA-458	Blockchain & Cryptocurrency	3	1	-	**4**

Audit Courses for MCA

Sr.	Paper Code	Subject Name	L	T	P	Cr.
1	BAS-03	Professional Communication	3	1	-	**4**
2	BAS-24	Applied Computational Methods	3	1	2	**5**
3	MAS-103	Communication for Business Management	3	1	2	**5**
4	MAS-105	Applied Probability & Statistics	3	1	0	**4**
5	MAS-106	Discrete Mathematics	3	1	0	**4**

MCA-111 OBJECT ORIENTED PROGRAMMING WITH C++ 5 Credits (3-1-2)

Course Objectives:

1. Understand object-oriented programming and advanced C++ concepts.
2. Be able to program using more advanced C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.
3. Be able to build C++ classes using appropriate encapsulation and design principles.

Learning Outcomes:

1. To understand the concept of Object-Oriented Programming and master OOP using C++.
2. Ultimate goal: to make you a good programmer
3. Improve your problem-solving skills.

UNIT I **9**

Object Oriented Programming Paradigm, Basic Concepts of Object Oriented Programming, Benefits of OOP, Object Oriented Languages, Tokens, Keywords, Identifiers and Constants, Basic Data Types, User-Defined and Derived Data Types, Type Compatibility, Reference, Variables, Scope Resolution Operator, Type Casting, Implicit Conversion, Operator Precedence, Control Structures, Structure, Function.

UNIT II **9**

Class specification, class objects, accessing class members, data hiding, empty classes, pointers within a class, passing objects as arguments, returning objects from functions, friend functions and friend classes, constant parameters and member functions, structures and classes, static members, objects and memory resource, class design steps. Constructors, destructor, constructor overloading, order of construction and destruction, constructors with default arguments, nameless objects, dynamic initialization through constructors, constructors with dynamic operations, constant objects and constructor, static data members with constructors and destructors, nested classes.

UNIT III **9**

Defining Operator Overloading, Overloading Unary Operators, Overloading Binary Operators, Overloading Binary Operators Using Friends, Manipulation of Strings Using Operators, Rules for Overloading Operators, Type Conversions, Deriving Derived Classes, Single, Multilevel, Multiple, Hierarchical, Hybrid Inheritance, constructors destructors in derived classes, constructor's invocation and data members initialization, Virtual Base Classes, Abstract Classes.

UNIT IV **9**

Pointers to Objects, Classes for File Stream Operations, Opening and Closing a File, File Pointers and their Manipulations, Sequential Input and Output Operations. Class Templates with multiple parameters, Function Templates, Overloading of Template Functions, Member Function Templates.

COMPUTER PROGRAMMING LAB

Write C++ Programs to illustrate the concept of the following:

1. Arrays
2. Structures
3. Pointers
4. Objects and Classes
5. Console I/O operations
6. Scope Resolution and Memory Management Operators
7. Inheritance
8. Polymorphism
9. Virtual Functions
10. Friend Functions
11. Operator Overloading
12. Function Overloading
13. Constructors and Destructors
14. this Pointer
15. File I/O Operations
16. Assignments on developing interfaces- multiple inheritance, extending interfaces
17. Assignments on multithreaded programming, handling errors and exceptions, applet programming and graphics programming

Books & References:

1. B. Trivedi, "Programming with ANSI C++", Oxford University Press, 2007.
2. Ira Pohl, "Object Oriented Programming using C++", Pearson Education, Second Edition Reprint
3. B. Stroustrup, "The C++ Programming language", Third edition, Pearson Education, 2004.
4. Robert Lafore, "Object Oriented Programming in Turbo C++", Galgotia Publication 1994.
5. E. Balagurusamy,"Object-Oriented Programming with C++", TMH Publication.
6. Booch, Maksimchuk, Engle, Young, Conallen and Houstan, "Object Oriented Analysis and Design with Applications", Pearson Education.
7. S. B. Lippman, Josee Lajoie, Barbara E. Moo, "C++ Primer", Fourth Edition, Pearson Education 2005.
8. Timthy Budd, "An Introduction to Object Oriented Programming with C++," Addition-Wesley.
9. Kip R. Irvine," C++ and Object-Oriented Programming," Prentice Hall

MCA-112 DATABASE MANAGEMENT SYSTEMS 5 Credits (3-1-2)

Course Objectives: Upon successful completion of the course, students

1. know and understand the fundamental concepts, techniques, and terminology of the database management system.
2. know and understand the basic syntax, semantics, and pragmatics of SQL & PL/SQL.
3. can analyze problems and apply DBMS concepts and techniques to develop appropriate programs to solve the problems,
4. can evaluate alternative database designs to determine which are better according to selected criteria,
5. know and understand the basic features of database transactions and concurrency control.
6. are able to reason about and manipulate concurrency control techniques.

Learning Outcomes:

1. list and define the fundamental concepts of database management system.
2. manually execute a given (simple) database design a transaction over it.
3. manually infer the type of a given (simple) database transaction.
4. implement (simple) algorithms and data structures as database transaction.
5. design (large) databases that are modular and have reusable components.
6. explain on a simple problem how to apply concurrency control over concurrent database transactions.

UNIT- I **9**

Introduction: An overview of database management system, database system Vs file system, Database system concepts and architecture, data models schema and instances, data independence and data base language and interfaces, Data definitions language, DML, Overall Database Structure.
Data Modeling using the Entity Relationship Model: ER model concepts, notation for ER diagram, mapping constraints, keys, Concepts of Super Key, candidate key, primary key, Generalization, aggregation, reduction of an ER diagrams to tables, extended ER model, relationships of higher degree

UNIT- II **9**

Relational data Model and Language: Relational data model concepts, integrity constraints: entity integrity, referential integrity, Keys constraints, Domain constraints, relational algebra, relational calculus, tuple and domain calculus, **Introduction to SQL:** Characteristics of SQL. Advantage of SQL. SQL data types and literals. Types of SQL commands. SQL operators and their procedure. Tables, views and indexes. Queries and sub queries. Aggregate functions. Insert, update and delete operations. Joins, Unions, Intersection, Minus, Cursors in SQL. induction over

UNIT- III **9**

Data Base Design & Normalization: Functional dependencies, normal forms, first, second, third normal forms, BCNF, inclusion dependences, loss less join decompositions, normalization using FD, MVD, and JDs, alternative approaches to database design.

UNIT- IV **9**

Transaction Processing Concepts: Transaction system, Testing of serializability, Serializability of schedules, conflict & view serializable schedule, recoverability, Recovery from transaction failures, log based recovery, checkpoints, deadlock handling. **Concurrency Control Techniques:** Concurrency control, locking techniques for concurrency control, Time stamping protocols for concurrency control, validation based protocol, multiple granularity, Multi version schemes, Recovery with concurrent transaction.

COMPUTER PROGRAMMING LAB

1. Exercises to be based on Sybase / Oracle / Postgres / VB / Power Builder / DB2 / MS-Access.

2. Applications involving vendor development systems, stores management system, finance management etc.
3. Creation and querying of database tables for following cases. .

    a. Write SQL queries using logical operations (=,<,>,etc)
    b. Write SQL queries using SQL operators
    c. Write SQL query using character, number, date and group functions
    d. Write SQL queries for relational algebra
    e. Write SQL queries for extracting data from more than one table
    f. Write SQL queries for sub queries, nested queries
    g. Write programme by the use of PL/SQL
    h. Concepts for ROLL BACK, COMMIT & CHECK POINTS
    i. Create VIEWS, CURSORS and TRGGERS & write ASSERTIONS.
    j. Create FORMS and REPORTS
    k.
4. Design of tables by normalization and dependency analysis.
5. Writing application software with host language interface

Books & References:

1. Date C J, “An Introduction To Database System”, Addision Wesley.
2. Korth, Silbertz, Sudarshan, “Database Concepts”, McGraw Hill.
3. Elmasri, Navathe, “Fundamentals Of Database Systems”, Addision Wesley.
4. Leon & Leon, “Database Management System”, Vikas Publishing House.
5. Bipin C. Desai, “An introduction to Database Systems”, Galgotia Publication.
6. Majumdar & Bhattacharya, “Database Management System”, TMH.
7. Ramakrishnan, Gehrke, “Database Management System”, McGraw Hill.
8. Kroenke, “Database Processing: Fundamentals, Design and Implementation”, Pearson Education.
9. Maheshwari Jain, “DBMS: Complete Practical Approach”, Firewall Media, New Delhi.

MCA-113 COMPUTER ORGANISATION & ARCHITECTURE 4 Credits (3-1-0)

Course Objectives:

1. To impart the essential knowledge on the fundamentals and applications of digital circuits and digital computing principles
2. To provide an overview on the design principles of digital computing systems
3. To provide technical knowledge about various digital hardware components
4. To understand how computers are constructed out of a set of functional units and how the functional units operate, interact, and communicate

Learning Outcomes:

1. Ability to understand the merits and pitfalls in computer performance measurements
2. Ability to understand memory hierarchy and its impact on computer cost/ performance

UNIT- I **9**

Data Representation, Binary, Octal, HEX and their inter-conversion, 1’s and 2’s complement, Binary Arithmetic, Number Systems – BCD, EBCDIC, ASCII.Basic Gates & its Truth tables, Boolean algebra, Fundamental concepts of Boolean algebra, Basic Theorem and properties of Boolean algebra, Boolean functions, Canonical and standard forms, Sum Of Product, Product of Sum, K-map method (up to 4 variables), don’t care conditions, Combination circuit design with AND, OR, NOT, NAND, NOR gates, Exclusive-OR and Equivalence Functions, Universal gates functionality

UNIT- II **9**

Combinational Circuits: Half Adder, Full Adder, Binary Adder and Subtracter, Decoder / Encoder, Multiplexer /Demultiplexer.
Sequential Circuits: Flip Flops - SR, D, JK, Master – Slave, Edge Triggered, Shift Registers, Synchronous Counter and Asynchronous Counter.

UNIT- III **9**

Register Transfer Language, Bus and Memory Transfers, Bus Architecture, Bus Arbitration, Arithmetic Logic, Shift Microoperation, Arithmetic Logic Shift Unit, Design of Fast address, IEEE standard for Floating point numbers. Control Design: Hardwired & Micro Programmed Control Unit. Processor Design: Processor

Organization: General register organization, Stack organization, Addressing mode, Instruction format, Data transfer & manipulations, Program Control, Reduced Instruction Set Computer.

UNIT- IV **9**

Input-Output Organization: I/O Interface, Modes of transfer, Interrupts & Interrupt handling, Direct Memory access, Input-Output Processor, Serial Communication.
Memory Organization: Memory Hierarchy, Main Memory (RAM and ROM Chips), Auxiliary memory, Cache memory, Virtual Memory.

Books & References:

1. Digital Design by M Morris Mano, M D Ciletti
2. Computer System Architecture, by M. Mano
3. Computer Organization, by Vravice, Zaky & Hamacher

MCA-114 **SOFTWARE ENGINEERING** **4 Credits (3-1-0)**

Course Outcomes

The students are expected to be able to demonstrate the following knowledge, skills and attitudes after completing this course.

1. Enhance the Software Project Management skills.
2. Develop functioning software which benchmarks to the international standards.

UNIT- I

Evolution and impact of Software engineering, software life cycle models: Waterfall, prototyping, Evolutionary and Spiral models. Feasibility study, Functional and Non-functional requirements, Requirements gathering, Requirements analysis and specification.

UNIT- II

Basic issues in software design, modularity, cohesion, coupling and layering, function-oriented software design: DFD and Structure chart, object modeling using UML, Object-oriented software development, user interface design. Coding standards and Code review techniques

UNIT- III

Fundamentals of testing, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, White box and black box testing, Alpha and Beta Testing of Products, Test coverage analysis and test case design techniques, mutation testing, Static and dynamic analysis, Software reliability metrics.

UNIT- IV

Software Maintenance and Software Project Management, Software as an evolutionary entity, need for maintenance, categories of maintenance: Preventive, Corrective and Perfective Maintenance, Cost of Maintenance, Software Re-Engineering, Reverse Engineering. Software Configuration Management Activities, Change Control Process, Software Version Control, An Overview of CASE Tools. Estimation of Various Parameters such as Cost, Efforts, Schedule/Duration, Constructive Cost Models (COCOMO), Resource Allocation Models, Software Risk Analysis and Management.

Textbooks

1. R. S. Pressman, "Software Engineering - A practitioner's approach", III Edition, McGraw Hill International editions, 1992.

Reference Books

1. IAN Sommerville, "Software Engineering", Pearson Education Asia, VI Edition, 2000.
2. Pankaj Jalote, "An Integrated Approach to software Engineering", Springer Verlag, 1997.

MCA-115 **Software Lab-1**

Course category : Programme Core (PC)

Pre-requisite Subject : NIL

Contact hours/week : Lecture: 0, Tutorial: 1 , Practical: 2

Number of Credits :2

Course Assessment methods : Continuous assessment through Viva-voce, Practical work/Record, attendance and Major Practical Examination

Course Outcomes : The students are expected to be able to demonstrate the following knowledge, skills and attitudes after completing this course

1. Understanding of Booting Process and installation of Operating system
2. Usage of Operating system commands
3. Understanding of Shell and its usage as a programming language
4. Understanding of Computer Networking concepts

Experiments

1. Understanding CMOS settings of operating system
2. Installation of Linux operating system using virtualization technique
3. Understanding and practice of various Linux commands
4. Creation/usage of various types of files supported by Linux
5. Practice of Computer networking commands
6. Programs using shell programming

MCA-201 JAVA PROGRAMMING 5 Credits (3-1-2)

Course Outcomes

The students are expected to be able to demonstrate the following knowledge, skills and attitudes after completing this course.

1. Analyze and explain the behavior of programs involving the fundamental program constructs
2. Identify and correct syntax and logic errors in short programs
3. Design and implement a class based on attributes and behaviors of objects
4. Describe the parameter passing mechanisms in terms of formal parameters, actual parameters, non-object parameters and object parameters

UNIT- I

Object oriented programming concepts – objects – classes – methods and messages – abstraction and encapsulation – inheritance – abstract classes – polymorphism. - Objects and classes in Java – defining classes – methods - access specifiers – static members – constructors – finalize method
Arrays – Strings - Packages – Java-Doc comments –- Inheritance – class hierarchy – polymorphism – dynamic binding – final keyword – abstract classes

UNIT- II

The Object class – Reflection – interfaces – object cloning – inner classes – proxies - I/O Streams - Graphics programming – Frame – Components – working with 2D shapes.

UNIT- III

Basics of event handling – event handlers – adapter classes – actions – mouse events – AWT event hierarchy – introduction to Swing – Model-View-Controller design pattern – buttons – layout management – Swing Components – exception handling – exception hierarchy – throwing and catching exceptions.

UNIT- IV

Motivation for generic programming – generic classes – generic methods – generic code and virtual machine – inheritance and generics – reflection and generics - Multi-threaded programming – interrupting threads – thread states – thread properties – thread synchronization – Executors – synchronizers.
JDBC: The connectivity Model, JDBC/ODBC Bridge, java.sql package, connectivity to remote database.

EXPERIMENTS

1. Basic programs of simple statements, conditional statements, iterative statements and arrays
2. Programs having object-oriented concepts like Inheritance and Interface
3. Programs for Exception Handling and Event Handling
4. Programs of Threads and Multithreading

5. Programs related to Applets and Swings
6. Programs including JAVA Beans and Servlets

Textbooks

1. Naughton, Schildt, The Complete Reference JAVA2, TMH.
2. Balaguruswamy E, Programming in JAVA, TMH
3. Cay S. Horstmann and Gary Cornell, "Core Java: Volume I – Fundamentals", Eighth Edition, Sun Microsystems Press
4. K. Arnold and J. Gosling, "The JAVA programming language", Third edition, Pearson Education, 2000.

Reference Books

1. Margaret Levine Young, The Complete Reference Internet, TMH.
2. Dustin R. Callway, Inside Servlets, Addison Wesley.
3. Mark Wutica, JAVA Enterprise Edition, QUE.
4. Steven Holzner, JAVA2 Black book, Dreamtech.
5. Timothy Budd, "Understanding Object-oriented programming with Java", Updated Edition, Pearson Education, 2000.
6. C. Thomas Wu, "An introduction to Object-oriented programming with Java", Fourth Edition, Tata McGraw-Hill Publishing company Ltd., 2006

MCA-202 DATA STRUCTURE & APPLICATIONS 5 Credits (3-1-2)

Course Objectives:

1. To understand the various techniques of sorting and searching
2. To design and implement arrays, stacks, queues, and linked lists
3. To understand the complex data structures such as trees and graphs

Learning Outcomes: On completion of this course, students are expected to be capable of understanding the data structures, their advantages and drawbacks, how to implement them in C, how their drawbacks can be overcome and what the applications are and where they can be used. Students should be able to learn about the data structures/ methods/algorithms mentioned in the course with a comparative perspective so as to make use of the most appropriate data structure/ method/algorithm in a program to enhance the efficiency (i.e. reduce the run-time) or for better memory utilization, based on the priority of the implementation.

UNIT- I: 9

Linear Data Structure Introduction: Need of data structure, Concepts a) Data and data structure b) Abstract Data Type and Data Type. Algorithms and programs, basic idea of pseudo-code, Algorithm efficiency and analysis, time and space analysis of algorithms – order notations Array: Different representations – row major, column major. Sparse matrix - its implementation and usage, Array representation of polynomials Linked List: Singly linked list, circular linked list, doubly linked list, linked list representation of polynomial and applications.

UNIT- II 9

Stack and Queue: Stack and its implementations (using array, using linked list), applications. Queue, circular queue, dequeue. Implementation of queue- both linear and circular (using array, using linked list), applications Recursion: Principles of recursion – use of stack, differences between recursion and iteration, tail recursion. Applications - The Tower of Hanoi, Eight Queens Puzzle

UNIT- III 9

Nonlinear Data structures Trees: Basic terminologies, forest, tree representation (using array, using linked list). Binary trees - binary tree traversal (pre-, in-, post- order), threaded binary tree (left, right, full) - non-recursive traversal algorithms using threaded binary tree, expression tree. Binary search tree- operations (creation, insertion, deletion, searching). Height balanced binary tree – AVL tree (insertion, deletion with examples only). B-Trees – operations (insertion, deletion with examples only). Graph definitions and concepts Graph representations/storage implementations – adjacency matrix, adjacency list, adjacency multi-list. Graph traversal and connectivity – Depth first search (DFS), Breadth-first search (BFS) – concepts of edges used in DFS and BFS, applications. Minimal spanning tree – Prim's algorithm (basic idea of greedy methods).

UNIT- IV 9

Searching, Sorting: Sorting Algorithms: Bubble sort and its optimizations, insertion sort, shell sort, selection sort, merge sort, quick sort, heap sort (concept of max heap, application – priority queue), radix sort. Searching: Sequential search, binary search, interpolation search Hashing: Hashing functions, collision resolution techniques.

EXPERIMENTS: Write C/C++ Programs to illustrate the concept of the following:

1. Arrays
2. Linked List
3. Stack
4. Queue
5. Graph
6. Tree
7. Searching & Sorting Algorithms

Books & References:

1. Data Structures and Program Design In C, 2/E by Robert L. Kruse, Bruce P. Leung.
2. Data Structures in C by Aaron M. Tenenbaum.
3. Fundamentals of Data Structures of C by Ellis Horowitz, Sartaj Sahni, Susan Anderson-freed.
4. Data Structures by S. Lipschutz.
5. Data Structures Using C by Reema Thareja.
6. Data Structure Using C, 2/e by A.K. Rath, A. K. Jagadev.
7. Introduction to Algorithms, by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein

MCA-203 COMPUTER NETWORK 5 Credits (3-1-2)

Course Outcomes

The students are expected to be able to demonstrate the following knowledge, skills and attitudes after completing this course.

1. Understand the concepts of communication architecture and protocols
2. Identify different types of communication mediums and techniques
3. Define and identify different types of multiplexing, data encoding, modulation, and switching techniques
4. Illustrate different standards of Local Area Network in terms of technologies and hardware used
5. Illustrate network addressing and analysis techniques
6. Understand the Wide Area Network technologies
7. Understand the network routing concepts
8. Understand the internetworking concepts and architectures
9. Understand the TCP/IP protocols and design architectures

Topics Covered

UNIT-I

Introductory Concepts: Goals and Applications of Networks, Network structure and architecture, the OSI reference model, services, networks topology, Physical Layer- transmission, switching methods, LAN Inter connection devices, Integrated services digital networks.

UNIT-II

Medium access sub layer: Channel allocations, LAN protocols, ALOHA Protocols- Pure ALOHA, slotted ALOHA, Carrier Sense Multiple Access Protocols, CSMA with Collision Detection, Collision free Protocols, IEEE standards, Ethernet, FDDI, Data Link Layer- basic design issues, error correction & detection algorithms, elementary data link layer protocols, sliding window protocols, error handling, High Level Data Link Control

UNIT-III

Network Layer: Packet switched networks – IP – ARP – RARP –DHCP – ICMP – Queuing discipline – Routing algorithms, congestion control algorithms, internetworking, TCP/IP protocol, IP addresses, IPv4 and IPv6.

UNIT-IV

Transport Layer: Design issues, connection management, Internet Transport Protocol (UDP), Transmission Control Protocol. (TCP) -Adaptive Retransmission - Congestion control , Congestion avoidance – QoS.
Application Layer: Domain Name System,Electronic mail (Email), File Transfer Protocol, Hyper Text Transfer, Protocol, Introduction to Cryptography and Network Security (DES, RSA algorithms), Communication Security (IPSec, Firewalls).

EXPERIMENTS

1. To create scenario and study the performance of CSMA/CD protocol through simulation.
2. To create scenario and study the performance of token bus and token ring protocols through simulation.
3. Implementation of Error detection and correction algorithms.
4. Implementation and study of 1-bit sliding window viz., stop and wait protocol.
5. Implementation and study of Go back-N protocol.
6. Implementation and study of selective repeat protocol.
7. To get the MAC or Physical address of the system using Address Resolution Protocol.
8. Implementation of distance vector routing algorithm.
9. Implementation of link state routing algorithm.
10. To write a client-server application for chat using TCP.
11. To write a C program to develop a DNS client server to resolve the given hostname.

Textbooks

1. Data Communication and Networking by Forouzan TMH
2. A. S Tanenbaum, Computer Networks, 4th, Edition", Pearson education

Reference Books

1. Data and Computer Communication by W. Stallings, Macmillan Press
2. Computer Networks & Internet with Internet Applications by Comer Pearson Education
3. Computer Networks with Internet Protocols by W Stallings, Pearson Education
4. Local and Metropolitan Area Networks by W Stallings, VIth edition, Pearson Education

MCA-204 OPERATING SYSTEM CONCEPTS 4 Credits (3-1-0)

Course Objectives:

This course should provide the students with good understanding of Operating System including its architecture and all its components. Good conceptions on all the subjects like processes, inter-process communication, semaphore, message passing, classical systems, security and protection mechanism, I/O hardware and software, deadlocks, etc.

Learning Outcomes:

1. List and define the fundamental concepts of Operating System
2. Study various CPU scheduling, Disk scheduling algorithms and compare their performance
3. Ability to simulate the given problems

UNIT- I **9**

Introduction: Basic architectural concepts, Operating System Services, interrupt handling, concepts of batch-processing, multiprogramming, time-sharing, real-time operations; Resource Manager view, process view and hierarchical view of an OS.
Memory management: Partitioning, paging, concepts of virtual memory demand paging, page replacement algorithms, segmentation, Segmentation and demand-paging, Cache memory management.

UNIT- II **9**

Processor management: CPU scheduling – short-term, medium term and long term scheduling, non-preemptive and preemptive algorithms, performance analysis of multiprogramming, multiprocessing and interactive systems, Concurrent processes, precedence graphs, critical section problem, semaphores; Classical process, co-ordination problems, Producer-consumer problem, Reader-writer problem, Dining philosophers problem, Barber's shop problem, Interprocess communication.

UNIT- III **9**

Concurrent Programming: Critical region, conditional critical region, monitors, Deadlocks: prevention, avoidance, detection and recovery. Device Management: Scheduling algorithms – FCFS, shortest-seek-time-first, SCAN, C- SCAN, LOOK, C-LOOK algorithms, spooling, spool management algorithm.

UNIT- IV **9**

Information Management: File concept, file support, directory structures, symbolic file directory, basic file directory, logical file system, physical file system, access methods, file protection, file allocation strategies. Protection: Goals, policies and mechanisms, domain of protection, access matrix and its implementation, access lists, capability lists, Lock/Key mechanisms, passwords, dynamic protection scheme, security concepts and public and private keys, RSA encryption and decryption algorithms.
A case study: A UNIX OS file system, shell, filters, shell programming, programming with the standard I/O, UNIX system calls.

Books & References:

1. A. Silberschatz and P. B. Galvin: Operating Systems Concepts, 5th ed., John Wiley and Sons, New York, 1998.
2. J. L. Peterson and A. Silberschatz: Operating Systems Concepts, Addison-Wesley, Reading, Mass., 1987.
3. P. B. Hansen: Operating System Principles, Prentice Hall, Englewood Cliffs, 1980.
4. A. S. Tannenbaum: Modern Operating Systems, Prentice Hall, Englewood Cliffs, 1992.
5. S. E. Madnick and J. J. Donovan: Operating Systems, McGraw Hill, New York

MCA-205 **Software Lab-2** **2 Credits (0-1-2)**

Course Objectives:

The objectives of this course are:

1. To give overview of advanced C programming (command line arguments & file handling)
2. To give knowledge about practical usage of LINUX system calls
3. To develop capability of system call programming for practical understanding of operating system's internal functioning

Learning Outcomes:

The students are expected to be able to demonstrate the following knowledge, skills and attitudes after completing this course

1. Understanding of process handling in Operating system
2. Usage of kernel system calls
3. Understanding of multiprogramming function of operating system using system calls

Experiments

1. Programs related to advanced C programming (command line arguments & file handling)
2. Understanding and usage of various LINUX kernel system calls
3. System call programming (programs based on LINUX kernel system calls using C language)