**ANNA UNIVERSITY**
**CENTRE FOR DISTANCE EDUCATION**
**MASTER OF SCIENCE IN COMPUTER SCIENCE**

**REGULATIONS - 2018 CURRICULUM**

**SEMESTER – I**

| CODE NO. | COURSE TITLE | CREDITS* | MARKS |
|---|---|---|---|
| DCS5101 | Computer Organization | 2 | 100 |
| DCS5102 | Problem Solving and Programming | 2 | 100 |
| DCS5103 | Database Management System | 2 | 100 |
| DCS5104 | Software Engineering | 2 | 100 |
| DCS5105 | Mathematical Foundations of Computer Science | 4 | 100 |
| DCS5111 | Programming Lab | 2 | 100 |
| DCS5112 | Database Management System Lab | 2 | 100 |
| | **Total** | **16** | **700** |

**SEMESTER – II**

| CODE NO. | COURSE TITLE | CREDITS* | MARKS |
|---|---|---|---|
| DCS5201 | Computer Networks | 4 | 100 |
| DCS5202 | Object Oriented Programming | 2 | 100 |
| DCS5203 | Data Structures and Algorithms | 2 | 100 |
| DCS5204 | Operating System | 4 | 100 |
| DCS5205 | Software Project Management | 2 | 100 |
| DCS5211 | Object Oriented Programming Lab | 2 | 100 |
| DCS5212 | Data Structures and Algorithms Lab | 2 | 100 |
| | **Total** | **18** | **700** |

**SEMESTER – III**

| CODE NO. | COURSE TITLE | CREDITS* | MARKS |
|---|---|---|---|
| DCS5301 | Information Security | 2 | 100 |
| DCS5302 | Data Warehousing and Mining | 2 | 100 |
| DCS5303 | Web Programming | 2 | 100 |
| DCS5304 | Object Oriented Analysis and Design | 4 | 100 |
| E1 | Elective I | 2 | 100 |
| DCS5311 | Web Programming Lab | 2 | 100 |
| DCS5312 | Software Development Lab | 2 | 100 |
| | **Total** | **16** | **700** |

**SEMESTER – IV**

| CODE NO. | COURSE TITLE | CREDITS* | MARKS |
|---|---|---|---|
| E2 | Elective II | 2 | 100 |
| E3 | Elective III | 2 | 100 |
| DCS5411 | Project Work | 12 | 200 |
| | **Total** | **16** | **400** |
| | **Total No. of Credits** | **66** | **2500** |

*Each credit is equivalent to 30 hours of student study comprising of all learning activities.

## ELECTIVES

### SEMESTER – III

### ELECTIVE – I

| CODE NO. | COURSE TITLE | CREDITS* | MARKS |
|---|---|---|---|
| DCS5001 | Mobile Computing | 2 | 100 |
| DCS5002 | XML and Web Services | 2 | 100 |
| DCS5003 | Theory of Computation | 2 | 100 |

### SEMESTER – IV

### ELECTIVE - II

| CODE NO. | COURSE TITLE | CREDITS* | MARKS |
|---|---|---|---|
| DCS5004 | Computer Graphics and Multimedia Systems | 2 | 100 |
| DCS5005 | Ethical Hacking and Cyber Forensics | 2 | 100 |
| DCS5006 | Software Testing | 2 | 100 |

### ELECTIVE – III

| CODE NO. | COURSE TITLE | CREDITS* | MARKS |
|---|---|---|---|
| DCS5007 | Cloud Computing | 2 | 100 |
| DCS5008 | Visual Programming | 2 | 100 |
| DCS5009 | E Commerce | 2 | 100 |

**ANNA UNIVERSITY**
**CENTRE FOR DISTANCE EDUCATION**
**MASTER OF SCIENCE IN COMPUTER SCIENCE**

**REGULATIONS - 2018**
**SYLLABUS I TO IV SEMESTERS**

**SEMESTER – I**

**DCS5101** **COMPUTER ORGANIZATION & DESIGN**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

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

- Master the binary and hexadecimal number systems including computer arithmetic.
- Design and implement digital systems with basic gates and other components using combinational and sequential circuits.
- Be familiar with the Von Neumann architecture.
- Be familiar with the functional units of the processor and addressing modes, instruction sets.
- Be familiar with the memories and cache subsystem.
- Be familiar with different ways of communicating with I/O devices and standard I/O interfaces.

**COURSE OUTCOMES**

- Understand the fundamentals of Boolean logic and functions.
- To have a thorough understanding of the basic structure and operation of a digital computer.
- Design and realize digital systems with basic gates and other components using combinational and sequential circuits.
- Discuss in detail the operation of the arithmetic and logic unit.
- To study the instruction sets and operation of a processor.
- To study the different ways of communicating with I/O devices and standard I/O Interfaces.
- To study the hierarchical memory system including cache memories and virtual memory.

**UNIT I** Digital systems, binary numbers, octal, hexadecimal conversions, signed binary numbers, complements, logic gates, Boolean algebra , K-maps, standard forms, NAND-NOR implementation.

**UNIT II** Combinational circuits, adder, subtractor, ALU design, decoder, encoder, multiplexers, Sequential circuits: latches, flip-flops, registers, memories, up- down counters.

**UNIT III** Von-neumann architecture, processor :definition, structure ,category, technology, ALU concept, stored programs, fetch execute cycle, instruction formats, clock rate instruction rate, pipeline, current processors, multi core processors.

**UNIT IV** Physical memory , addressing, virtual memory, address translation, paging, cache, L1,L2,L3 cache memories, cache mapping, LRU replacement.

**UNIT V** Data transfer, Serial and Parallal data transfer, Full duplex- half duplex interaction, Bus interface, Programmed I/O, Polling, Interrupt driven I/O, Hardware interrupt mechanism, Interrupt vectors, Multi level of interrupts,DMA, buffer chaining, operation chaining.

**REFERENCE BOOKS:**


1. Marris mano, “Digital design” PHI/Pearson fourth edition 2006 Essentials of Computer Architecture Douglas E.Comer Pearson sixth edition 2012
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, “Computer Organization”, Tata McGraw Hill, Fifth Edition, 2002
3. William Stallings, “ Computer Organization and Architecture – Designing for Performance”, Pearson Education, Seventh Edition, 2006.

4. David A Patterson and John L. Hennessy, “ Computer Organization and Design, The Hardware/Software Interface”, Morgan Kaufmann / Elsevier, Third Edition, 2005.

**DCS5102** **PROBLEM SOLVING AND PROGRAMMING**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- Understand the various problem solving techniques.
- To be aware of the top down design technique.
- To learn the syntax of C.
- To be exposed to the file processing techniques of C.
- To be familiarized with the preprocessor directives.

**COURSE OUTCOMES**

Upon completion of the course, the students should be able to

- Design and implement C programs for any given problem.
- Work with existing programs and modify it as per the requirements.
- Identify the errors in a C program.
- Identify the output of a C program without actually executing it.

**UNIT I** **PROBLEM SOLVING -** Introduction – The Problem–Solving Aspect – Top Down Design – Implementation of Algorithms – Program Verification – The Efficiency of Algorithms – The Analysis of Algorithms.

**UNIT II** **BASICS OF C PROGRAMMING -** Introduction to C Programming Environment – History of C – C Standard Library – Basics of C Program Development Environment - Introduction to C Programming - A simple C Program – Memory Concepts – Arithmetic – Decision Making – Relational Operators – Assignment – Increment and Decrement Operators- Structured Program Development – Algorithms – Pseudocode- Control Structures – if , if/else Selection Structure.

**UNIT III** **REPETITION CONTROL STRUCTURES, FUNCTIONS AND ARRAYS -** Essentials of Repetition – The while, do/while Repetition Structure - Counter-Controlled Repetition – for – Multiple Selection - Switch – Break – Continue – Logical Operators Functions- Definitions - Prototypes –Header Files – Storage Classes – Scope Rules Recursion- Comparing Iteration and Recursion. Arrays – Declaration – Usage – Passing Arrays to Functions.

**UNIT IV** **POINTERS, STRINGS AND AGGREGATE DATA TYPES -** Pointer Variable Declarations and Initialization – Operators – Uses--Pointer Expressions and Pointer Arithmetic – Relationship between Pointers and Arrays – Arrays of Pointers – Pointers to Functions. Fundamentals of Strings and Characters – Character Handling Library - String Handling Library. Structures-Definition – Initialization – Unions – Bitwise Operators – Enumeration Constants.

**UNIT V** **STREAMS, FILES AND PREPROCESSOR -** Streams – Formatting Output with printf –- Formatting Input with scanf. Files – Sequential-Access Files- Creation – Reading –Random-Access Files – Creation – Reading. C Preprocessor – Introduction- #include - #define – Symbolic Constants- Macros- Conditional Compilation - #error - #pragma – Operators # and ## - Line Numbers – Predefined Symbolic Constants.

**REFERENCE BOOKS:**


1. R.G.Dromey, “How to Solve it by Computer”, Pearson Education, 2007.
2. H. M. Deitel and P. J. Deitel, ”C How to Program”, 7th Edition, Pearson Education, 2013.
3. Pradip Dey, Manas Ghosh, “Programming in C”, Oxford University Press, 2007.
4. Cormen,Leiserson, Rivest, Stein, “ Introduction to Algorithms”, McGraw Hill Publishers, 2002.
5. Kernigan Brian W., and Dennis M. Ritchie, “ The C Programming Language”, Second Edition, Prentice Hall, 1988.

**DCS5103** **DATABASE MANAGEMENT SYSTEM**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- Learn the fundamentals of data models and to conceptualize and depict a database system using ER diagram.
- To make a study of SQL and relational database design.
- Understand the internal storage structures using different file and indexing techniques which will help in physical DB design.
- Know the fundamental concepts of transaction processing- concurrency control techniques and recovery procedure.
- Gain a fundamental knowledge about the Storage and Query processing Techniques.

**COURSE OUTCOMES**

Upon completion of the course, the students should be able to

- Design and create tables in database and query them.
- Know how transaction processing is done.
- Analyze and appraise different types of databases.

**UNIT I** **RELATIONAL DATABASES -** Purpose of Database System -– Views of data – Data Models – Database System Architecture –Entity–Relationship model – E-R Diagrams -- Introduction to relational databases -The relational Model –Keys - Relational Algebra – Relational Calculus – SQL fundamentals - Advanced SQL features –Embedded SQL– Dynamic SQL.

**UNIT II** **DATABASE DESIGN -** Functional Dependencies – Non-loss Decomposition – Functional Dependencies – First, Second, Third Normal Forms, Dependency Preservation – Boyce/Codd Normal Form- Multi-valued Dependencies and Fourth Normal Form – Join Dependencies and Fifth Normal Form.

**UNIT III** **TRANSACTIONS -** Transaction Concepts - Transaction Recovery – ACID Properties – System Recovery – Media Recovery – Two Phase Commit - Save Points – SQL Facilities for recovery – Concurrency – Need for Concurrency – Locking Protocols – Two Phase Locking – Deadlock- – Recovery Isolation Levels – SQL Facilities for Concurrency.

**UNIT IV** **IMPLEMENTATION TECHNIQUES -** Overview of Physical Storage Media – Magnetic Disks – RAID – Tertiary storage – File Organization – Organization of Records in Files – Indexing and Hashing –Ordered Indices – B+ tree Index Files – B tree Index Files – Static Hashing – Dynamic Hashing – Query Processing Overview – Catalog Information for Cost Estimation.

**UNIT V** **ADVANCED TOPICS -** Distributed Databases-Architecture-Transaction Processing-Data Warehousing and Mining-Classification-Association rules-Clustering-Information Retrieval-Relevance ranking-Crawling and Indexing the Web- Object Oriented Databases-XML Databases.

**REFERENCE BOOKS:**


1. Abraham Silberschatz, Henry F. Korth, S. Sudharshan, “Database System Concepts”, Sixth Edition, Tata McGraw Hill, 2011.
2. C.J.Date, A.Kannan, S.Swamynathan, “An Introduction to Database Systems”, Eighth Edition, Pearson Education, 2006.
3. Ramez Elmasri, Shamkant B. Navathe, “Fundamentals of Database Systems”, Fifth Edition , Pearson, 2008.
4. Raghu Ramakrishnan, “Database Management Systems”, Fourth Edition, Tata McGraw Hill, 2010.
5. G.K.Gupta,”Database Management Systems”, Tata McGraw Hill, 2011.

**DCS5104** **SOFTWARE ENGINEERING**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- To provide information about wider engineering issues that form the background to develop complex, evolving (software-intensive) systems.
- To plan a software engineering process to account for quality issues and non-functional requirements.
- To employ a selection of concepts and techniques to complete a small-scale analysis and design in mini projects.
- To impart knowledge to translate requirement specifications into a design, and then realize that design practically, all using an appropriate software engineering methodology.
- To provide basic knowledge about software project management.

**COURSE OUTCOMES**

Upon completion of the course, the students should be able to

- Familiar with basic concepts of Software design and implementation.
- Perform software testing on various applications.
- Understand and apply various software metrics on software quality products.

**UNIT I** **INTRODUCTION -** Software Engineering – Product and process – process models - Waterfall Life cycle model – Spiral Model – Prototype Model – fourth Generation Techniques – Agile methods.

**UNIT II** **REQUIREMENT ANALYSIS -** Software Requirements Analysis and Specification – Software Requirements – Problem Analysis – Requirements Specification – Validation – Metrics – Summary.

**UNIT III** **SOFTWARE DESIGN -** Abstraction – Modularity – Software Architecture – Cohesion – Coupling – Various Design Concepts and notations – Real time and Distributed System Design – Documentation – Dataflow Oriented design – Designing for reuse – Programming standards.

**UNIT IV** **SOFTWARE TESTING -** Coding – Programming Practice – Top-down and Bottom-up - structured programming – Information Hiding – Programming style – Internal Documentation Verification – Code Reading – Static Analysis – Symbolic Execution – Code Inspection or Reviews – Unit Testing – Fundamentals – Functional Testing versus structural Testing Coding.

**UNIT V** **SOFTWARE MAINTENANCE AND SOFTWARE METRICS -** Need for Software maintenance – Maintenance models - SCM – Version Control – SCM process – Software Configuration Items – Taxonomy – Basics of Case tools - Scope of Software Metrics – Classification of metrics – Measuring Process and Product attributes – Direct and Indirect measures – Reliability – Software Quality Assurance – Standards.

**REFERENCE BOOKS:**


1. Pankaj Jalote, “An Integrated Approach to Software Engineering”, Third Edition, Narosa publications, 2011.
2. Ian Sommerville, “Software engineering”, Ninth Edition, Pearson Education Asia, 2010.
3. Roger S. Pressman, “Software Engineering – A Practitioner’s Approach”, Seventh Edition, Tata McGraw-Hill International Edition, 2009.

**DCS5105 MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE**

| C | M |
|---|---|
| 4 | 100 |

**COURSE OBJECTIVES**

To introduce mathematical logic, combinatorial and counting techniques, Algebraic structures, Finite state system and grammar as Mathematical Foundation of computer Science so as to understand algorithms, computability and other theoretical aspects of Computer science.

**COURSE OUTCOMES**

Upon completion of the course, the students should be able to

- Understand mathematical logic and to develop analytical solutions for logical problems and they will be equipped with counting techniques to Solve combinatorial problems.
- Comprehend the algebraic structure and formal languages with their applications to handle abstract generalizations and computability.

**UNIT I** **LOGIC -** Statements - Connectives - Truth Tables - Normal Forms - Predicate Calculus – Inference -Theory for Statement Calculus**.**

**UNIT II** **COMBINATORICS -** Permutations and Combinations - Mathematical Induction - Pigeonhole principle - Principle of Inclusion and Exclusion - Recurrence relations - Solution by generating functions and characteristics equations.

**UNIT III** **ALGEBRAIC STRUCTURES -** Groups - Cyclic group - Permutation group (Sn and Dn) - Substructures - Homomorphism -Cosets and Lagrange's Theorem - Normal Subgroups - Rings and Fields (definition and examples).

**UNIT IV** **LATTICES -** Partial order relation – Posets - Hasse diagram - Lattices - Special Lattices - Boolean Algebra.

**UNIT V** **FINITE STATE AUTOMATA AND GRAMMARS -** Finite state automata - Deterministic and non-deterministic model - languages accepted by Finite State Automata - Regular expressions - Context-free grammars - Derivation trees.

**REFERENCE BOOKS:**


1. Trembley.J.P. and Manohar R., "Discrete Mathematical Structures with Applications to Computer Science", Tata McGraw – Hill Publishing Company Limited, New Delhi. Reprinted in 2007.
2. Grimaldi R.P. and Ramana B.V., "Discrete and Combinatorial Mathematics", Pearson Education, Reprinted in 2006. (5th Edition).
3. Hopcroft J.E. and Ullman J.D., "Introduction to Automata, Languages and Computation", Narosa Publishing House, 1987

**DCS 5111** **PROGRAMMING LAB**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- To practice the syntax of C.
- To be exposed to the file processing techniques of C.
- To be familiarized with control structures, functions, arrays and files.

**COURSE OUTCOMES**

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

- Design and implement C programs for any given problem.
- Understand an existing program and modify it as per the requirements.
- Identify the errors in a C program.
- Produce the output of a C program by actually executing it.

**EXPERIMENTS IN THE FOLLOWING TOPICS**:

- Non-iterative control structures.
- Iterative control structures and arrays.
- Functions with parameters.
- Functions with arrays, structures as arguments.
- Character and String handling Libraries.
- Files – Sequential access and random access.
- Preprocessor directives for other features like macros, conditional compilation.

**DCS5112** **DATABASE MANAGEMENT SYSTEM LAB**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- Understand the concepts of DBMS practically.
- To familiarize with SQL queries.
- To write stored procedures in DBMS.
- Learn front end tools and to integrate them with databases.

**COURSE OUTCOMES**

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

- Design and Implement databases practically.
- Formulate complex queries using SQL and execute them.
- Design and Implement applications that have GUI and access databases for backend connectivity.

**EXPERIMENTS IN THE FOLLOWING TOPICS**:

- Data Definition, Manipulation of Tables and Views
- Database Querying – Simple queries, Nested queries, Sub queries and Joins
- Triggers
- Transaction Control
- Embedded SQL
- Database Connectivity with Front End Tools
- Front End Tools / Programming Languages
- High level language extensions - PL/SQL Basics
- Procedures and Functions
- Database Design and Implementation (Case Study)

## SEMESTER – II

**DCS5201** **COMPUTER NETWORKS**

| C | M |
|---|---|
| 4 | 100 |

### COURSE OBJECTIVES

- Understand data communication techniques.
- To know network Fundamentals.
- Understand Network layers and its functionalities.

### COURSE OUTCOMES

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

- Trace the flow of information from one node to another node in the network.
- Identify the component required to build different types of networks.
- Understand the division of network functionalities into layers.
- Identify solution for each functionality at each layer.
- Choose the required functionality at each layer for given application.

**UNIT I** **INTRODUCTION -** Communication model – Data communications and Networking – Data transmission concepts and terminology – Transmission media –Data Encoding Techniques – Digital Data communication Techniques- Data link Control Protocols.

**UNIT II** **NETWORK FUNDAMENTALS -** Protocol architecture – OSI – TCP/IP – LAN Architecture – Topologies – MAC – Ethernet, Fast Ethernet, Token ring, FDDI, Wireless LANS : 802.11/ Wi-Fi/Bluetooth/WiMAX.

**UNIT III** **NETWORK LAYER -** Network layer functions – Switching concepts – Circuit switching networks – Packet Switching – Routing – Internetworking concepts – IP – Unreliable connectionless delivery – Datagrams – Routing IP datagrams – ICMP.

**UNIT IV** **TRANSPORT LAYER -** Transport layer functions – User Datagram Protocol – Transmission Control Protocol – Reliable Delivery Service – Connection Establishment – Flow Control – Congestion Control – Queuing disciplines – Congestion Avoidance.

**UNIT V** **APPLICATIONS -** Domain Name System(DNS) – Telnet – rlogin – FTP – SMTP – MIME – IMAP – HTTP – SNMP – Security.

### REFERENCES


1. Larry L. Peterson & Bruce S. Davie, “Computer Networks - A systems Approach”, 5th Edition, Morgan Kaufmann, 2012.
2. James F. Kurose, Keith W. Ross, “Computer Networking: A Top-Down Approach”, Sixth Edition, Addison-Wesley, 2008.
3. William Stallings, “Data and Computer Communications”, Ninth Edition, PHI, 2004.
4. Andrew S.Tanenbaum, “Computer Networks”, Tata McGraw Hill, 3rd Edition, 2001.

**DCS5202** **OBJECT ORIENTED PROGRAMMING**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- Understand the OO paradigm.
- To be aware of the OO design technique.
- To learn the syntax of C++.
- To be exposed to the file processing and exception handling techniques of C++.
- To be familiarized with the Standard Template Library.

**COURSE OUTCOMES**

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

- Design and implement C++ programs for any given problem.
- Understand an existing program and modify it as per the requirements.
- Identify the errors in a C++ program.
- Identify the output of a C++ program without actually executing it.
- Write generic programs using STL.

**UNIT I** **FUNDAMENTALS -** Object–Oriented Programming concepts – Encapsulation – Programming Elements – Program Structure – Enumeration Types –– Functions and Pointers – Function Invocation – Overloading Functions – Scope and Storage Class – Pointer Types – Arrays and Pointers – Call–by–Reference – Assertions – Standard template library.

**UNIT II** **IMPLEMENTING ADTS AND ENCAPSULATION -** Aggregate Type struct – Structure Pointer Operators – Unions – Bit Fields – Data Handling and Member Functions – Classes – Constructors and Destructors – Static Member – this Pointer – reference semantics – implementation of simple ADTs.

**UNIT III** **POLYMORPHISM -** ADT Conversions – Overloading – Overloading Operators – Unary Operator Overloading – Binary Operator Overloading – Function Selection – Pointer Operators – Visitation – Iterators – containers – List – List Iterators.

**UNIT IV** **TEMPLATES -** Template Class – Function Templates – Class Templates – Parameterizing – STL – Algorithms – Function Adaptors.

**UNIT V** **INHERITANCE -** Derived Class – Typing Conversions and Visibility – Code Reuse – Virtual Functions – Templates and Inheritance – Run–Time Type Identifications – Exceptions – Handlers – Standard Exceptions.

**REFERENCE BOOKS:**


1. Ira Pohl, “Object–Oriented Programming Using C++”, Pearson Education, Second Edition, 2003.
2. Stanley B.Lippman, Josee Lajoie, “C++ Primer”, Pearson Education, Third Edition, 2004.
3. Kamthane, “Object Oriented Programming with ANSI and Turbo C++”, Person Education, Third Edition, 2005.
1. Bhave , “ Object Oriented Programming With C++”, Pearson Education , 2004.

**DCS5203** **DATA STRUCTURES AND ALGORITHMS**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- Gain comprehensive introduction of common data structures, and algorithm design and analysis.
- To master the design of tree, sets and graph structures and its applications.
- Learn about sorting techniques and understand how common computational problems can be solved efficiently on a computer.

**COURSE OUTCOMES**

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

- Describe, explain, and use abstract data types including stacks, queues and lists.
- Design and Implement Tree data structures and Sets.
- Design algorithms using graph structure to solve real-life problems.
- Implement a variety of algorithms for sorting, including insertion sort, selection sort, merge sort, quick sort, and heap sort.
- Describe the asymptotic performance and algorithm design techniques studied in this course and understand the practical implications of that information.

**UNIT I** **BASIC DATA STRUCTURES -** From Problems to programs - Abstract Data Types - Data Types, Data Structures, and Abstract Data Types - The Running Time of a program - Calculating the Running Time of a program - Good Programming Practice; **Basic Data Types:** The Data Type "List" - Implementation of Lists – Stacks – Queues – Mappings - Stacks and Recursive Procedures.

**UNIT II** **TREES & SETS - Trees:** Basic Terminology - The ADT Tree - Implementation of Trees - Binary Trees; **Basic operations on sets:** Introduction to Sets - An ADT with Union, Intersection, and Difference - A Bit-Vector Implementation of Sets; **Advanced Set Representation Methods:** Binary Search Trees - Time Analysis of Binary Search Tree operations – Tries - Balanced Tree Implementations.

**UNIT III** **GRAPHS - Directed Graphs:** Basic Definitions - Representations of Directed Graphs - The Single-Source Shortest Paths Problem - The All-Pairs Shortest Path Problem - Traversals of Directed Graphs - Directed Acyclic Graphs - Strong Components; **Undirected Graphs:** Definitions - Minimum-Cost Spanning Trees – Traversals - Articulation Points and Biconnected Components - Graph Matching.

**UNIT IV** **SORTING & ALGORITHM ANALYSIS - Sorting:** The Internal Sorting Model - Some Simple Sorting Schemes - Quick Sort - Heap Sort - Bin Sorting - A Lower Bound for Sorting by Comparisons - Order Statistics; **Algorithm Analysis Techniques:** Efficiency of Algorithms - Analysis of Recursive programs - Solving Recurrence Equations - A General Solution for a Large Class of recurrences.

**UNIT V** **ALGORITHM DESIGN TECHNIQUES -** Algorithm Design Techniques: Divide-and-Conquer Algorithms - Dynamic Programming - Greedy Algorithms – Backtracking - Local Search Algorithms.

**REFERENCE BOOKS:**


1. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, "Data Structures and Algorithms", Pearson Education, Reprint 2006.
2. Robert Sedgewick and Kevin Wayne, "Algorithms", Fourth Edition, Pearson Education, 2011.
3. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", Third Edition, PHI Learning pvt.Limited,2012.
4. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C++", 2nd edition, Pearson Education, 2005.

**DCS5204** **OPERATING SYSTEM**

| C | M |
|---|---|
| 4 | 100 |

**COURSE OBJECTIVES**

- Learn the Operating System basics.
- Study the process management of Operating system.
- Gain knowledge in the storage management and I/O systems of Operating system.

**COURSE OUTCOMES**

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

- Ability to discuss on the basics of OS.
- In depth knowledge in process management, memory management and I/O Management of various operating systems.
- To explore the case studies with various operating systems.

**UNIT I** **OPERATING SYSTEMS OVERVIEW -** Operating system – Types of Computer Systems - Computer-system operation – I/O structure – Hardware Protection - System components – System calls – System programs – System structure - Process concept – Process scheduling – Operations on processes – Cooperating processes – Inter process communication – Communication in client-server systems - Multithreading models – Threading issues.

**UNIT II** **PROCESS MANAGEMENT -** Scheduling criteria – Scheduling algorithms – Multiple-processor scheduling – Real time scheduling – Algorithm Evaluation – Process Scheduling Models - The critical-section problem – Synchronization hardware – Semaphores – Classic problems of synchronization – critical regions – Monitors - System model – Deadlock characterization – Methods for handling deadlocks – Recovery from deadlock.

**UNIT III** **STORAGE MANAGEMENT -** Memory Management – Swapping – Contiguous memory allocation – Paging – Segmentation – Segmentation with paging. Virtual Memory: Background – Demand paging – Process creation – Page replacement – Allocation of frames – Thrashing.

**UNIT IV** **I/O SYSTEMS -** File concept – Access methods – Directory structure – File-system mounting – Protection - Directory implementation – Allocation methods – Free-space management - Disk scheduling – Disk management – Swap-space management.

**UNIT V** **CASE STUDY -** The Linux System - History – Design Principles – Kernel Modules – Process Management – Scheduling – Memory management – File systems – Input and Output – Inter-process Communication – Network Structure – Security – Windows 7 - History – Design Principles – System Components – Environmental subsystems – File system – Networking.

**REFERENCE BOOKS:**


1. Abraham Silberschatz, Peter B. Galvin and Greg Gagne, “Operating System Concepts”, Ninth Edition, John Wiley and Sons Inc 2012.
2. Andrew S. Tanenbaum, “Modern Operating Systems”, Second Edition, Addison Wesley, 2001.
3. Gary Nutt, “Operating Systems”, Second Edition, Addison Wesley, 2001.
4. H M Deital, P J Deital and D R Choffnes, “Operating Systems”, Pearson Education, 2004.

**DCS5205** **SOFTWARE PROJECT MANAGEMENT**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- Understand the cost evaluation techniques.
- Learn the concepts of project planning and monitoring.
- Understand the concepts of organizing teams for software projects.

**COURSE OUTCOMES**

- To perform planning and scheduling activities.
- Ability to draw activity network.
- Ability to manage people and project.

**UNIT I** **INTRODUCTION TO SOFTWARE PROJECT MANAGEMENT -** Project Definition – Contract Management – Activities Covered By Software Project Management – Overview of Project Planning – Stepwise Project Planning.

**UNIT II** **PROJECT EVALUATION -** Strategic Assessment – Technical Assessment – Cost Benefit Analysis – Cash Flow Forecasting – Cost Benefit Evaluation Techniques – Risk Evaluation.

**UNIT III** **ACTIVITY PLANNING -** Objectives – Project Schedule – Sequencing And Scheduling Activities – Network Planning Models – Forward Pass – Backward Pass – Activity Float – Shortening Project Duration – Activity On Arrow Networks – Risk Management – Nature Of Risk – Types Of Risk – Managing Risk – Hazard Identification – Hazard Analysis – Risk Planning And Control.

**UNIT IV** **MONITORING AND CONTROL -** Creating Framework – Collecting The Data – Visualizing Progress – Cost Monitoring – Earned Value – Prioritizing Monitoring – Getting Project Back To Target – Change Control – Managing Contracts – Introduction – Types Of Contract – Stages In Contract Placement – Typical Terms Of A Contract – Contract Management – Acceptance.

**UNIT V** **MANAGING PEOPLE AND ORGANIZING TEAMS -** Introduction – Understanding Behavior – Organizational Behaviour: A Background – Selecting The Right Person For The Job – Instruction In The Best Methods – Motivation – The Oldham – Hackman Job Characteristics Model – Working In Groups – Becoming A Team – Decision Making – Leadership – Organizational Structures – Stress – Health And Safety – Case Studies.

**REFERENCE BOOKS:**


1. Bob Hughes and MikeCotterell “Software Project Management”, Fifth Edition, TATA McGraw Hill Edition 2010.
2. Ramesh, Gopalaswamy: "Managing Global Projects ", Tata McGraw Hill, 2001.
3. Royce,” Software Project Theory”, Pearson Education, 1999.
4. P.Jalote, “Software Project Management In Practice”, Pearson Education, 2000.

**DCS5211** **OBJECT ORIENTED PROGRAMMING LAB**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- To implement the OO design technique.
- To learn the syntax of C++ and features of C++.
- To be exposed to the file processing and exception handling techniques of C++.
- To be familiarized with the Standard Template Library.

**COURSE OUTCOMES**

- Ability to write C++ programs
- Implement the Stack & queue structures using C++
- Ability to create Class Template files

**EXPERIMENTS IN THE FOLLOWING TOPICS:**

- Write a C++ Program to illustrate Enumeration and Function Overloading
- Write a C++ Program to illustrate Scope and Storage class
- Implementation of ADT such as Stack and Queues
- Write a C++ Program to illustrate the use of Constructors and Destructors and Constructor Overloading
- Write a Program to illustrate Static member and methods
- Write a Program to illustrate Bit fields
- Write a Program to overload as binary operator, friend and member function
- Write a Program to overload unary operator in Postfix and Prefix form as member and friend function
- Write a Program to illustrate Iterators and Containers
- Write a C++ Program to illustrate function templates
- Write a C++ Program to illustrate template class
- Write C++ Programs and incorporating various forms of Inheritance
- Write a C++ Program to illustrate Virtual functions
- Exception Handling

**DCS5212** **DATA STRUCTURES AND ALGORITHMS LAB**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- To develop skills in design and implementation of data structures and their applications.
- To learn and implement linear, non linear and tree data structures
- To learn Set ADT and Graph data structures and its applications
- To study, implement and analyze of different sorting techniques.

**COURSE OUTCOMES**

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

- Work with basic data structures that are suitable for problems to be solved efficiently.
- Implementation of linear, tree, and graph structures and its applications.
- Implementation of various sorting techniques its algorithm design and analysis.

**EXPERIMENTS IN THE FOLLOWING TOPICS:**

- Abstract Data type Implementation of List, Stack and Queues.
- Tree ADT
- Tries Implementation
- Set ADT- Bit Vector Implementation
- Graph Representations
- Graph Traversals
- Shortest Path Implementation
- Spanning Tree Implementation
- Sorting Algorithms
- Implementation of Algorithms using Dynamic Programming, Backtracking

## SEMESTER – III

**DCS5301** **INFORMATION SECURITY**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- Understand the security analysis principles
- Able to become familiar with Security Design.

**COURSE OUTCOMES**

Able to Develop

- Security Models using SDLC
- Implement Logical and Physical Security
- Perform Recovery using Disaster Recovery Techniques

**UNIT I** **INTRODUCTION -** Information Security Concepts - Critical Characteristics of Information - Components of an Information System - The CIA Triad, Securing the Components - Balancing Security and Access - The SDLC - Moving towards a Secure SDLC process

**UNIT II** **SECURITY INVESTIGATION -** Need for Security, Business Needs, Threats, Vulnerabilities, Attacks, Legal and Ethical issues in Security, Evolving a code of Professional Conduct for Information Security Professionals.

**UNIT III** **SECURITY ANALYSIS -** Risk Management : Identifying, Assessing, Mitigating, Transferring and Accepting Risk, Issues of Due Care and Due.

**UNIT IV** **LOGICAL DESIGN -** Blueprint for Security, Information Security Poicy, Standards and Practices, ISO 25001/ISO 17799, NIST Models, VISA International Security Model, Design of Security Architecture, Planning for Continuity.

**UNIT V** **PHYSICAL DESIGN -** Security Technology, IDS, Scanning and Analysis Tools, Cryptography, Access Control Devices, Physical Security, Security and Personnel.

**UNIT VI** **BUSINESS CONTINUITY & DISASTER RECOVERY -** The need for BC and DR Process, The BIA process, Understanding RTO and RPO as a business driven metrics, Selecting the appropriate BC and DR strategies, Testing and Maintenance of BC and DR Plans.

**REFERENCE BOOKS:**


1. Michael E Whitman and Herbert J Mattord, “Principles of Information Security”, Vikas Publishing House, New Delhi, 2010
2. Micki Krause, Harold F. Tipton, “ Handbook of Information Security Management”, Vol 1-3 CRC Press LLC, 2009.
3. Matt Bishop, “ Computer Security Art and Science”, Pearson/PHI, 2008.

**DCS5302** **DATA WAREHOUSING AND MINING**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- Understand Data mining principles and techniques and Introduce DM as a cutting edge business intelligence.
- Expose the students to the concepts of Datawarehousing Architecture and Implementation.
- Study the overview of developing areas – Web mining, Text mining and ethical aspects of Data mining.
- Identify Business applications and Trends of Data mining.

**COURSE OUTCOMES**

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

- Evolve Multidimensional Intelligent model from typical system.
- Discover the knowledge imbibed in the high dimensional system.
- Evaluate various mining techniques on complex data objects.

**UNIT I** **DATA WAREHOUSE -** Data Warehousing - Operational Database Systems vs Data Warehouses - Multidimensional Data Model - Schemas for Multidimensional Databases – OLAP operations – Data Warehouse Architecture – Indexing – OLAP queries & Tools.

**UNIT II** **DATA MINING & DATA PREPROCESSING -** Introduction to KDD process – Knowledge Discovery from Databases - Need for Data Preprocessing – Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization and Concept Hierarchy Generation.

**UNIT III** **ASSOCIATION RULE MINING -** Introduction - Data Mining Functionalities - Association Rule Mining - Mining Frequent Itemsets with and without Candidate Generation - Mining Various Kinds of Association Rules - Constraint-Based Association Mining.

**UNIT IV** **CLASSIFICATION & PREDICTION -** Classification vs Prediction – Data preparation for Classification and Prediction – Classification by Decision Tree Introduction – Bayesian Classification – Rule Based Classification – Classification by Back propagation – Support Vector Machines – Associative Classification – Lazy Learners – Other Classification Methods – Prediction – Accuracy and Error Measures – Evaluating the Accuracy of a Classifier or Predictor – Ensemble Methods – Model Section.

**UNIT V** **CLUSTERING -** Cluster Analysis: - Types of Data in Cluster Analysis – A Categorization of Major Clustering Methods – Partitioning Methods – Hierarchical methods – Density-Based Methods – Grid-Based Methods – Model-Based Clustering Methods – Clustering High-Dimensional Data – Constraint-Based Cluster Analysis – Outlier Analysis.

**REFERENCE BOOKS:**


1. Jiawei Han and Micheline Kamber “Data Mining Concepts and Techniques” Second Edition, Elsevier, Reprinted 2011.
2. K.P. Soman, Shyam Diwakar and V. Ajay “Insight into Data mining Theory and Practice”, Easter Economy Edition, Prentice Hall of India, 2006.
3. G. K. Gupta “Introduction to Data Mining with Case Studies”, Easter Economy Edition, Prentice Hall of India, 2006.
4. Pang-Ning Tan, Michael Steinbach and Vipin Kumar “Introduction to Data Mining”, Pearson Education, 2007

**DCS5303** **WEB PROGRAMMING**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- To understand the basics of HTML.
- To learn the concepts of XML related technologies.
- To learn the fundamentals of java.
- To understand the importance of server side programming and web development.

**COURSE OUTCOMES**

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

- Design and implementation of web forms and client side validation.
- XML authoring, Parsing, and related technologies.
- Object oriented concept programming using Java.
- Design and development of GUI based applications using Swing components.
- Design and development of servlet and JSP application with database connectivity.

**UNIT I** **HTML AND JAVA SCRIPT -** World Wide Web – XHTML - Cascading Style Sheet - JavaScript – java script objects - Date – Array – pattern matching using regular expressions – Dynamic documents with java script – HTML 5 – new features.

**UNIT II** **XML TECHNOLOGIES -** XML – validating XML - DTD – XML schema – XPath – XLink – parsing XML using DOM – parsing XML using SAX – transforming XML with XSL – Integrating XML with database – AJAX – RSS – JSON

**UNIT III** **JAVA BASICS -** Overview of Java – Java Fundamentals – Classes, Objects and Methods – Arrays and Array Lists – String – String Builder – Regular expressions – class pattern – class matcher - Packages and Interfaces – Exception Handling.

**UNIT IV** **JAVA GUI AND DATABASE CONNECTIVITY -** Generic classes – Generic methods – Applets – Applet life cycle methods – Applets based GUI – GUI components – Basic of Swings – Accessing database with JDBC - basics.

**UNIT V** **SERVER SIDE SCRIPT -** Overview of servlets – Servlet API – servlet life cycle – servlet configuration – running servet with database connectivity - servlet support for cookies – Session tracking – Java server pages – JSP Case study/ Applications – Developing Dynamic, Data driven web sites.

**REFERENCE BOOKS:**


1. Robert W. Sebesta, “Programming with World Wide Web”, Pearson Education, 2008.
2. Paul Deitel and Harvey Daitel, “Java – How to program”, Ninth Edition, PHI, 2012.
3. Kogent Solutions, “Java 6 Programming Black book”, Dreamtech Press, 2007

**DCS5304** **OBJECT ORIENTED ANALYSIS AND DESIGN**

| C | M |
|---|---|
| 4 | 100 |

**COURSE OBJECTIVES**

- Understand the basics of object oriented analysis and design.
- Learn UML models and tools.
- To apply design patterns to various applications.

**COURSE OUTCOMES**

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

- Familiarize with the topics of object oriented System designs.
- Design patterns using UML.
- Apply design patterns to various applications.

**UNIT I** **INTRODUCTION -** An overview – Object basics – Object state and properties – Behavior – Methods – Messages – Information hiding – Class hierarchy – Relationships – Associations – Aggregations- Identity – Dynamic binding – Persistence – Metaclasses – Object oriented system development life cycle.

**UNIT II** **METHODOLOGY AND UML -** Introduction – Survey – Rumbugh, Booch, Jacobson methods – Patterns – Frameworks – Unified approach – Unified modeling language – Static and Dynamic models – UML diagrams – Class diagram – Usecase diagrams – Dynamic modeling – Model organization – Extensibility.

**UNIT III** **OBJECT ORIENTED ANALYSIS -** Identifying Usecase – Business object analysis – Usecase driven object oriented analysis – Usecase model – Documentation – Classification – Identifying object, relationships, attributes, methods – Super-sub class – A part of relationships Identifying attributes and methods – Object responsibility.

**UNIT IV** **OBJECT ORIENTED DESIGN -** Design process – Axions – Cololaries – Designing classes – Class visibility – Refining attributes – Methods and protocols – Object storage and object interoperability – Databases – Object relational systems – Designing interface objects – Macro and Micro level processes – The purpose of a view layer interface.

**UNIT V** **SOFTWARE QUALITY -** Quality assurance – Testing strategies – Object orientation testing – Test cases – Test Plan – Debugging principles – Usability – Satisfaction – Usability testing – Satisfaction testing.

**REFERENCE BOOKS:**


1. Ali Bahrami, “Object Oriented System Development”, McGraw Hill International Edition, Second reprint 2008.
2. Craig Larman, “Applying UML and Patterns”, 2nd Edition, Pearson, 2002.
3. Grady Booch, James Rumbaugh, Ivar Jacobson, “The Unified Modeling Language User Guide”, Addison Wesley Long man, 1999.
4. Bernd Bruegge, Allen H. Dutoit, “Object Oriented Software Engineering using UML, Patterns and Java”, Pearson 2004.

**DCS5311** **WEB PROGRAMMING LAB**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- To learn web page creation.
- To understand the real time requirements of web page such as validation, use of DOM, role of XML.
- To understand OOP concepts and basics of Java language.
- To learn and use client server architecture based applications.
- To explore server side functionalities of an application.

**COURSE OUTCOMES**

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

- Make Web site creation and validation.
- Work with XML based technologies.
- Develop simple console application using Java.
- Develop GUI application using Swing and Applet.
- Build web based applications using JDBC, Servlet / JSP.

**EXPERIMENTS IN THE FOLLOWING TOPICS:**

- Creation of web pages having dynamic contents and validation using java script
- Creation of XML file and validation using XML schema and generation of XML using tools
- Simple xml based applications using DOM, SAX and XSL
- Basic Java programming covering objects, inheritance, polymorphism, interfaces, packages and exception handling
- String handling programs and regular expression programs
- Creation of applet based GUI's
- Application involving applet based GUI, JDBC, Servlet, JSP, cookies and session tracking.

**DCS5312** **SOFTWARE DEVELOPMENT LAB**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- Understand project planning
- Able to Analysis and Design Application
- Able to Implement and Test Software

**COURSE OUTCOMES**

- Able to Draw DFD and UML Diagrams
- Able to Write programs for Design
- Generate and Execute Test cases.

**EXPERIMENTS IN THE FOLLOWING TOPICS:**

Apply the following to typical application problems:

1. Project Planning
2. Software Requirement Analysis
3. Software Estimation
4. Software Design
5. Data Modelling & Implementation
6. Software Testing
7. Software Debugging

A possible set of applications may be the following:

a. Library System
b. Student Marks Analyzing System
c. Text Editor.
d. Create a dictionary.
e. Telephone dictionary.
f. Simulator Software for Parallel Processing Operation.
g. Inventory System.

## LIST OF ELECTIVES

**DCS5001** **MOBILE COMPUTING**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

Able

- To understand the principles of wireless communication
- To understand wireless LAN protocols
- To become familiar with mobile Routing protocols

**COURSE OUTCOMES**

Able to Develop Mobile Application

- Implement MAC protocols
- Able to perform Optimal Routing

**UNIT I** **WIRELESS COMMUNICATION -** Challenges of Wireless Transmission - Multi-carrier modulation - Spread Spectrum - Satellite Communication - Broadcast systems - Multiplexing - FDMA, TDMA and CDMA - Cellular organization of mobile telephone networks - Operation of cellular networks - Frequency Reuse - Tessellation - Handoff - Capacity Improvement.

**UNIT II** **WIRELESS NETWORKS -** IEEE 802.11 Wireless LAN - Architecture - Modes of Operation - CSMA/CA and its variants - Wireless LAN security - Bluetooth networks - Generation of cellular networks - Overview of GSM - GPRS Network Architecture and Operations - UMTS and IMT 2000 - Packet Switching Domain - Core Network - Radio Access Network - LTE - Control Plane - User Plane.

**UNIT III** **L3 AND L4 WIRELESS PROTOCOLS -** Mobile IP - Mobility features in IPv6 - Proactive and reactive ad hoc routing protocols - DSDV, DSR and AODV - Limitations of Traditional TCP in wireless networks - TCP improvements for Wireless Networks – Indirect TCP, Snoop TCP, Mobile TCP - Security issues in network layer and transport layer.

**UNIT IV** **MOBILE COMPUTING PLATFORM -** PDA - Device characteristics and Software components - Smart Phone - Convergence of Mobile devices - J2ME - Modes, Data store, GUI support - HTTP Connection Interface Push Registry - Application development using Android APIs - Palm OS Architecture and Program Development - Overview of other mobile Operating Systems.

**UNIT V** **MOBILE INTERNET -** WAP - WAP Gateways - WML - VoiceXML - Mobile Messaging - Multimedia Messaging Service - Synchronized Multimedia Integration Language - Application Servers - Internet portals - Device management - Synchronization Models - Communication to Servlets and Web Services - Location aware Mobile computing - IP Multimedia Subsystem.

**REFERENCE BOOKS:**


1. Asoke Talukder, Hasan Ahmed, Rupa Yavagal, “Mobile Computing: Technology, Applications and Services Creation”, Second Edition, TMH, 2010.
2. William Stallings, “Wireless Communication and Networks”, Pearson, 2009.
3. Jochen Schiller, “Mobile Communications”, Second Edition, Pearson, 2009
4. Uwe Hansmann et al, “Principles of Mobile Computing”, Springer, 2003
5. Ivan Stojmenovic, “Handbook of Wireless Networks and Mobile Computing”, Wiley, 2002

**DCS5002** **XML AND WEB SERVICES**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- Able to understand XML Data Representation
- Create Web Services
- Develop Security Mechanisms for XML Data.

**COURSE OUTCOMES**

- Able to Design Web Databases
- To compose and Deploy Web Services
- Implement Security principles

**UNIT I** **XML FUNDAMENTALS -** XML – structuring with schema DTD – XML Schema – XML Processing DOM – SAX – Presental XSL – Transformation XSLT – XPath –XQuery.

**UNIT II** **DISTRIBUTED INFORMATION SYSTEM -** Distributed information system – Design of IB – Architecture of IB – Communication in an IS – Middleware RPC – TP monitors – Object brokers – Message oriented middleware – EAI – EAI Middleware – Workflow –Management – benefits and limitations – Web technologies for Application Integration.

**UNIT III** **WEB SERVICES -** Web Services – Definition – Web Services and EAI – Web Services Technologies – web services Architecture – SOAP – WSDL – UDDI –WS – Addressing – WS – Routing WS- Security –WS –Policy –Web Service invocation framework web services using java – WS using .NET mobile web service.

**UNIT IV** **XML SECURITY -** XML Security and meta framework__XML signature – XML Encryption – SAML – XKMS – WS – Security – RDF – semantic Web service.

**UNIT V** **SERVICE COMPOSITION -** Service Coordination and Composition coordination protocols – WS – Coordination – WS – transaction – RosttaNet – ebXML –WSCI – Service Composition – Service Composition Models – Dependencies between coordination and composition – BPEL – Current trends.

**REFERENCE BOOKS**


1. Gystavo Alonso, Fabio casasi, Hareemi kuno, vijay machiraju, “web Services – concepts, Architecture and Applications”, Springer, 2004.
2. Ron Schmelzer etal “ XML and Web Services”, Pearson Education, 2002.
3. Sandeep chatterjee and james webber,” Developing
4. Enterprise web services: An Architect’s and Guide”, Practice Hall, 2004.
5. Freunk p.coyle,” XML, web Services and the Data Revolution”, Pearson, 2002

**DCS5003** **THEORY OF COMPUTATION**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

Able to

- Understand the equivalence of NFA, DFA and Regular Expression
- Write Grammar for a Language
- Develop a PDA

**COURSE OUTCOMES**

- Able to write programs for Logical Analysis
- Write programs to Syntax Analysis
- Implement stack operations for PDA

**UNIT I** **REGULAR EXPRESSIONS AND LANGUAGES -** Introduction to Formal Proof – Additional Forms of proof – Inductive proofs – Regular Expressions – Regular and Non Regular Languages - Closure Properties of Regular Languages - Proving Languages Not to Be Regular - Decision Properties of Regular Languages.

**UNIT II** **AUTOMATA -** Finite Automata – Deterministic Finite Automata – Non-deterministic Finite Automata – Finite Automata with Epsilon Transitions - Kleene's Theorem –Equivalence and Minimization of Automata - Finite Automata and Regular Expressions.

**UNIT III** **CONTEXT-FREE GRAMMARS AND LANGUAGES** - Context-Free Grammars – Parse Trees – Ambiguity in Grammars and Languages – Phases of a complier - Lexical Analysis – Parsing – Compiler Design using Lexical Analysis and Parsing – Grammars for Natural Language Processing.

**UNIT IV** **PUSHDOWN AUTOMATA AND TURING MACHINES -** Definition– Languages of a Pushdown Automata – Equivalence of Pushdown Automata and Context-Free Grammars - Deterministic Pushdown Automata, Normal forms for Context-Free Grammars – Pumping Lemma for Context-Free Languages - Closure and Decision Properties of Context-Free Languages - Turing Machines – Programming Techniques for Turing Machines - Basic Turing Machine Extensions.

**UNIT V** **UNDECIDABILITY -** Not Recursively Enumerable Language – Recursively Enumerable Undecidable problem– Undecidable Problems about Turing Machines – Post's Correspondence Problem - The classes P and NP - NP-complete problems.

**REFERENCE BOOKS:**


1. H.R.Lewis and C.H.Papadimitriou, "Elements of The theory of Computation", Second Edition, Pearson Education/PHI, 2007.
2. Micheal Sipser, "Introduction of the Theory and Computation", Thomson Brokecole, 2003.
3. Alfred V. Aho, Monica S. Lam, Ravi Sethi, and Jeffrey D. Ullman, "Compilers: Principles, Techniques, & Tools", Second Edition Boston: Addison-Wesley, 2007

**DCS5004 COMPUTER GRAPHICS AND MULTIMEDIA SYSTEMS**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- To understand the basic concepts of graphics designs.
- To familiarize the student with the transformation and projection techniques
- To expose the student to various color models

**COURSE OUTCOMES**

Upon Completion of the course, the students should be able to

- Implement basic graphics transformation and projection techniques.
- Design an application that incorporates different concepts of various color models.
- Apply and explore new techniques in the areas of compression techniques.
- To appreciate the use of multimedia authoring tools and multimedia compression techniques

**UNIT I** **INTRODUCTION -** Overview of Graphics System - Bresenham technique – Line Drawing and Circle Drawing Algorithms - DDA - Line Clipping - Text Clipping.

**UNIT II** **2D TRANSFORMATIONS -** Two dimensional transformations – Scaling and Rotations - Interactive Input methods - Polygons - Splines – Bezier Curves - Window view port mapping transformation.

**UNIT III** **3D TRANSFORMATIONS -** 3D Concepts - Projections – Parallel Projection - Perspective Projection – Visible Surface Detection Methods - Visualization and polygon rendering – Color models – XYZ-RGB-YIQ-CMY-HSV Models - animation – Key Frame systems - General animation functions - morphing.

**UNIT IV** **OVERVIEW OF MULTIMEDIA -** Multimedia hardware & software - Components of multimedia – Text, Image – Graphics – Audio – Video – Animation – Authoring.

**UNIT V** **MULTIMEDIA SYSTEMS AND APPLICATIONS -** Multimedia communication systems – Data base systems – Synchronization Issues – Presentation requirements – Applications – Video conferencing – Virtual reality – Interactive video – video on deman.

**REFERENCE BOOKS:**


1. Hearn D and Baker M.P, “Computer graphics – C Version”, 2nd Edition, Pearson Education, 2004 (unit 1, 2 &3).
2. Ralf Steinmetz, Klara steinmetz, “Multimedia Computing, Communications and Applications”, Pearson education, 2004 (unit 4 & 5).
3. Siamon J. Gibbs and Dionysios C. Tsichritzis, “Multimedia programming”, Addison Wesley, 1995.
4. John Villamil, Casanova and Leony Fernanadez, Eliar, “Multimedia Graphics”, PHI, 1998.

**DCS5005** **ETHICAL HACKING & CYBER FORENSICS**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- To understand the hacking techniques of computer forensics.
- To learn about data recovery methods.
- To identity the threats in computer forensics

**COURSE OUTCOMES**

- Able to distinguish between hackers and normal users.
- To apply the principles of computer forensics for security.
- To implement the data recovery methods.
- To manage threats and the tactics.

**UNIT I** **ETHICAL HACKING -** Foundation for Ethical Hacking-Ethical Hacking in Motion-Hacking Network Hosts-Hacking Operating Systems-Hacking Applications.

**UNIT II** **TYPES OF COMPUTER FORENSICS -** Computer Forensics Fundamentals – Types of Computer Forensics Technology – Types of Vendor and Computer Forensics Services.

**UNIT III** **DATA RECOVERY -** Data Recovery – Evidence Collection and Data Seizure – Duplication and Preservation of Digital Evidence – Computer Image Verification and Authentication.

**UNIT IV** **ELECTRONIC EVIDENCE -** Discover of Electronic Evidence – Identification of Data – Reconstructing Past Events – Networks.

**UNIT V** **THREATS -** Fighting against Macro Threats – Information Warfare Arsenal – Tactics of Military – Tactics of Terrorist and Rogues – Tactics of Private Companies.

**REFERENCE BOOKS:**


1. John R. Vacca, “Computer Forensics”, Firewall Media, 2004.
2. Kevin Beaver, “Hacking For Dummies”, John Wiley & Sons,2012George F Luger, “Artificial Intelligence, structures and strategies for complex problem solving”, Pearson Education, Delhi, 2001.
3. Chad Steel, “Windows Forensics”, Wiley India, 2006
4. Majid Yar, “Cybercrime and Society”, Sage Publications, 2006
5. Robert M Slade, “Software Forensics”, Tata McGrawHill, 2004.

**DCS5006** **SOFTWARE TESTING**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

- To introduce the basics and necessity of Software testing.
- To introduce various testing techniques along with software production.
- To introduce the concepts of Software bugs and its impact.

**COURSE OUTCOMES**

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

- Perform automated testing using test tools.
- Document the testing procedures.

**UNIT I** **INTRODUCTION -** Software Testing background – software bugs- cost of bugs-software testing realities- Testing Axioms – Precision and Accuracy-verification and validation- quality and reliability-testing and quality assurance.

**UNIT II** **SOFTWARE TESTING METHODOLOGY -** Functional testing- Structural testing – Static and Dynamic testing – low level specification test techniques – Equivalence Partitioning – Data testing – State Testing – formal reviews – coding standards and guidelines – code review checklist – data coverage- code coverage.

**UNIT III** **SOFTWARE TESTING TECHNIQUES -** Configuration testing – Compatibility testing – foreign language testing – usability testing – testing the documentation - testing for software security – website testing.

**UNIT IV** **AUTOMATED TESTING AND TEST TOOLS -** Benefits of automation and tools – viewers and monitors – drivers – stubs – stress and load tools – analysis tools- software test automation – random testing – beta testing.

**UNIT V** **TEST DOCUMENTATION -** Goal of Test Planning – test phases – test strategy – resource requirements – test schedule – writing and tracking test cases- Bug tracking systems – metrics and statistics- risks and issues.

**REFERENCE BOOKS:**


1. Glenford J.Myers, Tom Badgett, Corey Sandler, “The Art of Software Testing”,3rd edition, John Wiley & Sons publication, 2012.
2. Ron Patton, “Software testing” , second edition, Pearson education, 2009.
3. Boris Beizer, “Software testing techniques”, Dream Tech Press,2009.
4. Srinivasan Desikan, Gopalaswamy Ramesh, “Software testing- Principles and Practices”, Pearson education, 2009

**DCS5007** **CLOUD COMPUTING**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

Understand

- Cloud Services and Application
- Become familiar with MAP REDUCE Techniques
- Know about Cloud Tools and Programming

**COURSE OUTCOMES**

Able to

- Implement Virtualization Techniques
- Store and Retrieve Data in Cloud
- Provide Security through programs.

**UNIT I** **INTRODUCTION -** Evolution of Cloud Computing –System Models for Distributed and Cloud Computing – NIST Cloud Computing Reference Architecture -IaaS – On-demand provisioning – Elasticity in cloud – Egs of IaaS providers - PaaS – Egs. Of PaaS providers - SaaS – Egs. Of SaaS providers – Public , Private and Hybrid clouds.

**UNIT II** **VIRTUALIZATION -** Basics of virtualization - Types of Virtualization - Implementation Levels of Virtualization - Virtualization Structures - Tools and Mechanisms - Virtualization of CPU, Memory, I/O Devices - Desktop virtualization – Server Virtualization.

**UNIT III** **CLOUD INFRASTRUCTURE -** Architectural Design of Compute and Storage Clouds – Layered Cloud Architecture Development – Design Challenges - Inter Cloud Resource Management – Resource Provisioning and Platform Deployment – Global Exchange of Cloud Resources.

**UNIT IV** **PROGRAMMING MODEL -** Parallel and Distributed programming Paradigms – MapReduce , Twister and Iterative MapReduce – Hadoop Library from Apache – Mapping Applications - Programming Support - Google App Engine, Amazon AWS - Cloud Software Environments - Eucalyptus, Open nebula, OpenStack.

**UNIT V** **SECURITY IN THE CLOUD -** Security Overview – Cloud Security Challenges – Software-as-a-Service Security – Security Governance – Risk Management – Security Monitoring – Security Architecture Design – Data Security – Application Security – Virtual Machine Security.

**REFERENCE BOOKS:**


1. Distributed and Cloud Computing, From Parallel Processing to the Internet of Things by Kai Hwang, Geoffrey C Fox, Jack G Dongarra, Morgan Kaufmann Publishers, 2012.
2. Cloud Computing: Implementation, Management, and Security by John W.Rittinghouse and James F.Ransome : CRC Press 2010
3. Cloud Computing, A Practical Approach by Toby Velte, Anthony Velte, Robert Elsenpeter: TMH, 2018
4. Cloud Application Architectures: Building Applications and Infrastructure in the Cloud: Transactional Systems for EC2 and Beyond (Theory in Practice (O'Reilly)) by George Reese: O'Reilly
5. James E. Smith, Ravi Nair, Virtual Machines: Versatile Platforms for Systems and Processes, Elsevier/Morgan Kaufmann, 2005
6. Katarina Stanoevska-Slabeva, Thomas Wozniak, Santi Ristol, “Grid and Cloud Computing – A Business Perspective on Technology and Applications”, Springer

**DCS5008** **VISUAL PROGRAMMING**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

Able to

- Understand windows Environment
- Write programs in Visual Basic
- Understand the concepts of VC++

**COURSE OUTCOMES**

Able to

- Develop windows Applications
- User Interface Design using VB
- Application Development using VC++

**UNIT I** **WINDOWS PROGRAMMING -** The windows programming Model – Event driven programming – GUI concepts – Overview of Windows programming – Creating and displaying the window – Message Loop – windows procedure – WM_PAINT message – WM_DESTROY message – Data types – Resources – An Introduction to GDI – Device context – Text output – Scroll Bars – Keyboard – Mouse – Menus.

**UNIT II** **VISUAL BASIC PROGRAMMING -** Visual Basic Applications – Form and properties – Variables and Constants – Variant type – Procedure scope – Main – Control statements – control arrays – Creating and using Controls – Menus and Dialogs – Programming fundamentals – Objects and instances – Debugging – Responding to mouse events – Drag and Drag drop events Responding to keyboard events – keypress, keyup, keydown events – Using grid control – Graphics controls – shape and line control – File system controls – Common dialog controls – Processing files – Accessing databases with the data controls.

**UNIT III** **VISUAL C++ PROGRAMMING -** Visual C++ components – Introduction to Microsoft Foundation Classes Library – Getting started with AppWizard – Class Wizard – Event handling – Keyboard and Mouse events - WM_SIZE, WM_CHAR messages - Graphics Device Interface - Pen, Brush, Colors, Fonts - Single and Multiple document interface - Reading and Writing documents - Resources – Bitmaps creation, usage of BMP and displaying a file existing as a BMP.

**UNIT IV** **CONTROLS -** Dialog Based Applications, controls – Animate control, image list, CRect tracker – Tree control – CtabControl – Dynamic controls – slider control – progress control – Inheriting CTreeView – CRicheditView – Modal Dialog, – Modeless Dialog – CColorDialog – CfileDialog.

**UNIT V** **ADVANCED CONCEPTS -** Domain Name System – Email – World Wide Web (HTTP) – Simple Status bars – Splitter windows and multiple views – Dynamic Link Library – Data base Management with ODBC – TCP/IP – Winsock and WinInet, – ActiveX control – creation and usage – Container class.

**REFERENCE BOOKS:**


1. Charles Petzold, "Windows Programming", Microsoft press, 1996
2. J. David Kruglirski, "Programming Microsoft Visual C++", Fifth Edition, Microsoft press, 1998
3. Marion Cottingham "Visual Basic", Peachpit Press, 1999
4. Steve Holzner, "Visual C++ 6 programming", Wiley Dreamtech India Private Ltd., 2003.
5. Kate Gregory "Using Visual C++", Prentice Hall of India Pvt., Ltd., 1999.
6. Herbert Sheildt, "MFC from the Ground Up"
7. Deitel , " Visual Basic 6.0 How To Program", Pearson Education, 1999

**DCS5009** **E COMMERECE**

| C | M |
|---|---|
| 2 | 100 |

**COURSE OBJECTIVES**

Able to Understand

- the TCP/IP networks
- Electronic payment systems
- E-Security principles

**COURSE OUTCOMES**

Able to create

- XML and Web Databases
- Implement Electronic payment systems
- Implement E-Security systems

**UNIT I** Introduction: Infrastructure for Electronic Commerce - Networks - Packet Switched Networks - TCP/IP Internet protocol - Domain name Services - Web Service Protocols - Internet applications - Utility programs - Markup Languages - Web Clients and Servers - Intranets and Extranets - Virtual private Network.

**UNIT II** Core Technology: Electronic Commerce Models - Shopping Cart Technology - Data Mining - Intelligent Agents – Internet Marketing - XML and E-Commerce.

**UNIT III** Electronic Payment Systems: Real world Payment Systems - Electronic Funds Transfer - Digital Payment -Internet Payment Systems – Micro Payments - Credit Card Transactions – Mobile Marketing and Advertisement - Case Studies.

**UNIT IV** Security: Threats to Network Security - Public Key Cryptography - Secured Sockets Layer - Secure Electronic Transaction - Network Security Solutions - Firewalls.

**UNIT V** Inter/Intra Organizations Electronic Commerce: EDI - EDI application in business - legal, Security and Privacy issues - EDI and Electronic commerce - Standards - Internal Information Systems - Macro forces - Internal commerce - Workflow Automation and Coordination - Customization and Internal commerce - Supply chain Management.

**REFERENCE BOOKS:**


1. Ravi Kalakota and Andrew B Whinston , Frontiers of Electronic commerce, Pearson Education, 2003
2. Brian E.Mennecke, Troy J.Strader, “Mobile Commerce: (Soft Cover): Technology,Theory and Applications”, Idea group Inc., IRM Press,2003
3. Pete Loshin, Paul A Murphy , Electronic Commerce, 2nd Edition , Jaico Publishers1996.
4. David Whiteley, e - Commerce : Strategy, Technologies and Applications - McGraw Hill 2000.