

**ANNA UNIVERSITY**  
**CENTRE FOR DISTANCE EDUCATION**  
**MASTER OF COMPUTER APPLICATIONS**  
**REGULATIONS - 2018 CURRICULUM**

**SEMESTER - I**

<b>CODE NO.</b>	<b>COURSE TITLE</b>	<b>CREDITS*</b>	<b>MARKS</b>
DMC5101	Mathematical Foundations of Computer Science	4	100
DMC5102	Problem Solving and Programming	2	100
DMC5103	Database Management System	2	100
DMC5104	Software Engineering	2	100
DMC5105	Computer Organization & Design	2	100
DMC5111	Programming Lab	2	100
DMC5112	Database Management System Lab	2	100
<b>TOTAL</b>		<b>16</b>	<b>700</b>

**SEMESTER - II**

<b>CODE NO.</b>	<b>COURSE TITLE</b>	<b>CREDITS*</b>	<b>MARKS</b>
DMC5201	Computer Networks	4	100
DMC5202	Operating System	2	100
DMC5203	Data Structures and Algorithms	2	100
DMC5204	Computer Graphics and Multimedia Systems	2	100
DMC5205	Object Oriented Programming	2	100
DMC5211	Data Structures using C++ Lab	2	100
DMC5212	Operating System Lab	2	100
<b>TOTAL</b>		<b>16</b>	<b>700</b>

**SEMESTER - III**

<b>CODE NO.</b>	<b>COURSE TITLE</b>	<b>CREDITS*</b>	<b>MARKS</b>
DMC5301	Web Programming	2	100
DMC5302	Object Oriented Analysis and Design	2	100
DMC5303	Data Warehousing and Mining	4	100
DMC5304	Security Practice	2	100
E1	Elective I	2	100
DMC5311	Security Lab	2	100
DMC5312	Web Programming Lab	2	100
<b>TOTAL</b>		<b>16</b>	<b>700</b>

**SEMESTER - IV**

<b>CODE NO.</b>	<b>COURSE TITLE</b>	<b>CREDITS*</b>	<b>MARKS</b>
DMC5401	Unix and Network Programming	4	100
DMC5402	Enterprise Application Development	2	100
DMC5403	.NET Programming	2	100
E2	Elective II	2	100
E3	Elective III	2	100
DMC5411	Enterprise Application Development lab	2	100
DMC5412	.NET Programming Lab	2	100
<b>TOTAL</b>		<b>16</b>	<b>700</b>

**SEMESTER – V**

<b>CODE NO.</b>	<b>COURSE TITLE</b>	<b>CREDITS*</b>	<b>MARKS</b>
DMC5501	Web Services	2	100
DMC5502	Software Project Management	4	100
DMC5503	Mobile Application Development	2	100
DMC5504	Communication Skills	2	100
E4	Elective IV	2	100
DMC5511	Web Services Lab	2	100
DMC5512	Mobile Application Development Lab	2	100
<b>TOTAL</b>		<b>16</b>	<b>700</b>

**SEMESTER – VI**

<b>CODE NO.</b>	<b>COURSE TITLE</b>	<b>CREDITS*</b>	<b>MARKS</b>
E5	Elective V	2	100
DMC5601	Cloud Services	2	100
DMC5611	Project Work	12	200
<b>TOTAL</b>		<b>16</b>	<b>400</b>
<b>Total No. of Credits and Marks</b>		<b>96</b>	<b>3900</b>

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

**ELECTIVES****ELECTIVE – I**

<b>CODE NO.</b>	<b>COURSE TITLE</b>	<b>CREDITS*</b>	<b>MARKS</b>
DMC5001	Advanced Databases	2	100
DMC5002	TCP/IP Protocol Suite	2	100
DMC5003	Software Testing	2	100

**ELECTIVE – II**

<b>CODE NO.</b>	<b>COURSE TITLE</b>	<b>CREDITS*</b>	<b>MARKS</b>
DMC5004	Distributed Systems	2	100
DMC5005	Artificial Intelligence	2	100
DMC5006	Human Resource Management	2	100

**ELECTIVE – III**

<b>CODE NO.</b>	<b>COURSE TITLE</b>	<b>CREDITS*</b>	<b>MARKS</b>
DMC5007	Ethical Hacking and Cyber Forensics	2	100
DMC5008	E-Learning Management	2	100
DMC5009	Collaborative Web Design	2	100

**ELECTIVE – IV**

<b>CODE NO.</b>	<b>COURSE TITLE</b>	<b>CREDITS*</b>	<b>MARKS</b>
DMC5010	M - Commerce	2	100
DMC5011	Game Programming	2	100
DMC5012	Free / Open Source Software	2	100

**ELECTIVE – V**

<b>CODE NO.</b>	<b>COURSE TITLE</b>	<b>CREDITS*</b>	<b>MARKS</b>
DMC5013	Enterprise Resource Planning	2	100
DMC5014	Resource Management Techniques	2	100
DMC5015	Technology Commercialization & Transfer	2	100

**ANNA UNIVERSITY  
CENTRE FOR DISTANCE EDUCATION  
MASTER OF COMPUTER APPLICATIONS  
REGULATIONS - 2018  
SYLLABUS I TO VI SEMESTERS**

**SEMESTER – I**

**DMC5101            MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE**

<b>C</b>	<b>M</b>
<b>4</b>	<b>100</b>

**COURSE OBJECTIVES**

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 - Pigeonholeprinciple - Principle of Inclusion and Exclusion - Recurrence relations - Solution by generating functions and characteristics equations.

**UNIT III            ALGEBRAIC STRUCTURES** - Groups - Cyclic group - Permutation group ( $S_n$  and  $D_n$ ) - 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 andnon-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, Reprint – 2002.

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 ofRepetition – 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 andInitialization – 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", 7<sup>th</sup> 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.

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 – FunctionalDependencies – 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 – SystemRecovery – 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- DataWarehousing 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.



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, TataMcGraw-Hill International Edition, 2009.

C	M
2	100

## COURSE OBJECTIVES

- 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.
- To discuss in detail the operation of the arithmetic and logic unit.
- Study the instruction sets and operation of a processor.
- 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.

## COURSE OUTCOMES

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.

<b>UNIT I</b>	<b>DIGITAL FUNDAMENTALS</b> - Digital systems, binary numbers, octal, hexadecimal conversions, signed binary numbers, complements, logic gates, Boolean algebra , K-maps, standard forms, NAND-NOR implementation.
<b>UNIT II</b>	<b>COMBINATIONAL AND SEQUENTIAL CIRCUITS</b> - Combinational circuits, adder, subtractor, ALU design, decoder, encoder, multiplexers, Sequential circuits: latches, flip-flops, registers, memories, up- down counters.
<b>UNIT III</b>	<b>PROCESSOR FUNDAMENTALS</b> - 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.
<b>UNIT IV</b>	<b>MEMORY</b> - Physical memory , addressing, virtual memory, address translation, paging, cache, L1,L2,L3 cache memories, cache mapping, LRU replacement.
<b>UNIT V</b>	<b>I/O DATA TRANSFER</b> - Data transfer, Serial and Parallel 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. Morris mano, "Digital design" PHI/Pearson, fourth edition 2006.
2. "Essentials of Computer Architecture", Douglas E.Comer Pearson sixth edition 2012.

3. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, "Computer Organization", Tata McGraw Hill, Fifth Edition, 2002.
4. William Stallings, "Computer Organization and Architecture – Designing for Performance", Pearson Education, Seventh Edition, 2006.
5. David A Patterson and John L. Hennessy, "Computer Organization and Design, The Hardware/Software Interface", Morgan Kaufmann / Elsevier, Third Edition, 2005.

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.

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

DMC5201

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 Packet Switching – Routing – Internetworking concepts – IP – Unreliable

networks – 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, 3<sup>rd</sup> Edition, 2001.

C	M
2	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-processorscheduling – Real time scheduling – Algorithm Evaluation – Process Scheduling Models – Thecritical-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 – ProcessManagement – Scheduling – Memory management – File systems – Input and Output – Inter-process Communication – Network Structure – Security – Windows 7 - History – DesignPrinciples – 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.



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 - DataTypes, 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 - BinaryTrees; **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 - TheSingle-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 SimpleSorting 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-ConquerAlgorithms - 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.

C	M
2	100

**COURSE OBJECTIVES**

- Understand the basic concepts of graphics designs.
- To familiarize the student with the transformation and projection techniques.
- 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.
- 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 – Database systems – Synchronization Issues – Presentation requirements – Applications – Video conferencing – Virtual reality – Interactive video – video on demand.

**REFERENCE BOOKS:**

1. Hearn D and Baker M.P, “Computer graphics – C Version”, 2<sup>nd</sup> Edition, Pearson Education, 2004.
2. Ralf Steinmetz, Klara Steinmetz, “Multimedia Computing, Communications and Applications”, Pearson Education, 2004.
3. Simon J. Gibbs and Dionysios C. Tsichritzis, “Multimedia Programming”, Addison Wesley, 1995.
4. John Villamil, Casanova and Leony Fernandez, Eliar, “Multimedia Graphics”, PHI, 1998.

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 – ProgrammingElements – 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 PointerOperators – 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 – UnaryOperator 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 – VirtualFunctions – 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.
4. Bhave , " Object Oriented Programming With C++", Pearson Education , 2004.

C	M
2	100

**COURSE OBJECTIVES**

- Develop skills in design and implementation of data structures and their applications.
- Learn and implement linear, non linear and tree data structures using C++.
- Learn Set ADT and Graph data structures and its applications using C++.
- Study, implement and analyze of different sorting techniques using C++.

**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

C	M
2	100

**COURSE OBJECTIVES**

- To learn and understand the CPU scheduling algorithms.
- To learn and understand the implementation of memory management algorithm.
- To know the file management techniques practically.

**COURSE OUTCOMES**

- To implement the CPU scheduling methods.
- To implement the inter process communication techniques.
- To implement the page replacement algorithm.

**EXPERIMENTS IN THE FOLLOWING TOPICS:**

- Implement the following CPU Scheduling Algorithms.  
i) FCFS ii) Round Robin iii) Shortest Job First.
- Implement the Mutual Exclusion Problem Using Dekker's Algorithm.
- Implement Inter Process Communication Problem (Producer-Consumer/ Reader- Writer Problem ) Using Semaphores.
- Implement Best fit, First Fit Algorithm for Memory Management.
- Implement Memory Allocation with Pages.
- Implement FIFO page Replacement Algorithm.
- Implement LRU page Replacement Algorithm.
- Implement the creation of Shared memory Segment.
- Implement File Locking.
- Implement Banker's algorithm.

## SEMESTER – III

DMC5301

WEB PROGRAMMING

C	M
2	100

### COURSE OBJECTIVES

- Understand the basics of HTML.
- Learn the concepts of XML related technologies.
- Learn the fundamentals of java.
- 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 servlet 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.

C	M
2	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 – Usecasedriven 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 – Axioms – Colollaries – 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", 2<sup>nd</sup> 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.



C	M
4	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 DataWarehouses - 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 – KnowledgeDiscovery 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 RuleMining - 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 forClassification 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 ofMajor 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.

C	M
2	100

## COURSE OBJECTIVES

- Understand the concepts and models of security in computing. Understand the cryptographic techniques used.
- Explain the security standards followed at the network level and at the application level.
- Estimate the level of security risk faced by an organization and the counter measures to handle the risk.
- Learn secured software development.

## COURSE OUTCOMES

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

- Compare various Cryptographic Techniques.
- Design secure applications.
- Inject secure coding in the developed applications.

**UNIT I        SECURITY – OVERVIEW** - The Threat Environment – attackers and attacks – Security Planning and Policy – risk analysis – governance frameworks.

**UNIT II        CRYPTOGRAPHY**- Elements of cryptography – ciphers – encryption systems – symmetric /asymmetric - DES, AES, RSA – key management – authentication – cryptographic systems - standards – secure networks VPNs, SSL/TLS, IPSec, LAN security.

**UNIT III        ACCESS CONTROL** - Physical access control – access cards – authentication mechanisms –directory servers – Firewalls – packet filtering – stateful packet inspection – NAT – IDS – Firewall architectures.

**UNIT IV        HOST AND DATA SECURITY**- Host Hardening – OS hardening – managing vulnerabilities,permissions - data protection – Application security – issues – e-commerce security – e-mail security - Incident and Disaster Response.

**UNIT V        SECURE CODING** - OWASP/SANS Top Vulnerabilities - Buffer Overflows - Incompletemediation - XSS - Anti Cross Site Scripting Libraries anonical Data Format - Command Injection - Redirection - Inference – Application Controls - - C Secured Software Development Life Cycle - Testing, Maintenance and Operation - Evaluation of Security Systems.

## REFERENCE BOOKS:

1. Raymond R. Panko, "Corporate computer and network security", Second edition, Pearson, 2012.
2. Wade Trappe, Lawrence C Washington, "Introduction to Cryptography with Coding and Theory", Second Edition, Pearson, 2007.
3. Matt Bishop, "Computer Security: Art and Science", Pearson, 2003.
4. Charles Pfleeger, Shari Lawrence Pfleeger, Devin N Paul, "Security in Coding", Pearson, 2007.
5. Wenbo Mao, "Modern Cryptography Theory and Practice", Pearson, 2004.

<b>C</b>	<b>M</b>
<b>2</b>	<b>100</b>

**COURSE OBJECTIVES**

- Understand the application number theory in security.
- Study the symmetric key and public key algorithms.
- Understand the compression techniques for security.

**COURSE OUTCOMES**

- Able to implement program using modules arithmetic for security.
- To implement symmetric key and public key algorithm.
- Ability to implement algorithms for digital signature and hashing.

**EXPERIMENTS IN THE FOLLOWING TOPICS:**

1. Write programs to implement the following number theory concept Prime and Relatively Prime Numbers  
Arithmetic Modulo 8 and Multiplication  
Modulo 8 Fermat's Theorem and Euler's  
Totient Function
2. Write programs to implement the following cryptography algorithms Playfair cipher and Hill cipher  
Simplified DES  
algorithm RSA  
algorithm
3. Write programs to implement the following  
hash a MD5  
SHA-1
4. Write programs to implement the following  
Authentication Digital Signature and Digital Certificate  
Kerberos  
System X.509
5. Write a program to implement Hacking  
windows. BIOS Passwords.  
Windows login password  
Internet explorer users  
Changing windows  
visuals Accessing  
restricted drives.

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.

## SEMESTER - IV

DMC5401

### UNIX AND NETWORK PROGRAMMING

C	M
4	100

#### COURSE OBJECTIVES

- To understand the design of the Unix operating system using system calls.
- Familiarize with the various inter process communication.
- To learn the various low-level algorithms for socket programming used in UNIX.

#### COURSE OUTCOMES

- Ability to develop the application using system calls.
- Able to write programs for UNIX networking.
- Able to write programs for communication applications.

- UNIT I      INTRODUCTION & FILE SYSTEM** - Overview of UNIX OS - File I/O – File Descriptors – File Sharing - Files and directories – File types - File access permissions – File systems – Symboliclinks - Standard I/O library – Streams and file objects – Buffering - System data files and information - Password file – Group file – Login accounting – system identification.
- UNIT II      PROCESSES** - Environment of a UNIX process – Process termination – command line arguments - Process control – Process identifiers - Process relationships terminal logins – Signals - threads.
- UNIT III      INTERPROCESS COMMUNICATION** - Introduction - Message passing (SVR4)- pipes – FIFO– message queues - Synchronization (SVR4) – Mutexes – condition variables – read – write locks – file locking – record locking – semaphores – Shared memory(SVR4).
- UNIT IV      SOCKETS** - Introduction – transport layer – socket introduction - TCP sockets – UDP sockets -raw sockets – Socket options - I/O multiplexing - Name and address conversions.
- UNIT V      APPLICATIONS** - Debugging techniques - TCP echo client server - UDP echo client server -Ping - Trace route - Client server applications like file transfer and chat.

#### REFERENCE BOOKS:

1. W.Richard Stevens, Stephen A.Rago “Advanced programming in the UNIX environment”, Pearson education, 2<sup>nd</sup> Edition 2005.
2. W.Stevens, Bill Fenner, Andrew Rudoff, “Unix Network Programming”, Volume 1, The Sockets Networking API, 3<sup>rd</sup> Edition, Pearson education, Nov 2003.
3. Meeta Gandhi, Tilak Shetty and Rajiv Shah “The ‘C’ Odyssey Unix –The open Boundless C”, 1<sup>st</sup> Edition ,BPB Publications 1992.

<b>C</b>	<b>M</b>
<b>2</b>	<b>100</b>

**COURSE OBJECTIVES**

- To understand the J2EE, J2SE and J2ME concepts.
- Familiarize with java networking and RMI.
- To learn about CORBA and web services.

**COURSE OUTCOMES**

- To develop the application with JAVA networking and RMI features.
- To implement the database connectivity.
- Creation of web services.

**UNIT I**      **BASIC CONCEPTS** - Distributed and Enterprise Systems- Variants on Java platform (J2EE,J2SE, J2ME)-Enterprise Systems Architecture-J2EE model architectures

**UNIT II**      **JAVA NETWORKING AND RMI** - Input / output Streams – Java Networking – UDP and TCPSockets - Java Remote Method Invocation – STUB and Skeleton

**UNIT III**      **CORBA AND JDBC** - Introduction to CORBA – CORBA Environment – Database Connectivity– JDBC – JDBC with Servlets and JSP.

**UNIT IV**      **ENTERPRISE JAVA BEANS** - Entity Beans-Session Beans-Message Driven Beans –Applications.

**UNIT V**      **SOA AND WEB SERVICES** - SOA Fundamentals – XML – Comparison XML and HTML – WebServices – SOAP Protocols – UDDI.

**REFERENCE BOOKS:**

1. Herbert Schildt, "Java The Complete Reference", Eighth Edition, McGraw Hill Professional, 2011.
2. Jayson Falkner and Kevin Jones , "Servlets and JavaServer Pages: The J2EE Technology Web Tier", Addison-Wesley, 2006.

C	M
2	100

### COURSE OBJECTIVES

- To understand the control structures in C#.
- To become familiar with event handling.
- To study the .NET features.

### COURSE OUTCOMES

- Able to write the programs in C#.
- Write programs for database access.
- Develop web applications using .NET.

- UNIT I**      **BASICS OF C#** - C# and the .NET framework – C# basics – Objects and types – Inheritance – Arrays – Operators and casts – Indexers.
- UNIT II**      **DELEGATES, COLLECTIONS & EXCEPTIONS** - Delegates and events – Strings and regular expressions – Generics – Collections – Memory management and pointers – Errors and exceptions.
- UNIT III**      **MULTITHREADING & NETWORKING** - Tracing and events - threading and synchronization - .Net security – localization – Manipulating XML - Managing the file system – basic network programming.
- UNIT IV**      **APPLICATION DEVELOPMENT** - Window based applications – Data access with .NET – basics of ASP .NET - Introduction to web services.
- UNIT V**      **.NET ASSEMBLIES** - Architecture – Assemblies – shared assemblies – CLR hosting – Appdomains – Reflection.

### REFERENCE BOOKS:

1. Christian Nagel et al. "Professional C# 2005 with .NET 3.0", Wiley India, 2007.
2. Jesse Liberty, "Programming C#", O'Reilly, 2001.
3. Andrew Troelson, "Pro C# with .NET 3.0", Apress, 2007.
4. Kevin Hoffman, "Visual C# 2005", Pearson Education, 2006.
5. S. Thamarai Selvi, R. Murugesan, "A Text Book on C#", Pearson Education, 2003.

C	M
2	100

**COURSE OBJECTIVES**

- To learn about HTML and CSS programming.
- To understand XML schema and programs for client side validation using Javascript.
- Familiarize with server side programs.

**COURSE OUTCOMES**

- Ability to develop client/server application.
- Create web services using HTML and XML programming.
- Ability to write programs for database connectivity.

**EXPERIMENTS IN THE FOLLOWING TOPICS:**

- Web programming with HTML tags, CSS for styling, Page layout
- Develop web pages using JavaScript for client side programming and HTML forms Using The DOM and the JavaScript object models
- Website optimization crunching HTML, using CSS to replace HTML and light-weight graphics to speed up websites
- Creating XML file with XML DTD and XML schema, SAX, XSL
- Constructing dynamic server-side web pages using JSF and integrate the Web application with many of the other Java2 Enterprise Edition application server methodologies such as Enterprise Java Beans, JavaMail, and SOAP.
- [Developing Java Enterprise Applications Using EJB3](#) Session beans, entity beans and message-drivenbeans.
- Working with JNDI, JDBC, JMS.
- Application development using J2ME.



C	M
2	100

**COURSE OBJECTIVES**

- Understand the control structures in C#.
- To become familiar with event handling and XML scripts.
- Learn database connectivity and web services in .NET.

**COURSE OUTCOMES**

- Ability to write programs for client/server application
- Develop the web applications using .NET.
- Ability to create web services using .NET.

**EXPERIMENTS IN THE FOLLOWING TOPICS:**

- Control Structures in C#.
- Arrays and pointers.
- XML Scripts.
- Asp/ VB .NET programs for client/server applications.
- Database Connectivity.
- Web services application.

## SEMESTER - V

DMC5501

**WEB SERVICES**

<b>C</b>	<b>M</b>
<b>2</b>	<b>100</b>

### **COURSE OBJECTIVES**

- Learn the basics of XML technology.
- Understand the background of distributed information system.
- Learn the security features of web services and service composition.

### **COURSE OUTCOMES**

The student should be able to

- Create, validate, parse, and transform XML documents.
- Design a middleware solution based application.
- Develop web services using different technologies.
- Compose set of complex web services.

**UNIT I          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 II          WEB SERVICES BUILDING BLOCK** - Web Services – Definition – Web Services and EAI –Web Services Technologies – XML basics - web services Architecture – SOAP – WSDL –UDDI –WS – Addressing – WS – Routing – Web service implementation – Java based web services - .NET based web services.

**UNIT III          WEB SERVICE SECURITY** - XML signature – XML Encryption – SAML - XKMS – WS-Security –WS Policy –Web service security framework – .NET and passport – UDDI and security - web service security in java – mobile web service security.

**UNIT IV          SEMANTIC WEB SERVICES** - Semantic web service – architecture – RDF Data model – RDF schema – OWL – ontology – role of ontology in web services - semantic Web service implementation issues .

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

### **REFERENCES**

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,” Enterprise web services: An Architect’s Guide”, Practice Hall, 2004.
4. Freunk p.coyle,” XML, web Services and the Data Revolution”, Pearson, 2002.

<b>C</b>	<b>M</b>
<b>4</b>	<b>100</b>

## 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, Gopaldaswamy: "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.

<b>C</b>	<b>M</b>
<b>2</b>	<b>100</b>

**COURSE OBJECTIVES**

- To learn the characteristics of mobile applications.
- Understand the intricacies of UI required by mobile applications.
- To study about the design aspects of mobile application.
- To learn development and programming of mobile applications.

**COURSE OUTCOMES**

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

- To design and implement the user interfaces of mobile applications.
- To design the mobile applications that is aware of the resource constraints of the mobile devices.
- To develop advanced mobile applications that accesses the databases and the web.
- To develop useful mobile applications in the current scenario using Google Android and Eclipse simulator.

**UNIT I INTRODUCTION** - Mobile Applications – Characteristics and Benefits – Application Model – Infrastructure and Managing Resources – Mobile Software Engineering – Frameworks and Tools – Mobile devices Profiles.

**UNIT II USER INTERFACE** - Generic UI Development – VUIs and Mobile Applications – Text to Speech techniques – Designing the right UI – Multimodal and Multichannel UI – Gesture based UIs – Screen Elements and Layouts – Voice XML – Java API.

**UNIT III APPLICATION DESIGN** - Memory Management – Design patterns for limited memory – Work flow for Application Development – Techniques for composing Applications – Dynamic Linking – Plug ins and rules of thumb for using DLLs – Concurrency and Resource Management – Look and feel.

**UNIT IV APPLICATION DEVELOPMENT** - Intents and Services – Storing and Retrieving data – Communication via the Web – Notification and Alarms – Graphics and Multimedia – Telephony – Location based services – Packaging and Deployment – Security and Hacking.

**UNIT V TOOLS** - Google Android Platform – Eclipse Simulator – Android Application Architecture – Event based programming – Apple iPhone Platform – UI tool kit interfaces – Event handling and Graphics services – Layer Animation.

**REFERENCE BOOKS:**

1. Zigurd Mednieks, Laird Dornin, G,Blake Meike and Masumi Nakamura “Programming Android”, O’Reilly, 2011.
2. Reto Meier, “Professional Android 2 Application Development”, Wrox Wiley, 2010.
3. Alasdair Allan, “iPhone Programming”, O’Reilly, 2010.
4. Wei-Meng Lee, “Beginning iPhone SDK Programming with Objective-C”, Wrox Wiley, 2010.
5. Poslad, “Ubiquitous Computing: Smart Devices, Environments and Interactions”, Wiley, 2009.

C	M
2	100

**DMC5504**

**COMMUNICATION SKILLS**

**COURSE OBJECTIVE**

To understand how communication works, and to manage the assumptions more effectively Helps students communicate effectively, appropriately and clearly in all situations.

**COURSE OUTCOME**

Students will be able to identify barriers to effective communication and how to overcome them.

- UNIT I            COMMUNICATION IN BUSINESS** - Systems approach- forms - functions and principles of communication - management and communication- communication patterns - barriers to communication - interpersonal perception – SWOT analysis -Johari Window - Transactional Analysis.
- UNIT II            NON-VERBAL AND INTERCULTURAL COMMUNICATION** - Importance of non-verbal communication - personal appearance - facial expressions- movement- posture – gestures - eye contact –voice - beliefs and customs- worldview and attitude.
- UNIT III            ORAL COMMUNICATION** - Listening - types and barriers to listening - speaking - planning and audience awareness - persuasion- goals - motivation and hierarchy of needs - attending and conducting interviews-participating in discussions, debates - and conferences - presentation skills- paralinguistic features -fluency development strategies.
- UNIT IV            BUSINESS CORRESPONDENCE** - Business letter - principles of business writing- memos -e-mails – agendas- minutes- sales letter- enquiries- orders- letters of complaint- claims and adjustments- notice and tenders- circulars- letters of application and resume.
- UNIT V            BUSINESS PROPOSALS AND REPORTS** - Project proposals- characteristics and structure-Project reports – types- characteristics,-structure-Appraisal reports – performance appraisal, product appraisal- Process and mechanics of report writing- visual aids- abstract - executive summary- recommendation writing- definition of terms.

**REFERENCE BOOKS:**

1. Lesikar, Raymond V., John D Pettit, and Mary E Flatly Lesikar's, "Basic Business Communication", Tata McGraw-Hill, 11<sup>th</sup> edition, New Delhi, 2007.
2. Gerson, Sharan J., and Steven M Gerson, "Technical Writing: Process and Product", Pearson Education, New Delhi, 8<sup>th</sup> Edition, 2013.
3. Murphy, Herta, Herbert W Hildebrandt, and Jane P Thomas, "Effective Business Communication", 7<sup>th</sup> ed. Tata McGraw-Hill, New Delhi.
4. Bovee, Courtland and John V Thill, "Business Communication Today", Pearson Education, New Delhi, 11<sup>th</sup> edition, 2012.
5. McGrath, E. H., S. J, "Basic Managerial Skills for All", Prentice-Hall of India, New Delhi, 8<sup>th</sup> ed. 2011.
6. Raman, Meenakhshi, and Prakash Singh, "Business Communication. O U P", New Delhi, 2<sup>nd</sup> Edition, 2012.
7. Stuart Bonne E., Marilyn S Sarow and Laurence Stuart, "Integrated Business Communication in a Global Market Place", 3<sup>rd</sup> ed. John Wiley India, New Delhi, 2007.
8. Guffey, Mary Ellen., "Business Communication: Process and Product", Thomson and South-western, 7<sup>th</sup> edition, 2010.

C	M
2	100

**COURSE OBJECTIVES**

- Analyze and design a web service based application.
- Learn the security features of web services and service composition.

**COURSE OUTCOMES**

- Ability to write programs to Create, validate, parse, and transform XML documents.
- To develop a middleware solution based application.
- To develop web services using different technologies.

**EXPERIMENTS IN THE FOLLOWING TOPICS:**

- Create an XML file for any domain with multiple sublevel complexity.( Example: Students data, Employee information, Product details etc..).
- Create a DTD and XML schema for the XML file.
- Tabulate the xml content using XSL.
- Validate a XML file using java script with XMLDOM.
- Write a java program to parse an XML file using DOM.
- Write a java program to parse an XML file using SAX.
- Write a program to implement XML – RPC.
- Write a program to implement a web service using java and .NET.

C	M
2	100

**COURSE OBJECTIVES**

- To know about various platforms and tools available for developing mobile applications.
- To realize the differences between developing conventional applications and mobile applications.
- To learn programming skills in J2ME and Android SDK.
- To study about micro browser based applications to access the Internet using Sun Java Toolkit.

**COURSE OUTCOMES**

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

- Develop useful mobile applications for the current scenario in mobile computing and pervasive computing.

**EXPERIMENTS IN THE FOLLOWING TOPICS:**

- Survey of Mobile Application Development Tools.
- Form design for mobile applications.
- Applications using controls.
- Graphical and Multimedia applications.
- Data retrieval applications.
- Networking applications.
- Gaming applications
  - o (Perform the experiments from 2 to 7 in J2ME and Android SDK framework)
- Micro browser based applications using WAP, WML and WML scripts
  - o (Perform experiments in 8 using Sun Java Wireless toolkit)

**SEMESTER -  
VI**

---

**DMC5601**

**CLOUD SERVICES**

<b>C</b>	<b>M</b>
<b>2</b>	<b>100</b>

**COURSE OBJECTIVES**

- Understand the concept of cloud and utility computing.
- To understand the various issues in cloud computing.
- Familiarise themselves with the lead players in cloud.
- To appreciate the emergence of cloud as the next generation computing paradigm.

**COURSE OUTCOMES**

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

- Articulate the main concepts, key technologies, strengths and limitations of cloud computing.
- Identify the architecture, infrastructure and delivery models of cloud computing.
- Explain the core issues of cloud computing such as security, privacy and interoperability.
- Choose the appropriate technologies, algorithms and approaches for the related issues.
- To be able to set up a private cloud.

**UNIT I            INTRODUCTION** - Evolution of Cloud Computing –System Models for Distributed and CloudComputing – 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 ofVirtualization - 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 – LayeredCloud 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”, John W. Rittinghouse and James F.Ransome : CRC Press 2010.
3. “Cloud Computing, A Practical Approach” by Toby Velté, Anthony Velté, Robert Elsenpeter: TMH, 2009.



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, 2009.
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, 2010.

**LIST OF  
ELECTIVES**

---

**DMC5001**

**ADVANCED DATABASES**

<b>C</b>	<b>M</b>
<b>2</b>	<b>100</b>

**COURSE OBJECTIVES**

- Learn the modeling and design of databases.
- Acquire knowledge on parallel and distributed databases and its applications.
- Study the usage and applications of Object Oriented and Intelligent databases.
- Understand the usage of advanced data models.
- To learn emerging databases such as XML, Cloud and Big Data.
- Acquire inquisitive attitude towards research topics in databases.

**COURSE OUTCOMES**

- Develop in-depth understanding of relational databases and skills to optimize database performance in practice.
- Understand and critique on each type of databases.
- Design faster algorithms in solving practical database problems.
- Implement intelligent databases and various data models.

**UNIT I            RELATIONAL DATABASES** - Relational Model - Querying - Storage Structures - QueryProcessing - Normalization.

**UNIT II            OBJECT ORIENTED DATABASES** - Introduction to Object Oriented Data Bases - Approaches- Modeling and Design - Persistence - Transaction - Concurrency - Recovery - Database Administration.

**UNIT III           EMERGING SYSTEMS** - Enhanced Data Models - Client/Server Model - Data Warehousing and Data Mining - Web Databases – Mobile Databases.

**UNIT IV            CURRENT ISSUES** - Rules - Knowledge Bases - Active and Deductive Databases - Distributed Databases and Parallel databases.

**UNIT V            DATABASE DESIGN ISSUES** - Security - Integrity - Consistency - Database Tuning - Optimization and Research Issues.

**REFERENCE BOOKS:**

1. R. Elmasri and S.B. Navathe, "Fundamentals of Database Systems", Addison Wesley, 2011.
2. Gary W. Hanson and James V. Hanson, "Database Management and Design", Prentice Hall of India Pvt Ltd, 1999.
3. Alex Benson, Stephen Smith and Kurt Thearling, "Building Data Mining Applications for CRM", Tata McGraw-Hill, 2000.

C	M
2	100

**COURSE OBJECTIVES**

- Understand the interaction between TCP/IP suite and OS.
- To study about the complicated data structures that are used to implement the various protocols.
- Learn about the routing methodologies within AS and across AS.
- Study about the timer management of TCP in detail.
- To learn the implementation of ICMP and IGMP.

**COURSE OUTCOMES**

At the end of the course the student should be able

- Design a sample protocol stack.
- To come up with more efficient data structures for the protocols.
- To embed the protocol suite in a better and secure way in the OS.
- To come up with the variants of TCP according to the applications.
- To modify IP according to the applications.

- UNIT I            INTRODUCTION** - Standards – Internet – History- OSI model – Protocol suite – Addressing –Transmission media – Local Area and Wide Area Networks – Switching – Connecting devices – IP addressing.
- UNIT II            INTERNET PROTOCOL** - Subnetting – Supernetting – IP packets – Delivery – Routing – Routing model – Routing table – Datagram – Fragmentation – Checksum – IP Design – ARP – RARP – Internet control message protocol – Internet group management protocol.
- UNIT III            TRANSMISSION CONTROL PROTOCOL** - User Datagram protocol – UDP operation – Use –UDP design – TCP services – Flow control – Error control – TCP operation and design – connection – Transition diagram – Congestion control.
- UNIT IV            APPLICATION LAYER AND CLIENT SERVER MODEL** - Concurrency – BOOTP – DHCP –Domain name system – Name space – Distribution – Resolution – Messages – Telnet – Rlogin – Network Virtual Terminal – Character Set – Controlling the server – Remote login.
- UNIT V            APPLICATION PROTOCOLS** - File Transfer Protocol – Connections – Communication – Simple Mail Transfer Protocol – Simple Network Management Protocol – Hyper Text Transfer Protocol – Transaction – Request and Response messages.

**REFERENCE BOOKS**

1. Behrouz A. Forouzan, "TCP/IP Protocol Suite", Tata McGraw Hill Edition 2000.
2. Douglas E. Comer, David L. Stevens, "Internetworking with TCP/IP – Volume I, II and III", Prentice-Hall of India Pvt. Ltd., 5th Edition 2006.

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.

<b>UNIT I</b>	<b>INTRODUCTION</b> - 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.
<b>UNIT II</b>	<b>SOFTWARE TESTING METHODOLOGY</b> - 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.
<b>UNIT III</b>	<b>SOFTWARE TESTING TECHNIQUES</b> - Configuration testing – Compatibility testing – foreign language testing – usability testing – testing the documentation - testing for software security – website testing.
<b>UNIT IV</b>	<b>AUTOMATED TESTING AND TEST TOOLS</b> - Benefits of automation and tools – viewers and monitors – drivers – stubs – stress and load tools – analysis tools- software test automation – random testing – beta testing.
<b>UNIT V</b>	<b>TEST DOCUMENTATION</b> - 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, Gopaldaswamy Ramesh, “Software testing- Principles and Practices”, Pearson education, 2009.

C	M
2	100

## COURSE OBJECTIVES

- Understand the distributed system architectures.
- Know distributed system resource management.
- Understand the various fault tolerant techniques.

## COURSE OUTCOMES

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

- Develop fault tolerant distributed applications.
- Compare various distributed operating system characteristics.
- Apply efficient Resource allocation methodologies in distributed applications.

**UNIT I COMMUNICATION IN DISTRIBUTED ENVIRONMENT** - Introduction – Various Paradigms in Distributed Applications – Remote Procedure Call – Remote Object Invocation – Message-Oriented Communication – Unicasting, Multicasting and Broadcasting – Group Communication.

**UNIT II DISTRIBUTED OPERATING SYSTEMS** - Issues in Distributed Operating System – Threads in Distributed Systems – Clock Synchronization – Causal Ordering – Global States – Election Algorithms – Distributed Mutual Exclusion – Distributed Transactions – Distributed Deadlock – Agreement Protocols.

**UNIT III DISTRIBUTED RESOURCE MANAGEMENT** - Distributed Shared Memory – Data-Centric Consistency Models – Client-Centric Consistency Models – Ivy – Munin – Distributed Scheduling – Distributed File Systems – Sun NFS.

**UNIT IV FAULT TOLERANCE AND CONSENSUS** - Introduction to Fault Tolerance – Distributed Commit Protocols – Byzantine Fault Tolerance – Impossibilities in Fault Tolerance.

**UNIT V CASE STUDIES** - Distributed Object-Based System – CORBA – COM+ – Distributed Coordination-Based System – JINI.

## REFERENCE BOOKS:

1. George Coulouris, Jean Dollimore, Tim Kindberg, "Distributed Systems Concepts and Design", Third Edition, Pearson Education Asia, 2002.
2. Hagit Attiya and Jennifer Welch, "Distributed Computing: Fundamentals, Simulations and Advanced Topics", Wiley, 2004.
3. Mukesh Singhal, "Advanced Concepts In Operating Systems", McGrawHill Series in Computer Science, 2001.
4. A.S.Tanenbaum, M.Van Steen, "Distributed Systems", Pearson Education, 2004.
5. M.L.Liu, "Distributed Computing Principles and Applications", Pearson Addison Wesley, 2004.

C	M
2	100

**COURSE OBJECTIVES**

- Provide a strong foundation of fundamental concepts in Artificial Intelligence.
- To enable Problem-solving through various searching techniques.
- To enable the student to apply these techniques in applications which involve perception, reasoning and learning.
- To apply AI techniques primarily for machine learning, vision, and robotics.

**COURSE OUTCOMES**

- Provides a basic exposition to the goals and methods of Artificial Intelligence.
- Study of the design of intelligent computational agents.
- The knowledge acquired through learning can be used both for problem solving and for reasoning.
- Improves problem solving, reasoning, planning, natural language understanding, computer vision, automatic programming and machine learning.

- UNIT I INTRODUCTION** - Artificial Intelligence Definition – Importance of Artificial Intelligence – Knowledge based Systems – Knowledge Representation – State space search – Production systems – Artificial Intelligence Programming Language – PROLOG – Heuristic search - Depth First - Breadth first – Hill climbing – 4<sup>th</sup> algorithms – Game Playing.
- UNIT II KNOWLEDGE REPRESENTATION** - Propositional Logic – Clause form – Predicate logic – Resolution – Inference Rules – Unification – Semantic networks – frames – conceptual dependency – Scripts – Representing Knowledge using rules.
- UNIT III SYMBOLIC REASONING AND UNCERTAINTY** - Non monotonic Reasoning – Truth maintenance systems – closed world assumption – modal and temporal Logics – Bayes Theorem - certainty factors – Bayesian networks – Dempster – Shafer Theory – Fuzzy logic.
- UNIT IV NATURAL LANGUAGE PROCESSING AND DISTRIBUTED ARTIFICIAL INTELLIGENCE** - Overview of Linguistics – grammars and Languages – Basic parsing techniques – semantic Analysis and representation structures – Natural language generation – natural language systems – Distributed Reasoning systems – Intelligent agents.
- UNIT V EXPERT SYSTEMS** - Architecture – Non production systems Architectures – Knowledge acquisition and validation – Knowledge system building tools – Types of Learning – General Learning model – Learning by induction – Generalization and specialization – Inductive bias – Explanation based Learning.

**REFERENCE BOOKS:**

1. Dan W. Patterson, "Introduction to Artificial Intelligence and Expert Systems", Prentice Hall of India, Delhi, 2001.
2. Elaine Rich and Kevin Knight, S.B.Nair, "Artificial Intelligence" TMH Pub. Delhi, 2009.
3. George F Luger, "Artificial Intelligence, structures and strategies for complex problem solving", Pearson Education, Delhi, 2001.

C	M
2	100

**COURSE OBJECTIVES**

To provide knowledge about management issues related to staffing, training, performance, compensation, human factors consideration and compliance with human resource requirements.

**COURSE OUTCOMES**

Students will gain knowledge and skills needed for successful human resources professional.

- UNIT I      PERSPECTIVES IN HUMAN RESOURCE MANAGEMENT** - Evolution of human resource management – The importance of the human factor – Challenges – Inclusive growth and affirmative action -Role of human resource manager – Human resource policies – Computer applications in human resource management – Human resource accounting and audit.
- UNIT II      THE CONCEPT OF BEST FIT EMPLOYEE** - Importance of Human Resource Planning – Forecasting human resource requirement –matching supply and demand - Internal and External sources. Recruitment - Selection – induction – Socialization benefits.
- UNIT III     TRAINING AND EXECUTIVE DEVELOPMENT** - Types of training methods –purpose-benefits- resistance. Executive development programmes – Common practices - Benefits – Self development – Knowledge management.
- UNIT IV     SUSTAINING EMPLOYEE INTEREST** - Compensation plan – Reward – Motivation – Application of theories of motivation – Career management – Development of mentor – Protégé relationships.
- UNIT V      PERFORMANCE EVALUATION AND CONTROL PROCESS** - Method of performanceevaluation – Feedback – Industry practices. Promotion, Demotion, Transfer and Separation – Implication of job change. The control process – Importance – Methods – Requirement of effective control systems grievances – Causes – Implications – Redressal methods.

**REFERENCE BOOKS:**

1. Dessler, "Human Resource Management", Pearson Education Limited, 2007.
2. Decenzo and Robbins, "Human Resource Management", Wiley, 8<sup>th</sup> Edition, 2007.
3. Luis R.Gomez-Mejia, David B.Balkin, Robert L Cardy. "Managing Human Resource", PHI Learning, 2012.
4. Bernadin , "Human Resource Management" , Tata Mcgraw Hill ,8<sup>th</sup> edition 2012.
5. Wayne Cascio, "Managing Human Resource", McGraw Hill, 2007.
6. Ivancevich, "Human Resource Management", McGraw Hill 2012.
7. Uday Kumar Haldar, Juthika Sarkar, "Human Resource management", Oxford. 2012.

<b>C</b>	<b>M</b>
<b>2</b>	<b>100</b>

**COURSE OBJECTIVES**

- To understand the hacking techniques of computer forensics.
- To learn about data recovery methods.
- To identify 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 the 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, 2012.
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.



C	M
2	100

**COURSE OBJECTIVES**

- Learn the basics of E-Learning concepts.
- Learn the content development techniques.

**COURSE OUTCOMES**

- Develop e – learning application on their own.
- Ability to develop contents for e-learning.
- To perform course management using tools.

- UNIT I INTRODUCTION** - Introduction – Training and Learning, Understanding e-learning, components and models of e- learning, Advocacy of e-learning – benefits, learning styles, criteria for choosing, - Applications of E-learning.
- UNIT II CONCEPTS and DESIGN** - E-Learning Strategy, the essential elements of e-learning strategy, Quality assuring e-learning, suppliers and resources, virtual learning environments, authoring tools, e-assessment, Learning Design Issues – purpose, general principles, designing live e-learning, designing self managed learning.
- UNIT III APPLICATIONS** - Moodle 2.0 E-Learning Course Development – Features, Architecture, Installation and Configuring Site.
- UNIT IV COURSE MANAGEMENT** - Creating – Categories, Courses, Adding Static Course Material –Links, Pages, Moodle HTML Editor, Media Files, Interacting with Lessons and Assignments – Evaluating Students – Quizzes and Feedback.
- UNIT V ENHANCEMENT** - Adding Social Activities - Chat, Forum, Ratings, Blocks – Types, Activities,Courses, HTML, Online Users – Features for Teachers.

**REFERENCE BOOKS:**

1. “Delivering E-Learning: A complete Strategy for Design, Application and Assessment”, Kenneth Fee, Kogan page, 2009.
2. “Designing Successful e-Learning”, Michael Allen, Pfeiffer Publication, 2007.
3. “Moodle 2.0 E-learning Course Development”, William Rice, PACKT, 2011.
4. “Moodle 2.0 First Look”, Mary Cooch, 2010.

C	M
2	100

## COURSE OBJECTIVES

- Understand JOOMLA and become familiar with Extensions.
- Learn the concept of web applications for group discussion.

## COURSE OUTCOMES

- Able to create web content using JOOMLA.
- Able to develop components, web sites and discussion forum.

**UNIT I INTRODUCTION** - History of Joomla - Content management system – Joomlasphere – Domainnames - Usability – Joomla Installation – Database creation – Uploading- Web installation-Configuration – Steps – Global option – User and Media – Smart search.

**UNIT II CONTENT CREATION** - Defining and managing content in web site using joomla - Workingwith Media Manager - Menus and Menu Items – Types – Parameters – Articles- Categories – Layouts – Integration – Permissions.

**UNIT III EXTENSIONS** - Components – Content- Web links – News feed – Contacts – Search - Polls –Modules – Plugins – Languages – Adding extensions – Popular Extensions.

**UNIT IV TEMPLATES** - Basics of Joomla Templates – Design Styling and CSS – Customizing theDefault Template Beez – Beez color schemes - Adding logo – Create own Joomla template with basic template structure.

**UNIT V PRACTICAL APPLICATIONS** - Basic Planning of Business Sites, Education Sites and GroupSites - E-commerce Web Sites – Joomla for NGOs – NPOs – Groups –Clubs – Organizations – Education - Case Studies – Education Web Site.

## REFERENCE BOOKS:

1. Jennifer Marriott, Elin Waring, "The Official Joomla! ", Pearson Education, Second Edition, 2013.
2. Themas A. Powell, "The Complete Reference – Web Design", Tata McGraw Hill, Third Edition, 2003.
3. Ashley Friedlein, "Web Project Management", Morgan Kaufmann Publishers, 2001.
4. H. M. Deitel, P. J. Deitel, A. B. Goldberg, "Internet and World Wide Web – How to Program", Third Edition, Pearson Education 2004.

C	M
2	100

### COURSE OBJECTIVES

- Understand Mobile Business strategies.
- To understand Mobile marketing tools and techniques.
- To know Mobile technologies.

### COURSE OUTCOMES

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

- Analyze various mobile marketing strategies.
- Market Mobile based Applications.
- Apply various tools in mobile marketing.

**UNIT I INTRODUCTION** - Introduction – Mobile Marketing Campaign, Fortune 500 and MobileMarketing, consumers engagement with mobile, Terminologies.

**UNIT II MOBILE MARKETING** - Businesses Vs mobile marketing, classic mistakes in mobilemarketing, laying foundation for successful mobile marketing campaign, understanding technology behind mobile marketing – Android, iOS, Windows Phone.

**UNIT III MOBILE MARKETING TOOLS** - Strategic thinking about Mobile marketing campaign, MobileMarketing Tools – setting up mobile website for different firms, using SMS, MMS and apps to drive customers to business and other ways to attract customers.

**UNIT IV MOBILE APPLICATIONS** - Location Based Marketing: LBS, NFC, Bluetooth and LBA, 2Dcodes, Tablet, Other Mobile Applications, Business Firms connecting to customers using Mobile – case study, Mobile Marketing for B2B companies, Mobile E-commerce to Drive Revenue.

**UNIT V MOBILE APPLICATION DEVELOPMENT** Mobile Payments, Present and Future Mobile - Technology, Mobile Application Development.

### REFERENCE BOOKS:

1. "Go Mobile: Location Based Marketing, Apps, Mobile Optimized Ad Campaigns, 2D codes and other Mobile Strategies to Grow your Business", Jeanne Hopkins, Jamie Turner, John Wiley&Sons Inc., 2012.
2. "M- Commerce", Paul Skeldon, Crimson Publishing, 2012.
3. "M-Commerce Technologies, Services and Business Models", Norman Sadeh , Wiley 2002.
4. "Mobile Commerce, Opportunities, Applications and Technologies of Wireless Business", Paul Mary, Tom Jell, Cambridge University Press, 2001.

C	M
2	100

**COURSE OBJECTIVES**

To get subsequent understanding of game design and development, which includes the processes, mechanics, issues in game design, game engine development, modeling, techniques, handling situations, and logic. At the end, the student will be in a position to create interactive games.

**COURSE OUTCOMES**

- Ability to design graphics for game programming.
- To implement 3D graphics animation techniques for game programming.

**UNIT I      3D GRAPHICS FOR GAME PROGRAMMING** - Coordinate Systems, Ray Tracing, Modeling in Game Production, Vertex Processing Rasterization, Fragment Processing and Output Merging, Illumination and Shaders, Parametric Curves and Surfaces, Shader Models, Image Texturing, Bump Mapping, Advanced Texturing, Character Animation, Physics-based Simulation.

**UNIT II      GAME DESIGN PRINCIPLES** - Character development, Story Telling, Narration, Game Balancing, Core mechanics, Principles of level design, Genres of Games, Collision Detection, Game Logic, Game AI, Path Finding.

**UNIT III     GAMING ENGINE DESIGN** - Renderers, Software Rendering, Hardware Rendering, and Controller based animation, Spatial Sorting, Level of detail, collision detection, standard objects, and physics.

**UNIT IV     GAMING PLATFORMS AND FRAMEWORKS** - Flash, DirectX, OpenGL, Java, Python, XNA with Visual Studio, Mobile Gaming for the Android, iOS, Game engines - Adventure Game Studio, DXStudio, Unity.

**UNIT V      GAME DEVELOPMENT** - Developing 2D and 3D interactive games using OpenGL, DirectX – Isometric and Tile Based Games, Puzzle games, Single Player games, Multi Player games.

**REFERENCE BOOKS:**

1. David H. Eberly, "3D Game Engine Design, Second Edition: A Practical Approach to Real-Time Computer Graphics" Morgan Kaufmann, 2 Edition, 2006.
2. JungHyun Han, "3D Graphics for Game Programming", Chapman and Hall/CRC, 1st edition, 2011.
3. Mike McShaffry, "Game Coding Complete", Third Edition, Charles River Media, 2009.
4. Jonathan S. Harbour, "Beginning Game Programming", Course Technology PTR, 3 edition, 2009.
5. Ernest Adams and Andrew Rollings, "Fundamentals of Game Design", Prentice Hall 1st edition, 2006.
6. Roger E. Pedersen, "Game Design Foundations", Edition 2, Jones & Bartlett Learning, 2009.
7. Scott Rogers, "Level Up!: The Guide to Great Video Game Design", Wiley, 1<sup>st</sup> edition, 2010.

C	M
2	100

## COURSE OBJECTIVES

- Gain the knowledge of FREE / OPEN SOURCE SOFTWARE.
- Learn the concepts of TCP/IP networking and routing, server set up and configuration.
- Summarize the programming tools and basics of X windows server architecture.

## COURSE OUTCOMES

- Design and configure the system networking.
- Develop GUI applications for network.
- Configure the different categories of servers.

- UNIT I HISTORY AND OVERVIEW OF GNU/LINUX AND FOSS** - Definition of FOSS & GNU, History of GNU/Linux and the Free Software Movement, Advantages of Free Software and GNU/Linux, FOSS usage, trends and potential— global and Indian.
- UNIT II SYSTEM ADMINISTRATION** - GNU/Linux OS installation--detect hardware, configure disk partitions & file systems and install a GNU/Linux distribution ; Basic shell commands - logging in, listing files, editing files, copying/moving files, viewing file contents, changing file modes and permissions, process management ; User and group management, file ownerships and permissions, PAM authentication ; Introduction to common system configuration files & log files ; Configuring networking, basics of TCP/IP networking and routing, connecting to the Internet (through dialup, DSL, Ethernet, leased line) ; Configuring additional hardware - sound cards, displays & display cards, network cards, modems, USB drives, CD writers ; Understanding the OS boot up process ; Performing every day tasks using gnu/Linux -- accessing the Internet, playing music, editing documents and spreadsheets, sending and receiving email, copy files from disks and over the network, playing games, writing CDs ; X Window system configuration and utilities--configure X windows, detect display devices ; Installing software from source code as well as using binary packages.
- UNIT III SERVER SETUP AND CONFIGURATION** - Setting up email servers--using postfix ( SMTP services), courier ( IMAP & POP3 services), squirrel mail ( web mail services) ; Setting up web servers --using apache ( HTTP services), php (server-side scripting), perl ( CGI support) ; Setting up file services --using samba ( file and authentication services for windows networks), using NFS ( file services for gnu/Linux / Unix networks) ; Setting up proxy services --using squid ( http / ftp / https proxy services) ; Setting up printer services -using CUPS (print spooler), foomatic (printer database) ; Setting up a firewall - Using netfilter and iptables.
- UNIT IV PROGRAMMING TOOLS** - Using the GNU Compiler Collection --GNU compiler tools ; the C preprocessor (cpp), the C compiler (gcc) and the C++ compiler (g++), assembler (gas) ; Understanding build systems --constructing make files and using make, using autoconf and autogen to automatically generate make files tailored for different development environments ; Using source code versioning and management tools --using cvs to manage source code revisions, patch & diff ; Understanding the GNU Libc libraries and linker --linking against object archives (.a libraries) and dynamic shared object libraries (.so libraries), generating statically linked binaries and libraries, generating dynamically linked libraries ; Using the GNU debugging tools --gdb to debug programs, graphical debuggers like ddd, memory debugging / profiling libraries mpatrol and valgrind ; Review of common programming practices and guidelines for GNU/Linux and FOSS ; Introduction to Bash, sed & awk scripting.
- UNIT V APPLICATION PROGRAMMING** - Basics of the X Windows server architecture ; Qt localisation support.

**REFERENCE BOOKS:**

1. N. B. Venkateshwarlu (Ed); "Introduction to Linux: Installation and Programming", B S Publishers; 2005.

2. Matt Welsh, Matthias Kalle Dalheimer, Terry Dawson, and Lar Kaufman, "Running Linux", Fourth Edition, O'Reilly Publishers, 2002.
3. Carla Schroder, "Linux Cookbook", First Edition, O'Reilly Cookbooks Series, 2004.

**DMC5013**

**ENTERPRISE RESOURCE PLANNING**

<b>C</b>	<b>M</b>
<b>2</b>	<b>100</b>

**COURSE OBJECTIVE**

- Become familiarize with ERP process.
- Learn ERP implementation process using information technology.

**COURSE OUTCOMES**

- Design and Develop ERP applications by using features of ERP tools.

**UNIT I           BASICS OF ERP** - ERP essentials – ERP evolution – ERP market – ERP tiers – informationsystems – Presentation tier – application tier – database tier.

**UNIT II           ENTERPRISE SYSTEMS** - Enterprise systems – stand alone mainframe systems – clientserver architecture – service oriented architecture – types of enterprise systems – types of data – SAP overview.

**UNIT III          PROCESS IN ERP** - Basic Procurement process – physical flow – document flow – informationflow – financial impact- role of enterprise systems in the procurement process – fulfillment process – production process.

**UNIT IV          INTEGRATION** - Integrated processes – Integrated processes – additional intracompany processes – extended (intracompany) processes.

**UNIT V          CASE STUDY** - ERP for construction industry – ERP for a corrugated box manufacturingcompany – ERP for lens making company – ERP for furniture manufacturing company – ERP for toys manufacturing company - Mc Donald's story – Automobile enterprises.

**REFERENCE BOOKS:**

1. Simha R Magal, Jeff Word, "Essentials of Business Processes and Information Systems", Wiley Publications, 2009.
2. Marianne Bradford, "Modern ERP: Select, Implement and use Today's advanced business systems", Lulu Publishers, Second Edition, 2010.
3. Jyotindra Zaveri, "Enterprise Resource Planning", Second edition, Himalaya Publishing house, 2012.

<b>C</b>	<b>M</b>
<b>2</b>	<b>100</b>

**COURSE OBJECTIVES**

- Understand the Linear Programming models.
- To understand assignment and transportation problem.
- To understand the concepts of project scheduling.

**COURSE OUTCOMES**

- Able to solve optimization problem.
- Able to design project planning methods.
- To use queuing models for network problems.

<b>UNIT I</b>	<b>LINEAR PROGRAMMING MODELS</b> - Mathematical Formulation - Graphical Solution of linear programming models – Simplex method – Artificial variable Techniques- Variants of Simplex method.
<b>UNIT II</b>	<b>TRANSPORTATION AND ASSIGNMENT MODELS</b> - Mathematical formulation of transportation problem- Methods for finding initial basic feasible solution – optimum solution - degeneracy – Mathematical formulation of assignment models – Hungarian Algorithm – Variants of the Assignment problem.
<b>UNIT III</b>	<b>INTEGER PROGRAMMING MODELS</b> - Formulation – Gomory's IPP method – Gomory's mixed integer method – Branch and bound technique.
<b>UNIT IV</b>	<b>SCHEDULING BY PERT AND CPM</b> - Network Construction – Critical Path Method – Project Evaluation and Review Technique – Resource Analysis in Network Scheduling.
<b>UNIT V</b>	<b>QUEUING MODELS</b> - Characteristics of Queuing Models – Poisson Queues - (M / M / 1) : (FIFO / ∞ / ∞), (M / M / 1) : (FIFO / N / ∞), (M / M / C) : (FIFO / ∞ / ∞), (M / M / C) : (FIFO / N / ∞) models.

**REFERENCE BOOKS:**

1. Taha H.A., "Operations Research : An Introduction " 7<sup>th</sup> Edition, Pearson Education, 2008.
2. A.M.Natarajan, P.Balasubramani, A.Tamilarasi, "Operations Research", Pearson Education, Asia, 2005.
3. Prem Kumar Gupta, D.S. Hira, "Operations Research", S.Chand & Company Ltd, New Delhi, 3<sup>rd</sup> Edition , 2003.



C	M
2	100

### COURSE OBJECTIVES

- Learn the categories of technology commercialization.
- Understand the concepts of technology Negotiation.
- Gain knowledge about pattern filing and Commercialization.

### COURSE OUTCOMES

- Device successful Commercialization process.
- Implementation of Modernization with effective material transfer agreements.
- Register patent and follow up for commercialization.

**UNIT I      COMMERCIALIZATION PROCESS** - Technology as asset - Technology and economical changes - Competitive technology strategic options - Types of commercialization – Commercialization Process.

**UNIT II      TECHNOLOGY LICENSING** - Technology Licensing - Rights of licence holders - Financial terms - documentation - cross licenses - Collaboration and public policy.

**UNIT III      TECHNOLOGY NEGOTIATION** - Technology Negotiation - Preparation and conduct of negotiations - Technology outsourcing - Socio, economic, political, legal and cultural considerations.

**UNIT IV      TECHNOLOGY PATENTING** - Technology patenting - Filing patent applications - Patent classifications - Commercializing patented technology - Arbitration and mediation.

**UNIT V      TECHNOLOGY DIFFUSION** - Technology diffusion - WTO implication on Technology Commercialization – Global trends in technology commercialization.

### REFERENCE BOOKS:

1. "Corporate Venturing", – Zeans Block & Ian c. Macmillan – Harvard Business School Press, 1993.
2. "Innovation Management, Strategies, Implementation and Profit", by Afuah Oxford University Press 2<sup>nd</sup> edition. 2003.
3. "Mastering The Dynamics of Innovation", by UTTERBACK, J. Harvard Business School Press 1996.