



# MIC COLLEGE OF TECHNOLOGY (Autonomous)

## DEPARTMENT OF COMPUTER APPLICATIONS Proposed Course Structure (Autonomous) for MCA. MCA I Semester

S.No	Course Code	Course Category	Course Title	Hours per week			Total Contact Hours	Credits
				Lecture	Tutorial	Practical		
1	PG18MC1T01	ESC	Introduction to Computer and C Programming	3			3	3
2	PG18MC1T02	PCC	Computer Organization	3			3	3
3	PG18MC1T03	PCC	Discrete Mathematical Structures and Graph Theory	3			3	3
4	PG18MC1T04	HSMC	Accountancy & Financial Management	3			3	3
5	PG18MC1T05	BSC	Probability & Statistics	3			3	3
6	PG18MC1L06	HSMC	English Communication Skills Lab			4	2	2
7	PG18MC1L07	ESC	C Programming Lab			4	2	2
8	PG18MC1T08	MC	Environmental Studies	2			2	--
								19

## MCA II Semester

S.No	Course Code	Course Category	Course Title	Hours per week			Total Contact Hours	Credits
				Lecture	Tutorial	Practical		
1	PG18MC2T01	PCC	Data Structures	3			3	3
2	PG18MC2T02	PCC	Operating Systems	3			3	3
3	PG18MC2T03	PCC	Data Base Management Systems	3			3	3
4	PG18MC2T04	HSMC	Management Science	3			3	3
5	PG18MC2T05	PCC	Unix Shell Programming	3			3	3
6	PG18MC2L06	PCC	Data Structures through C Lab			4	2	2
7	PG18MC2L07	PCC	DBMS Lab			4	2	2
8	PG18MC2L08	PCC	Unix Shell Programming Lab			4	2	2
								21



# MIC COLLEGE OF TECHNOLOGY (Autonomous)

## MCA III Semester

S.No	Course Code	Course Category	Course Title	Hours per week			Total Contact Hours	Credits
				Lecture	Tutorial	Practical		
1	PG18MC3T01	PCC	Java Programming	3			3	3
2	PG18MC3T02	ESC	R Programming	3			3	3
3	PG18MC3T03	PCC	Software Engineering	3			3	3
4	PG18MC3T04	PCC	Computer Networks	3			3	3
5	PG18MC3T05	PCC	Design & Analysis of algorithms	3			3	3
6	PG18MC3L06	PCC	Java Programming Lab			4	2	2
7	PG18MC3L07	ESC	R Programming Lab			4	2	2
8	PG18MC3T08	HSMC	Professional Ethics & Human Values	2			2	--
								19

## MCA IV Semester

S.No	Course Code	Course Category	Course Title	Hours per week			Total Contact Hours	Credits
				Lecture	Tutorial	Practical		
1	PG18MC4T01	PCC	Web Technologies	3			3	3
2	PG18MC4T02	PCC	Python Programming	3			3	3
3	PG18MC4T03	PCC	Data warehousing & Data Mining	3			3	3
4	PG18MC4T04		E-Commerce	3			3	3
5	PG18MC4T05/ PG18MC4T06/ PG18MC4T07	PEC	Elective 1: Cryptography & Network Security Distributed Systems Artificial Intelligence	3			3	3
6	PG18MC4L08	PCC	Web Technologies Lab			4	2	2
7	PG18MC4L09	PCC	Python Programming Lab			4	2	2
8	PG18MC4L10	PCC	Object Oriented Analysis & Design Lab			4	2	2
9	PG18MC4T11	HSMC	Employability Skills	3			3	--
								21



# MIC COLLEGE OF TECHNOLOGY (Autonomous)

## MCA V Semester

S.No	Course Code	Course Category	Course Title	Hours per week			Total Contact Hours	Credits
				Lecture	Tutorial	Practical		
1	PG18MC5T01	PCC	Software Testing Methodologies	3			3	3
2	PG18MC5T02	PCC	Hadoop & Big Data	3			3	3
3	PG18MC5T03	PCC	Mobile Application Development	3			3	3
4	PG18MC5T04/ PG18MC5T05/ PG18MC5T06	PEC	Elective 2: Machine Learning Cloud Computing Human Computer Interaction	3			3	3
5	PG18MC5T07/ PG18MC5T08/ PG18MC5T09	PEC	Elective 3: Internet of Things Software Project Management Neural Networks	3			3	3
6	PG18MC5L10	PCC	Hadoop & Big Data Lab			4	2	2
7	PG18MC5L11	PCC	Mobile Application Development Lab			4	2	2
8	PG18MC5T12		Project - I			4	2	2
<b>21</b>								

## MCA VI Semester

S.No	Course Code	Course Category	Course Title	Hours per week			Total Contact Hours	Credits
				Lecture	Tutorial	Practical		
1	PG18MC6T01		Technical Seminar		3		3	3
2	PG18MC6L02		Project - II			32	16	16
<b>19</b>								



Devineni Venkata Ramana & Dr. Hima Sekhar

# MIC COLLEGE OF TECHNOLOGY (Autonomous)

---

# SEMESTER I SYLLABUS



## **Introduction to Computer and C Programming**

**Lecture : 3 Practical : 0**

**Internal Marks:30**

**Credits: 3 Tutorial : 0**

**External Marks:70**

### **Course Outcomes**

The student will learn

- To formulate simple algorithms for arithmetic, logical problems and translate them to programs in c language.
- To implement conditional branching, iteration and recursion.
- To decompose a problem into functions and synthesize a complete program using divide and conquer approach.
- To use arrays, pointers and structures to formulate algorithms and programs.
- To apply programming to solve matrix addition and multiplication problems and searching and sorting problems.
- To use structures and files

### **UNIT – I:**

**Introduction to Computer Problem Solving:** Programs and Algorithms, characteristics of an algorithm, Flowchart, Pseudo-code, The Problem – Solving Aspect, Top-Down design

**Fundamental Algorithms** Introduction, Summation of a set of numbers, Factorial computation, Generation of Fibonacci sequence, reversing the digits of an Integer.

### **UNIT – II:**

**Factoring Methods:** Introduction, GCD of two Integers, Generating Prime numbers, Computing the Prime Factors of an Integer, Generation of pseudo-random numbers

**Array Techniques:** Introduction, Array Order Reversal, Finding the Maximum number in a set, Removal of duplicates from an ordered array, Partitioning an Array

### **UNIT-III:**

**Programming Languages and Introduction to C Programming:** Properties of Machine Language, Assembly Language, High-Level Languages, Procedural and Object-Oriented Languages. Structure of C program, Indentation, Comments, Identifiers & variables, Data Types

Interactive Input, Formatted Output, Format Modifiers, Operators, Operator precedence & Associativity, Relational expressions, Type Casting, Mathematical Library Functions, Selection control statements: if and switch

### **UNIT -IV:**

**Repetition structures:** Basic Loop Structures: while, for , do-while, Nested loops, **Modular Programming:** Functions and parameter declarations, Returning a Value, Functions with Empty Parameter Lists, Variable Scope

**Modular Programming** Variable Storage Class: Local, Global, **Pointers:** declaration and its usage, Functions with parameters: pass by value, pass by address, pointer to a function and function pointer



**UNIT-V:**

**Arrays:** One-Dimensional Arrays, Input and Output of Array Values, Array Initialization, Arrays as Function Arguments, Two-Dimensional Arrays, Larger Dimensional Arrays- Matrices

**Strings:**String Fundamentals, String Input and Output, String Processing, Library Functions related to strings.

**TEXT BOOKS:**

[1] R.G. Dromey, “How to Solve it By Computer”, Prentice-Hall International Series in Computer Science, 1982.

[2] ReemaThareja, “Computer Fundamentals and C Programming”, Oxford, 2012

**REFERENCE BOOKS:**

[1] DENNIS M. RITCHIE, BRIAN W KERNIGHAN, “ The C Programming Language”, Prentice-Hall International Series in Computer Science, Second Edition.

[2] Michael Schneider, Steven W. Weingart, David M. Perlman, “An Introduction to Programming and Problem Solving With Pascal”, John Wiley and Sons Inc ,1984.

[3] David Gries, “The Science of Programming”, Springer Verlag, 1981.



**COMPUTER ORGANIZATION**

**Lecture : 3 Practical : 0**

**Internal Marks:30**

**Credits: 3 Tutorial : 0**

**External Marks:70**

**PREREQUISITES: -DLD**

**COURSE OUTCOMES:**

**Upon successful completion of the course, the student will be able to:**

1. Understand the architecture of a modern computer with its various processing units.
2. Understand RTL, micro operations, instruction cycle
3. Understand the features of hardwired and micro programmed control units.
4. Analyze the memory hierarchy system and performance improvement by cache memory.
5. Analyze the communication methods of I/O devices and standard I/O interfaces.

**SYLLABUS:**

**UNIT I:**

**Basic Structure of Computers:** Computer Types, Functional unit, Basic Operational concepts, Bus structures, Data Representation: Data types, Complements, Fixed Point Representation. Floating – Point Representation. Other Binary Codes, Error Detection codes. Performance, The history of computer development.

**UNIT II:**

**Register Transfer Language And Micro Operations:** Register Transfer language. Register Transfer Bus and memory transfers, Arithmetic Micro operations, logic micro operations, shift micro operations, Arithmetic logic shift unit.

**Basic Computer Organization and Design:** Instruction codes, Computer Register, Computer instructions, Timing and control, Instruction cycle, Memory – Reference Instructions. Input – Output and Interrupt.

**UNIT III :**

**Central Processing Unit:** Stack Organization, Instruction formats, Addressing modes, Data Transfer and Manipulation Instructions, Program control Instructions.

**Control Unit:** Control Memory, Hard wired control, Micro programmed control and Micro Instruction Format, Address Sequencing, Design of Control Unit.

**UNIT IV:**

**Memory Organization:**

Memory Hierarchy, Primary Memory, Introduction to Secondary Memory, Associative Memory, Cache  
Department of Master of Computer Application



Memory, virtual Memory, Memory Management hardware.

#### **UNIT V:**

**PIPELINE AND VECTOR PROCESSING:** Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline Vector Processing, Array Processors.

**Input-Output Organization:** Peripheral Devices, Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupts, Direct memory Access, IOP, Serial Communication.

#### **TEXT BOOKS**

1. M.Morris Mano, —Computer Systems Architecture, Pearson Education publishers, 3rd edition.
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, —Computer Organization, TMH publications, 5th edition, 2002.

#### **REFERENCE BOOKS:**

1. William Stallings, —Computer Organization and Architecture, Pearson/PHI publishers, 6th edition, 2004.
2. Andrew S. Tanenbaum, —Structured Computer Organization, Pearson/PHI publishers, 4th edition, 2005.
3. John D Carpinelli, —Computer Systems Organization and Architecture, Pearson Education, 1st edition, 2001





**Discrete Mathematical Structures and Graph Theory**

**Lecture : 3 Practical : 0**

**Internal Marks:30**

**Credits: 3 Tutorial : 0**

**External Marks:70**

**SYLLABUS:**

**UNIT-I**

**Mathematical Logic:** Introduction, Statements and Notation, Connectives, Normal forms, Theory of inference for Statement Calculus, The Predicate Calculus, Inference theory of Predicate calculus.

**UNIT-II**

**Set Theory:** Introduction, Basic concepts of set theory, Principle of Inclusion and Exclusion, Properties of Binary relations, Relation matrix and Digraph, operations on relations, Partition and covering, Transitive closure, Equivalence, Compatibility and Partial Ordering Relations, Hasse Diagrams, Bijective functions, Inverse functions, Composition of functions, Recursive functions, Pigeonhole principle and its applications.

**UNIT-III**

**Algebraic Structures:** Algebraic systems and examples, general properties, semigroup, monoid, groups and subgroups.

**Number Theory:** Properties of integers, Division algorithm, The greatest common divisor, Euclidean algorithm (without proof), Least common multiple, testing of prime numbers, The fundamental theorem of Arithmetic, Fermat's theorem and Euler's theorem (without proofs) and its applications.

**UNIT-IV**

**Combinatorics and Recurrence Relations:** Basic counting principles- sum rule, solving recurrence relations by substitution and by the method of characteristic roots.

**UNIT -V:**

**Graph Theory:** Basic Concepts of Graphs, Sub graphs, Matrix Representation of Graphs: Adjacency Matrices, Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian and Hamiltonian Graphs, Multigraphs, Planar Graphs, Euler's Formula, Graph Colouring and Covering, Chromatic Number, Spanning Trees, Algorithms for Spanning Trees (Problems Only and Theorems without Proofs).

**TEXT BOOKS:**



1. Discrete Mathematics for Computer Scientists and Mathematicians, J. L. Mott, A. Kandel, T.P. Baker, 2nd Edition, Prentice Hall of India.
2. Discrete Mathematical Structures with Applications to Computer Science, J. P. Tremblay and P. Manohar, Tata McGraw Hill
3. Mathematical Foundation for Computer science, S. Santha, E.V. Prasad, Cengage publications.

..

**REFERENCE BOOKS:**

1. Discrete Mathematics and its Applications with Combinatorics and Graph Theory, K. H. Rosen, 7th Edition, Tata McGraw Hill.
2. Discrete Mathematical Structures, Bernard Kolman, Robert C. Busby, Sharon Cutler Ross, PHI.
3. Discrete Mathematics, S. K. Chakraborty and B.K. Sarkar, Oxford, 2011.



## **Accountancy & Financial Management**

**Lecture : 3 Practical : 0**

**Internal Marks:30**

**Credits: 3 Tutorial : 0**

**External Marks:70**

### **COURSE OBJECTIVES:**

**The main objectives of this course are given below:**

- The Learning objective of this Unit is to understand the concept and nature of Managerial Economics and its relationship with other disciplines, Concept of Demand and Demand forecasting
- The Learning objective of this Unit is to understand the concept of Production function, Input Output relationship, different Cost Concepts and Concept of Cost-Volume-Profit Analysis
- The Learning Objective of this Unit is to understand the Nature of Competition, Characteristics of Pricing in the different market structure and significance of various pricing methods.
- The Learning objective of this Unit is to know the different forms of Business organization and their Merits and Demerits both public & private Enterprises and the concepts of Business Cycles
- The Learning objective of this Unit is to understand the different Accounting Systems preparation of Financial Statements and uses of different tools for performance evaluation
- The Learning objective of this Unit is to understand the concept of Capital, Capitalization, Capital Budgeting and to know the techniques used to evaluate Capital Budgeting proposals by using different methods.

### **COURSE OUTCOMES:**

**At the end of this course the student will able to:**

- The Learner is equipped with the knowledge of estimating the Demand for a product and the relationship between Price and Demand.
- One should understand the Cost Concepts for decision making and to estimate the least cost combination of inputs.
- One has to understand the nature of different markets and Price Output determination under various market conditions.
- One should be equipped with the knowledge of different Business Units
- The Learner is able to prepare Financial Statements and the usage of various Accounting tools for Analysis.
- The Learner is able to evaluate various investment project proposals with the help of capital budgeting techniques for decision making.

### **Syllabus**

#### **UNIT – I:**

#### **Introduction to Managerial Economics and demand Analysis:**

Definition of Managerial Economics and Scope-Managerial Economics and its relation with other subjects- Concepts of Demand-Types-Determinants-Law of Demand its Exception-Elasticity of Demand-Types and Measurement- Demand forecasting and its Methods.



**UNIT – II:**

**Production and Cost Analyses:**

Production function-Isoquants and Isocosts-Law of Variable proportions- Cobb-Douglas Production function- Economics of Sale-Cost Concepts- Opportunity Cost-Fixed vs Variable Costs-Explicit Costs vs Implicit Costs-Out of Pocket Costs vs Imputed Costs-Cost Volume Profit analysis- Determination of Break-Even Point (Simple Problem).

**UNIT – III:**

**Introduction to Markets, Theories of the Firm & Pricing Policies:**

Market Structures: Perfect Competition, Monopoly and Monopolistic and Oligopoly – Features – Price, Output Determination – Managerial Theories of firm: Maris and Williamson's models – Methods of Pricing: Limit Pricing, Market Skimming Pricing, Internet Pricing: Flat Rate Pricing, Usage sensitive, Transaction based pricing, Priority Pricing.

**UNIT – IV:**

**Types of Business Organization and Business Cycles:**

Features and Evaluation of Sole Trader – Partnership – Joint Stock Company – State/Public Enterprises and their forms – Business Cycles – Meaning and Features – Phases of Business Cycle.

**UNIT – V:**

**Introduction to Accounting & Financing Analysis:**

Introduction to Double Entry Systems – Preparation of Financial Statements- Analysis and Interpretation of Financial Statements

**Capital Budgeting:** Meaning of Capital Budgeting-Need for Capital Budgeting- Techniques of Capital Budgeting-Traditional and Modern Methods.

**TEXT BOOKS :**

1. Dr. N. Appa Rao, Dr. P. Vijay Kumar: 'Managerial Economics and Financial Analysis', Cengage Publications, New Delhi – 2011.
2. Dr. A. R. Aryasri – Managerial Economics and Financial Analysis, TMH 2011.
3. Prof. J.V.Prabhakara rao, Prof. P. Venkatarao. 'Managerial Economics and Financial Analysis', Ravindra Publication.

**REFERENCES :**

1. V. Maheswari : Managerial Economics, Sultan Chand.
2. Suma Damodaran : Managerial Economics, Oxford 2011.
3. Dr. B. Kuberudu and Dr. T. V. Ramana : Managerial Economics & Financial Analysis, Himalaya Publishing House 2011.
4. Vanitha Agarwal : Managerial Economics, Pearson Publications 2011.
5. Sanjay Dhameja : Financial Accounting for Managers, Pearson.
6. Maheswari : Financial Accounting, Vikas Publications.
7. S. A. Siddiqui & A. S. Siddiqui : Managerial Economics and Financial Analysis, New Age International Publishers, 2012.



**PROBABILITY AND STATISTICS**

**Lecture : 3 Practical : 0**

**Internal Marks:30**

**Credits: 3 Tutorial : 0**

**External Marks:70**

**SYLLABUS:**

**UNIT I: Discrete Random variables and Distributions:**

Introduction-Random variables- Discrete Random variable-Distribution function- Expectation-Moment Generating function-Moments and properties. Discrete distributions: Binomial and Poisson distributions.

**UNIT II: Continuous Random variable and distributions:**

Introduction-Continuous Random variable-Distribution function- Expectation-Moment Generating function- Moments and properties. Continuous distribution: Normal distributions, Normal approximation to Binomial distribution.

**UNIT III: Sampling Theory:**

Introduction - Population and samples- Sampling distribution of means (s known)-Central limit theorem- t-distribution- Sampling distribution of means (s unknown)- Sampling distribution of variances  $\chi^2$  and F-distributions- Point estimation- Maximum error of estimate - Interval estimation.

**UNIT IV: Tests of Hypothesis:**

Introduction –Hypothesis-Null and Alternative Hypothesis- Type I and Type II errors –Level of significance - One tail and two-tail tests- Tests concerning one mean and proportion, two means- Proportions and their differences- ANOVA for one-way and two-way classified data.

**UNIT V: Curve fitting and Correlation:**

Introduction - Fitting a straight line –Second degree curve-exponential curve-power curve by method of least squares-Goodness of fit.Correlation and Regression – Properties.

**Text Books:**

1. **Richards A Johnson, Irvin Miller and Johnson E Freund.** Probability and Statistics for Engineering, 9th Edition, PHI.
2. **Jay L.devore,** Probability and Statistics for Engineering and the Sciences, 8<sup>th</sup> edition, Cengage.

**Reference Books:**

1. **ShronL.Myers, Keying Ye, Ronald E Walpole,** Probability and Statistics Engineers and the Scientists, 8th Edition, Pearson 2007.
2. **William Menden Hall, Robert J. Bever and Barbara Bever,** Introduction to probability and statistics, Cengage learning, 2009.



**ENGLISH COMMUNICATION SKILLS LAB**

**Lecture : 0 Practical : 4**

**Internal Marks:40**

**Credits: 2 Tutorial : 0**

**External Marks:60**

**Course Outcomes**

The student will acquire basic proficiency in English including reading and listening comprehension, writing and speaking skills.

List of Experiments:

- 1 Greetings and Introduction**
- 2 Request Permission & Giving Directions**
- 3 Inviting/Complaining/Congratulating**
- 4 Root Words**
- 5 Phonetics-Sounds and Symbols**
- 6 Pronunciation Rules**

**References:**

- 1. *Strengthen Your Steps*, Maruti Publications**
- 2. *Interact*, Orient Blackswan**
- 3. *Word Power Made Easy*, Pocket Books**



## C PROGRAMMING LAB

**Lecture : 0 Practical : 4**

**Internal Marks:40**

**Credits: 2 Tutorial : 0**

**External Marks:60**

List of Experiments:

1.
  - a. Write an algorithm, flowchart and pseudo code to perform all arithmetic operations
  - b. Write an algorithm, flowchart and pseudo code to find average of three numbers
  - c. Write an algorithm, flowchart and pseudo code to find smallest of three numbers
2.
  - a. Write an algorithm, flowchart and pseudo code for finding smallest divisor of an integer.
  - b. Write an algorithm, flowchart and pseudo code to find  $x^y$
3. Write a C program to convert temperature from Fahrenheit to Celsius and vice versa.
4. Write a C program to find the roots of quadratic equation
5. Write a C program to find whether a given number is prime
6. Write a C program find whether a given number is armstrong
7. Write a C program to display reverse of a given number
8. Write a C program to generate first n- terms of a fibonacci sequence.
9. Write a C program to calculate  $\sin(x)$  value, where x is input given by user
10. Write a C program to calculate  $\cos(x)$  value, where x is input given by user
11. Write a C program to perform operations on one dimensional array
  - a. Smallest element of an array
  - b. Largest element of an array
  - c. swap smallest and largest element in an array
12. Write a C program to implement the following
  - a. Addition of two matrices
  - b. Multiplication of two matrices
13. Write a C program to perform the following operations on strings without using string handling functions
  - a. To display length of the string
  - b. To check whether a string is palindrome
  - c. To delete n characters from a given position in a given string
14. Write recursive and non recursive programs for the following
  - a. Factorial of a number
  - b. GCD of two numbers
  - c. Fibonacci series
15. Write a program which illustrates Storage classes



**ENVIRONMENTAL STUDIES**

**Lecture : 2 Practical : 0**

**Internal Marks:-**

**Credits: 0 Tutorial : 0**

**External Marks:-**

**COURSE OUTCOMES:**

1. The importance of environment, Natural resources and current global environmental challenges for the sustenance of the life on planet earth.
2. The concepts of the ecosystem and its function in the environment.
3. The biodiversity of India and the threats to biodiversity, and conservation practices to protect the biodiversity
4. The various attributes of the pollution and their impacts and measures to reduce or control the pollution along with waste management practices.
5. The environmental legislations of India and Social issues and the possible means
6. Environmental assessment and the stages involved in EIA.

**SYLLABUS:**

**UNIT-I: MULTIDISCIPLINARY NATURE OF ENVIRONMENTAL STUDIES**

Introduction- Scope of Environmental Studies- Importance of Environmental Studies- Need for public awareness, Environmental ethics- Contemporary Environmentalists- Environmental Global moves: Stockholm conference, Earth summit

Concept of an ecosystem - Structure of an ecosystem- function of an ecosystem- Food chains, food webs- ecological pyramids- Energy flow in the ecosystem- Ecological succession- Nutrient cycling- 1<sup>o</sup>production & 2<sup>o</sup>production- Major ecosystems: Forest ecosystem- Grassland ecosystem, Desert ecosystem- Aquatic ecosystem: pond, lake ecosystem- Streams, river ecosystem, Oceans

**UNIT-II : NATURAL RESOURCES AND CONSERVATION**

Introduction and classification of natural resources- Forest resources: Use and over-exploitation

- Deforestation-Timber extraction-Mining- Conservation- Water resources: Use and over utilization of surface and ground water,- Floods, drought, Dams and associated problems- Water conservation, rain water harvesting, water shed management- Energy resources: renewable energy sources –solar-wind-hydro-tidal- Ocean thermal-geo thermal-bio mass-bio gas-bio fuels- Hydrogen.- Non-renewable energy sources-coal-petroleum-natural gas-Nuclear energy

**UNIT-III: BIODIVERSITY AND ITS CONSERVATION**

Definition, classification- Value of biodiversity- Threats to biodiversity: habitat loss, man-wildlife





conflicts- Endangered and endemic species of India- Conservation of biodiversity- Biodiversity at national and local levels, Hot-spots of biodiversity

#### **UNIT-IV: ENVIRONMENTAL PROBLEMS**

Global warming, Climate change- Acid rain , Ozone depletion- Air pollution- Water pollution- Soil pollution- Noise pollution, Nuclear hazards- Solid Waste Management: Causes, Consequences and Control methods- Solid Waste Management- Population growth and explosion, effects, control measures- Pollution case studies- Role of an individual in prevention of pollution

#### **UNIT-V: ENVIRONMENTAL LEGISLATION & MANAGEMENT**

Sustainable development- Air (Prevention and Control of Pollution) Act- Drawbacks- Water (Prevention and control of Pollution) Act- Drawbacks- Wildlife Protection Act- Drawbacks- Forest Conservation Act- Drawbacks- Environmental Protection Act- Drawbacks- Environmental Impact Assessment and its significance- Preparation of Environmental Management Plan and Environmental Impact Statement- Ecotourism

#### **TEXT BOOKS:**

1. Environmental Studies, Anubha Kaushik, C P Kaushik, New Age Publications, New Delhi
2. Environmental Studies, K. V. S. G. Murali Krishna, VGS Publishers, Vijayawada
3. Environmental Studies, R. Rajagopalan, 2nd Edition, 2011, Oxford University Press.
4. Environmental Studies, P. N. Palanisamy, P. Manikandan, A. Geetha, and K. Manjula Rani; Pearson Education, Chennai

#### **REFERENCE:**

1. Text Book of Environmental Studies, Deeshita Dave & P. Udaya Bhaskar, Cengage Learning.
2. A Textbook of Environmental Studies, Shaashi Chawla, TMH, New Delhi
3. Environmental Studies, Benny Joseph, Tata McGraw Hill Co, New Delhi
4. Perspectives in Environment Studies, Anubha Kaushik, C P Kaushik, New Delhi



# **II SEMESTER**

  

# **SYLLABUS**



## Data Structures

**Lecture : 3 Practical : 0**

**Internal Marks:30**

**Credits: 3 Tutorial : 0**

**External Marks:70**

**Prerequisites: C PROGRAMMING**

### **SYLLABUS:**

#### **UNIT - I:**

**Sortings:** Bubble sort, Insertion sort, section sort

**Searching:** linear search, binary search

#### **UNIT II:**

**Pointers:** Introduction, Pointer Arithmetic and Arrays, Memory Allocations Functions ,Compatibility, Lvalue and Rvalue, Arrays and Pointers, Passing an Array to a Function, , Array of Pointers. Pointers For Inter Function Communications, Pointers to Pointers

#### **UNIT III:**

**Structures:** Structure Type Declaration, Initialization, Accessing Structures, Operations on Structures, Complex Structures, Structures and Functions, Sending the Whole Structure, Passing Structures through Pointers.

**Unions:** Referencing Unions, Initializers, Unions and Structures, Applications.

**Text Input/output:** Files, Streams, Standard Library Input/Output Functions, Formatting Input/output Functions and Character Input/Output Functions, Command-Line Arguments.

#### **UNIT IV:**

**Stacks:** Definition, Representing stacks, ADT Stack and its operations: Applications of Stacks: Expression Conversion and evaluation – corresponding algorithms. Recursion, Towers of Hanoi problem.

**Queues:** Queue and its Sequential Representation, Queue as an abstract data type, Types of Queue: Simple Queue, Circular Queue, Operations on each types of Queues: Algorithms.

#### **UNIT V:**

**Linked lists:** Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; **Doubly linked list:** operations and algorithms.

**Circular Linked Lists:** all operations their algorithms, Linked representation of Stack and Queue.



Polynomials: Addition, Multiplication.

**TEXT BOOKS:**

- [1] ReemaThareja, “Computer Fundamentals and C Programming”, Oxford, 2012
- [2] Mark Allen Weiss,”Data structure and Algorithm Analysis in C”. Addison Wesley Publication. 2006.
- [3] SEYMOUR LIPSCHUTZ, “Data Structures With C – by Schaum Series” .

**REFERENCE BOOKS:**

- [1] Horowitz Sahni and Anderson-Freed “Fundamentals of Data Structures in C”. 2nd Edition, Universities Press, 2008.
- [2] Richard F. Gilberg & B. A. Forouzan “Data Structures A Pseudocode Approach with C”, Second Edition, CENGAGE Learning.



**OPERATING SYSTEMS**

**Lecture : 3 Practical : 0**

**Internal Marks:30**

**Credits: 3 Tutorial : 0**

**External Marks:70**

**COURSE OUTCOMES:**

**Upon successful completion of the course, the student will be able to:**

1. Understand the functionalities of an operating system and Evaluate different CPU scheduling algorithms.
2. Apply synchronization to cooperating processes and handle the deadlocks
3. Learn various management techniques for efficient utilization of system memory.
4. Understand and analyze theory and implementation of files and Evaluate different disk scheduling algorithms.
5. Analyze the functionalities in various operating systems.

**SYLLABUS:**

**UNIT I**

**Introduction to Operating System Concept:** Types of operating systems, operating systems concepts, operating systems services, Introduction to System call, System call types.

**Process Management** – Process concept, the process, Process State Diagram, Process control block, Process Scheduling- Scheduling Queues, Schedulers, Operations on Processes, Inter process Communication, Scheduling- Basic Concepts, Scheduling Criteria, and Scheduling Algorithms.

**UNIT-II:**

**Concurrency:** Process Synchronization, The Critical-Section Problem, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors, Synchronization examples

**Principles of deadlock** – System Model, Deadlock Characterization, Deadlock Prevention, Detection and Avoidance, Recovery form Deadlock

**UNIT-III:**

**Memory Management:** Swapping, Contiguous Memory Allocation, Paging, structure of the Page Table, Segmentation

**Virtual Memory Management:**

Virtual Memory, Demand Paging, Page-Replacement Algorithms, Thrashing

**UNIT-IV:**

**File system Interface-** the concept of a file, Access Methods, Directory structure, File system mounting, file sharing, protection.



**File System implementation-** File system structure, allocation methods, free-space management

**Mass-storage structure** overview of Mass-storage structure, Disk scheduling, Device drivers,

#### **UNIT V:**

**Linux System:** Components of LINUX, Inter process Communication, Synchronization, Interrupt, Exception and System Call.

**Android Software Platform:** Android Architecture, Operating System Services, Android Runtime Application Development, Application Structure, Application Process management

#### **TEXT BOOKS:**

1. Operating System Concepts, Abraham Silberschatz, Peter Baer Galvin and Greg Gagne 9th Edition, John Wiley and Sons Inc., 2012.
2. Operating Systems – Internals and Design Principles, William Stallings, 7th Edition, Prentice Hall, 2011.
3. Operating Systems-S Halder, Alex A Aravind Pearson Education Second Edition 2016 .

#### **REFERENCES:**

1. Modern Operating Systems, Andrew S. Tanenbaum, Second Edition, Addison Wesley, 2001.
2. Operating Systems: A Design-Oriented Approach, Charles Crowley, Tata Mc Graw Hill Education”, 1996.
3. Operating Systems: A Concept-Based Approach, D M Dhamdhare, Second Edition, Tata Mc Graw-Hill Education, 2007.



**DATA BASE MANAGEMENT SYSTEMS**

**Lecture : 3 Practical : 0**

**Internal Marks:30**

**Credits: 3 Tutorial : 0**

**External Marks:70**

**COURSE OUTCOMES:**

**Upon successful completion of the course, the student will be able to:**

1. Describe a database and different database models
2. Design Entity Relationship models And Relational Model
3. Design and implement queries using Structured Query Language
4. Design database schema using normalization.
5. Understand the characteristics of database transaction management.

**SYLLABUS:**

**Unit – I:**

**Introduction:** Database system, Characteristics (Database Vs File System), Database Users (Actors on Scene, Workers behind the scene), Advantages of Data base systems, and Database applications.

Brief introduction of different Data Models- the ER Model – Relational Model – Other Models; Concepts of Schema, Instance and data independence; Three tier schema architecture for data independence; Database system structure.

**Unit – II:**

**Entity Relationship Model:** Introduction, Representation Of Entities, Attributes, Entity Set, Relationship, Relationship Set, Constraints, Sub Classes, Super Class, Inheritance, Specialization, And Generalization Using ER Diagrams.

**Relational Model:** Introduction to Relational Model, Concepts of Domain, Attribute, Tuple, Relation, Importance Of Null Values, Constraints (Domain, Key Constraints, Integrity Constraints) And Their Importance

**Unit – III:**

**SQL:** Simple Database schema, data types, table definitions (create, alter), different DML operations (insert, delete, update), basic SQL querying (select and project) using where clause, arithmetic & logical operations, SQL functions (Date and Time, Numeric, String conversion), Creating tables with relationship, implementation of key and integrity constraints, nested queries, sub queries, grouping, aggregation, ordering.

Implementation of different types of joins, view(updatable and non-updatable), relational set operations, Definition of NOSQL, History of NOSQL and Different NOSQL products, Applications, features of NoSQL, Difference between SQL and NoSQL

**Unit-IV**

**Schema Refinement (Normalization):** Purpose of Normalization or schema refinement, concept of functional



dependency, normal forms based on functional dependency(1NF, 2NF and 3 NF), concept of surrogate key, Boyce-Codd normal form(BCNF), Lossless join and dependency preserving decomposition, Fourth normal form(4NF).

**Indexing:** Hashing, single and multi-level indexes, dynamic multilevel indexing using B-Tree and B+ tree, index on multiple keys.

### **Unit-V**

**Transaction Management And Concurrency Control:** Transaction, properties of transactions, transaction log, and transaction management with SQL using commit rollback and savepoint, Concurrency control for lost updates, uncommitted data, inconsistent retrievals and the Scheduler. Concurrency control with locking methods: lock granularity, lock types, two-phase locking for ensuring serializability, deadlocks.

**Recovery System:** Introduction to ARIE, The Log, The Write-Ahead Log Protocol, check pointing, Recovery from system crash

### **TEXT BOOKS:**

1. Raghurama Krishnan, Johannes Gehrke, “*Data base Management Systems*”, 3rd Edition, TATA McGrawHill, 2008.
2. Silberschatz, Korth, “*Data base System Concepts*”, 6th Edition, McGraw Hill, 2010.
3. C.J.Date, “*Introduction to Database Systems*”, 7th Edition, Pearson Education, 2002.
4. Professional NOSQL” by Shashank Tiwari, 2011, WROX Press.

### **REFERENCES:**

1. Peter Rob & Carlos Coronel, “*Data base Systems design, Implementation, and Management*”, 7th Edition, Pearson Education, 2000.
2. Elmasri Navrate, “*Fundamentals of Database Systems*”, 5th Edition, Pearson Education, 2007.





**UNIX SHELL PROGRAMMING**

**Lecture : 3 Practical : 0**

**Internal Marks:30**

**Credits: 3 Tutorial : 0**

**External Marks:70**

**OBJECTIVES:**

- Written technical communication and effective use of concepts and terminology.
- Facility with UNIX command syntax and semantics.
- Ability to read and understand specifications, scripts and programs.
- Individual capability in problem solving using the tools presented within the class.
- Students will demonstrate a mastery of the course materials and concepts within in class discussions.

**UNIT-I**

Introduction to unix-Brief History-What is Unix-Unix Components-Using Unix-Commands in Unix-Some Basic Commands-Command Substitution-Giving Multiple Commands.

**UNIT-II**

The File system –The Basics of Files-What’s in a File-Directories and File Names-Permissions-I Nodes-The Directory Hierarchy, File Attributes and Permissions-The File Command knowing the File Type-The Chmod Command Changing File Permissions-The Chown Command Changing the Owner of a File-The Chgrp Command Changing the Group of a File.

**UNIT-III**

Using the Shell-Command Line Structure-Met characters-Creating New Commands-Command Arguments and Parameters-Program Output as Arguments-Shell Variables- -More on I/O Redirection-Looping in Shell Programs.

**UNIT-IV**

Filters-The Grep Family-Other Filters-The Stream Editor Sed-The AWK Pattern Scanning and processing Language-Good Files and Good Filters.

**UNIT-V**

Shell Programming-Shell Variables-The Export Command-The Profile File a Script Run During Starting-The First Shell Script-The read Command-Positional parameters-The \$? Variable knowing the exit Status-More about the Set Command-The Exit Command-Branching Control Structures-Loop Control Structures-The Continue and Break Statement-The Expr Command: Performing Integer Arithmetic-Real Arithmetic in Shell Programs-The here Document(<<)-The Sleep Command-Debugging Scripts-The Script Command-The Eval Command-The Exec Command.

**TEXT BOOKS:**

1. The Unix programming Environment by Brain W. Kernighan & Rob Pike, Pearson.
2. Introduction to Unix Shell Programming by M.G.Venkateshmurthy, Pearson.

**REFERENCE BOOKS:**

1. Unix and shell programmingby B.M. Harwani, OXFORD university press.



**Data Structures Lab**

**Lecture : 0 Practical : 4**

**Internal Marks:40**

**Credits: 2 Tutorial : 0**

**External Marks:60**

**List of Experiments**

1. Write C programs to sort the list of elements using following techniques
  - a. Bubble Sort
  - b. Insertion Sort
  - c. Selection Sort
2. Write C programs to search for an element in an array using following techniques
  - a. Linear Search
  - b. Binary Search
3. Write a C program to demonstrate call by value and call by reference
4. Write a C program to display student information using structures
5. Write a C program to count number of lines, words and characters in a file
6. Write a C program to perform stack operations using arrays
7. Write a C program to perform queue operations using arrays
8. Write C program to implement stack applications.
  - a. Conversion of Infix expression to postfix expression
  - b. Evaluation of postfix expression
  - c. Towers of Hanoi
9. Write a C program to perform circular queue operations using arrays
10. Write a C program to implement following operations on Single Linked List
  - a. Insertion
  - b. Deletion
  - c. Search
11. Write a C program to implement following operations on Double Linked List
  - a. Insertion
  - b. Deletion
  - c. Search
12. Write a C program to implement stack operations using linked list
13. Write a C program to implement queue operations using linked list
14. Write a C program to add two polynomials using linked list
15. Write a C program to multiply two polynomials using linked list



**DATA BASE MANAGEMENT SYSTEMS LAB**

**Lecture : 0 Practical : 4**

**Internal Marks:40**

**Credits: 2 Tutorial : 0**

**External Marks:60**

**COURSE OUTCOMES:**

**Upon successful completion of the course, the student will be able to:**

1. Create own database.
2. Manipulate data in database using SQL language.
3. Experiment with various SQL queries with database created
4. Write programs using PL/SQL language.
5. Create triggers using PL/SQL.

**LIST OF LAB EXPERIMENTS:**

1. Introduction to SQL: DDL, DML, DCL, TCL.
2. Queries for Creating Tables with Constraints, Views.
3. Example SQL Queries using select.
4. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN).
5. Queries using Group By, Order By, and Having Clauses and Working with Index, Sequence, Synonym.
6. Queries to facilitate acquaintance of Built-In Functions, String Functions, Numeric Functions, Date Functions and Conversion Functions.
7. Queries on Joins and Correlated Sub-Queries.
8. Write a PL/SQL Code using Basic Variable, Anchored declarations, and Usage of Assignment Operation.
9. Write a PL/SQL Code Bind and Substitution Variables. Printing in PL/SQL.
10. Write a PL/SQL block using SQL and Control Structures in PL/SQL.
11. Write a PL/SQL Code using Cursors, Exceptions and Triggers.
12. Write a PL/SQL Code using Procedures, Functions, and Packages.

**TEXT BOOKS :**

- 1) ORACLE PL/SQL by example. Benjamin Rosenzweig, Elena Silvestrova, Pearson
- 2) ORACLE DATA BASE LOG PL/SQL Programming SCOTT URMAN, TMH.
- 3) SQL & PL/SQL for Oracle 10g, Black Book, Dr.P.S. Deshpande.
- 4) Data Base Management System, Oracle SQL and PL/SQL, Pranab kumar Das Gupta, P Radha Krishna, PHI.



**UNIX PROGRAMMING LAB**

**Lecture : 0 Practical : 4**

**Internal Marks:40**

**Credits: 2 Tutorial : 0**

**External Marks:60**

1. a) Study of Unix/Linux general purpose utility command list  
man, who, cat, cd, cp, ps, ls, mv, rm, mkdir, rmdir, echo, more, date, time, kill, history, chmod, chown, finger, pwd, cal, logout, shutdown.
- b) Study of vi editor.
- c) Study of Bash shell, Bourne shell and C shell in Unix/Linux operating system.
- d) Study of Unix/Linux file system (tree structure).
- e) Study of .bashrc, /etc/bashrc and Environment variables.
2. Write a C program that makes a copy of a file using standard I/O, and system calls
3. Write a C program to emulate the UNIX `ls -l` command.
4. Write a C program that illustrates how to execute two commands concurrently with a command pipe.  
Ex: - `ls -l | sort`
5. Write a C program that illustrates two processes communicating using shared memory
6. Write a C program to simulate producer and consumer problem using semaphores
7. Write C program to create a thread using pthreads library and let it run its function.
8. Write a C program to illustrate concurrent execution of threads using pthreads library.



Devineni Venkata Ramana & Dr. Hima Sekhar

# MIC COLLEGE OF TECHNOLOGY (Autonomous)

---

# SEMESTER III SYLLABUS



**JAVA PROGRAMMING**

<b>Lecture : 3</b>	<b>Practical : 0</b>	<b>Internal Marks:30</b>
<b>Credits: 3</b>	<b>Tutorial : 0</b>	<b>External Marks:70</b>

**COURSE OUTCOMES:**

1. Understand the principles of object oriented concepts. Define classes and objects by identifying real world entities, their properties and functionalities.
2. Reuse the existing classes by using inheritance and understand the concepts of packages and exception handling.
3. Make use of built-in classes in Java and understand the concept of thread.
4. Develop user interfaces using applets, AWT and Event handling in java.
5. Create portable GUI applications using Swing components.

**UNIT-I:**

Introduction to OOP, procedural programming language vs object oriented language, principles of OOP, applications of OOP, history of java, java features, JVM. Variables, primitive data types, identifiers, literals, operators, expressions, precedence rules and associativity, primitive type conversion and casting, flow of control.

Classes and objects, class declaration, creating objects, methods, constructors and constructor overloading, garbage collector.

**UNIT-II:**

Inheritance, types of inheritance, super keyword, final keyword, overriding and abstract class. Interfaces, creating the packages, using packages, importance of CLASSPATH and java.lang package, importance of static keyword and examples, this keyword, arrays, command line arguments, nested classes.

Exception handling, importance of try, catch, throw, throws and finally block, userdefined exceptions, Assertions.

**UNIT-III:**

Multithreading: Introduction, thread life cycle, creation of threads, thread priorities, thread synchronization, communication between threads. Reading data from files and writing data to files, random access file.

**UNIT-IV:**

Applet class, Applet structure, Applet life cycle, sample Applet programs. Event handling: event delegation model, sources of event, Event Listeners, adapter classes, inner classes.

**UNIT-V:**



AWT: introduction, components and containers, Button, Label, Checkbox, Radio Buttons, List, Boxes, Choice Boxes, Container class, Layouts, Menu and Scrollbar.

Swing: Introduction, JFrame, JApplet, JPanel, Componets in Swings, Layout Managers in Swings, JList and JScrollPane, Split Pane, JTabbedPane, JTree, JTable, Dialog Box.

**TEXT BOOKS:**

1. The complete Reference Java, 8th edition, Herbert Schildt, TMH.
2. Programming in JAVA, Sachin Malhotra, SaurabhChoudary, Oxford.
3. JAVA for Beginners,4e,Joyce Farrell,Ankit R. Bhavsar,Cengage Learning.
4. Object oriented programming with JAVA,Essentials and Applications, Raj Kumar Bhuyya,Selvi,Chu,TMH.
5. Introduction to java programming, 7th edition by Y Daniel Liang, Pearson.

**REFERENCE BOOKS:**

- 1.Java Programming,K.Rajkumar.Pearson
- 2.Core Java,Black Book,R Nageswara rao,Wiley,Dream Tech
- 3.Core Java for Beginners,Rashmi Kanta Das,vikas.
- 4.Object Oriented Programming Through java, P.Radha Krishna, Universities Press



**R PROGRAMMING**

**Lecture : 3**

**Practical : 0**

**Internal Marks:30**

**Credits : 3**

**Tutorial : 0**

**External Marks:70**

**COURSE OBJECTIVES:**

After taking the course, students will be able to

- Use R for statistical programming, computation, graphics, and modeling,
- Write functions and use R in an efficient way,
- Fit some basic types of statistical models
- Use R in their own research,
- Be able to expand their knowledge of R on their own.

**UNIT-I:** Introduction, How to run R, R Sessions and Functions, Basic Math, Variables, Data Types, Vectors, Conclusion, Advanced Data Structures, Data Frames, Lists, Matrices, Arrays, Classes.

**UNIT-II:** R Programming Structures, Control Statements, Loops, - Looping Over Nonvector Sets,- If-Else, Arithmetic and Boolean Operators and values, Default Values for Argument, Return Values, Deciding Whether to explicitly call return- Returning Complex Objects, Functions are Objective, No Pointers in R, Recursion, A Quick sort Implementation Extended Extended Example: A Binary Search Tree.

**UNIT-III:** Doing Math and Simulation in R, Math Function, Extended Example Calculating Probability-Cumulative Sums and Products-Minima and Maxima- Calculus, Sorting, Linear Algebra Operation on Vectors and Matrices, Extended Example: Vector cross Product- Set Operation, Input /output, Accessing the Keyboard and Monitor, Reading and writer Files,

**UNIT-IV:** Graphics, Creating Graphs, The Workhorse of R Base Graphics, the plot () Function – Customizing Graphs, Saving Graphs to Files.

**UNIT-V:** Probability Distributions, Normal Distribution- Binomial Distribution- Poisson Distributions Other Distribution, Basic Statistics, Correlation and Covariance, T-Tests,- ANOVA. Linear Models, Simple Linear Regression, -Multiple Regression.

**OUTCOMES:**

**At the end of this course, students will be able to:**

- List motivation for learning a programming language
- Access online resources for R and import new function packages into the R workspace
- Import, review, manipulate and summarize data-sets in R
- Explore data-sets to create testable hypotheses and identify appropriate statistical tests
- Perform appropriate statistical tests using R Create and edit visualizations with

**TEXT BOOKS:**

- 1) The Art of R Programming, A K Verma, Cengage Learning.
- 2) R for Everyone, Lander, Pearson
- 3) The Art of R Programming, Norman Matloff, No starch Press.

**REFERENCE BOOKS:**

- 1) R Cookbook, Paul Teetor, Oreilly.
- 2) R in Action, Rob Kabacoff, Manning





**SOFTWARE ENGINEERING**

<b>Lecture : 3</b>	<b>Practical : 0</b>	<b>Internal Marks:30</b>
<b>Credits : 3</b>	<b>Tutorial : 0</b>	<b>External Marks:70</b>

**OUTCOMES**

At the end of the Course Students will be able to:

- CO1: Understand the software life cycle models and the importance of the software process.
- CO2: Acquire skills to think about problems and their solutions using appropriate methods of analysis and design.
- CO3: Understand the importance of modeling and modeling languages.
- CO4: Design and develop correct and robust software products.
- CO5: Understand the basics of deployment and maintenance of software development project.

**UNIT I-INTRODUCTION**

Software Engineering, Software Process-Generic process model-Prescriptive process model-specialized, unified process-Agile development-Agile Process-Extreme Programming-Other agile Process models-Software Engineering Knowledge-core Principles.

**UNIT II-REQUIREMENTS**

Requirements Engineering-Establishing the Groundwork-Eliciting Requirements-Developing use cases-Building the requirements model-Negotiating, validating Requirements-Requirements Analysis-Requirements Modeling Strategies.

**UNIT III-DESIGN**

MODELING WITH UML: Modeling Concepts and Diagrams-Use Case Diagrams-Class Diagrams-Interaction Diagrams-State chart Diagrams-Activity Diagrams-Package Diagrams-Component Diagrams-Deployment Diagrams

Design Process-Design concepts: Abstraction, Architecture, patterns, Separation of Concerns, Modularity, Information Hiding, Functional Independence, Refinement, Aspects, Refactoring, Object Oriented Design Concepts, Design Classes-Design Model: Data, Architectural, Interface, Component, Deployment Level Design Elements .

**UNIT IV-SOFTWARE IMPLEMENTATION**

Structured coding Techniques-Coding Styles-Standards and Guidelines-Documentation Guidelines-Modern Programming Language Features: Type checking-User defined data types-Data Abstraction-Exception Handling-Concurrency Mechanism.

**UNIT V-TESTING AND MAINTENANCE**

TESTING: Software Quality-Software Quality Dilemma-Achieving Software Quality-Testing: Strategic Approach to software Testing-Strategic Issues-Testing: Strategies for Conventional Software, Object oriented software, System Testing, Art of Debugging

MAINTENANCE: Software Maintenance-Software Supportability-Reengineering-Business Process Reengineering-Software Reengineering-Reverse Engineering-Restructuring-Forward Engineering-Economics of Reengineering

**TEXT BOOKS**

1. Roger S, "Software Engineering-A Practitioner's Approach", seventh edition, Pressman, 2010.
2. Pearson Edu, "Software Engineering by Ian sommerville", 9<sup>th</sup> edition, 2010.

**REFERENCES**

1. Hans Van Vliet, "Software Engineering: Principles and Practices", 2008.
2. Richard Firley, "Software Engineering Concepts", 2008.



**COMPUTER NETWORKS**

**Lecture : 3**

**Practical : 0**

**Internal Marks:30**

**Credits: 3**

**Tutorial : 0**

**External Marks:70**

**OUTCOMES:**

After completing this course the student must demonstrate the knowledge and ability to:

- Independently understand basic computer network technology.
- Identify the different types of network topologies and protocols.
- Enumerate the layers of the OSI model and TCP/IP. Explain the function(s) of each layer.
- Perform internal connections between the devices.
- Implement IP address and manipulate within the network.

**UNIT – I:**

**Introduction:** OSI overview, TCP/IP and other networks models, Examples of Networks: Novell Networks, Arpanet, Internet, Network Topologies WAN, LAN, MAN.

**UNIT – II:**

**Physical Layer and overview of PL Switching:** Multiplexing: frequency division multiplexing, wave length division multiplexing, synchronous time division multiplexing, statistical time division multiplexing, introduction to switching: Circuit Switched Networks, Datagram Networks, Virtual Circuit Networks.

**UNIT – III:**

**Data link layer:** Design issues, **Framing:** fixed size framing, variable size framing, flow control, error control, error detection and correction, CRC, Checksum: idea, one's complement internet checksum, services provided to Network, **Elementary Data Link Layer protocols:** simplex protocol, Simplex stop and wait, Simplex protocol for Noisy Channel. **Sliding window protocol:** One bit, Go back N, Selective repeat-Stop and wait protocol.

**UNIT – IV:**

**Random Access:** ALOHA, MAC addresses, Carrier sense multiple access (CSMA), CSMA with Collision Detection, CSMA with Collision Avoidance, Controlled Access: Reservation, Polling, Token Passing, Channelization: frequency division multiple access(FDMA), time division multiple access(TDMA), code division multiple access(CDMA).

**IEEE Standards:** – data link layer, physical layer, Manchester encoding, Standard Ethernet: MAC sub layer, physical layer, Fast Ethernet: MAC sub layer, physical layer, IEEE-802.11: Architecture, MAC sub layer, addressing mechanism, frame structure.

**UNIT – V:**

**Application layer (WWW and HTTP):** ARCHITECTURE : Client (Browser) ,Server ,Uniform Resource Department of Master of Computer Application



Locator, Resource Record, HTTP: HTTP Transaction, HTTP Operational Model and Client/Server Communication, HTTP Request Message Format, HTTP Response Message Format.

**TEXT BOOKS:**

1. Data Communications and Networks – Behrouz A. Forouzan. Third Edition TMH.
2. Computer Networks, 5ed, David Patterson, Elsevier
3. Computer Networks — Andrew S Tanenbaum, 4th Edition. Pearson Education/PHI
4. Computer Networks, Mayank Dave, CENGAGE

**REFERENCES:**

1. An Engineering Approach to Computer Networks-S.Keshav, 2nd Edition, Pearson Education
2. Understanding communications and Networks, 3rd Edition, W.A. Shay, Thomson



**DESIGN AND ANALYSIS OF ALGORITHMS**

<b>Lecture : 3</b>	<b>Practical : 0</b>	<b>Internal Marks:30</b>
<b>Credits : 3</b>	<b>Tutorial : 0</b>	<b>External Marks:70</b>

**COURSE OUTCOMES:**

- CO1) Understand the basic notation for analyzing the performance of the algorithms.
- CO2) Describe Divide-and-Conquer Paradigm
- CO3) Describe Greedy Paradigm to find the Optimal Solution.
- CO4) Describe Dynamic Programming Paradigm
- CO5) Describe and Distinguish Backtracking and Branch and Bound Paradigms

**UNIT 1 :**

**Introduction:** Algorithm, Pseudo code for expressing algorithms, performance Analysis-Space complexity, Time complexity, Asymptotic Notation- Big oh notation, Omega notation, Theta notation and Little oh notation, Solving Recurrence relations, Amortized Analysis

**UNIT 2 :**

**Divide and Conquer Method:** General Method, Applications: Binary search, Quick sort, Merge sort, Finding the Maximum and Minimum ,Defective Chessboard.

**UNIT 3:**

**Greedy Method:** General method, Applications: Minimum cost spanning tree(Prim's and Kruskal's Algorithms) , Optimal Merge Patterns ,Fractional Knapsack Problem, Single source shortest path, Job Sequencing with Deadlines.

**UNIT 4:**

**Dynamic programming:** General Method, Applications: Optimal Binary Search Tree, String Editing, 0/1 knapsack , Travelling Salesperson Problem ,Reliability Design.

**UNIT 5 :**

**Back tracking:** General Method, Applications: Sum of Subsets, Hamiltonian Cycles. Graph Coloring,8-Queen Problem.

**Branch and bound:** General Method , Applications: Travelling Salesman Problem,0/1 Knapsack problem using FIFO and LC Branch and Bound.

**Text Books:**

1. Fundamentals of computer algorithms E. Horowitz S. Sahni, University Press
2. Introduction to Algorithms by Thomas H. Cormen, Charles E. Leiserson ,Ronald L. Rivest, Clifford Stein . PHI Learning.

**Reference Books :**

- 1.The Design and Analysis of Computer Algorithms, Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman
2. Algorithm Design, Jon Kleinberg, Pearson.

**JAVA PROGRAMMING LAB****Lecture : 0****Practical : 4****Internal Marks:40****Credits : 2****Tutorial : 0****External Marks:60****COURSE OUTCOMES:**

1. Understand the behavior of primitive data types, object references, and arrays.
2. Implement Java classes from specifications
3. Implement interfaces, inheritance, and polymorphism as programming techniques
4. Apply exceptions handling
5. Design and develop GUI applications

**Exercise - 1 (Basics)**

- a). Write a JAVA program to display default value of all primitive data type of JAVA
- b). Write a java program that display the roots of a quadratic equation  $ax^2+bx=0$ . Calculate the discriminate D and basing on value of D, describe the nature of root.
- c). Five Bikers Compete in a race such that they drive at a constant speed which may or may not be the same as the other. To qualify the race, the speed of a racer must be more than the average speed of all 5 racers. Take as input the speed of each racer and print back the speed of qualifying racers.
- d) Write a case study on **public static void main(250 words)**

**Exercise - 2 (Operations, Expressions, Control-flow, Strings)**

- a). Write a JAVA program to search for an element in a given list of elements using binary search mechanism.
- b). Write a JAVA program to sort for an element in a given list of elements using bubble sort
- c). Write a JAVA program to sort for an element in a given list of elements using merge sort.
- d) Write a JAVA program using StringBuffer to delete, remove character.

**Exercise - 3 (Class, Objects)**

- a). Write a JAVA program to implement class mechanism. – Create a class, methods and invoke them inside main method.
- b). Write a JAVA program to implement constructor.

**Exercise - 4 (Methods)**

- a). Write a JAVA program to implement constructor overloading.
- b). Write a JAVA program implement method overloading.

**Exercise - 5 (Inheritance)**

- a). Write a JAVA program to implement Single Inheritance
- b). Write a JAVA program to implement multi level Inheritance
- c). Write a java program for abstract class to find areas of different shapes

**Exercise - 6 (Inheritance - Continued)**

- a). Write a JAVA program give example for “super” keyword.
- b). Write a JAVA program to implement Interface. What kind of Inheritance can be achieved?

**Exercise - 7 (Exception)**

- a). Write a JAVA program that describes exception handling mechanism
- b). Write a JAVA program Illustrating Multiple catch clauses

**Exercise – 8 (Runtime Polymorphism)**

- a). Write a JAVA program that implements Runtime polymorphism
- b). Write a Case study on run time polymorphism, inheritance that implements in above problem.



**Exercise – 9 (User defined Exception)**

- a). Write a JAVA program for creation of Illustrating throw
- b). Write a JAVA program for creation of Illustrating finally
- c). Write a JAVA program for creation of Java Built-in Exceptions
- d). Write a JAVA program for creation of User Defined Exception

**Exercise – 10 (Threads)**

- a). Write a JAVA program that creates threads by extending Thread class .First thread display “Good Morning “every 1 sec, the second thread displays “Hello “every 2 seconds and the third display “Welcome” every 3 seconds ,(Repeat the same by implementing Runnable)
- b). Write a program illustrating **isAlive** and **join ()**
- c). Write a Program illustrating Daemon Threads.

**Exercise - 11 (Threads continuity)**

- a). Write a JAVA program Producer Consumer Problem
- b). Write a case study on thread Synchronization after solving the above producer consumerproblem.

**Exercise – 12 (Packages)**

- a). Write a JAVA program illustrate class path
- b). Write a case study on including in class path in your os environment of your package.
- c). Write a JAVA program that import and use the defined your package in the previous Problem.

**Exercise - 13 (Applet)**

- a). Write a JAVA program to paint like paint brush in applet.
- b). Write a JAVA program to display analog clock using Applet.
- c). Write a JAVA program to create different shapes and fill colors using Applet.

**Exercise - 14 (Event Handling)**

- a). Write a JAVA program that display the x and y position of the cursor movement using Mouse.
- b). Write a JAVA program that identifies key-up key-down event user entering text in a Applet.

**Exercise - 15 (Swings)**

- a). Write a JAVA program to build a Calculator in Swings
- b). Write a JAVA program to display the digital watch in swing tutorial.

**Exercise – 16 (Swings - Continued)**

- a). Write a JAVA program that to create a single ball bouncing inside a JPanel.
- b). Write a JAVA program JTree as displaying a real tree upside down



**R PROGRAMMING LAB**

**Lecture : 0**

**Practical : 4**

**Internal Marks:40**

**Credits : 2**

**Tutorial : 0**

**External Marks:60**

**COURSE OUTCOMES:**

1. Implement the basic concepts of R.
2. Implement the basic data structures in R
3. Implement loops and functions in R
4. Implement mathematical functions and handling files
5. Use various graphical tools in R

1. Write a program to illustrate basic Arithmetic in R
2. Write a program to illustrate Variable assignment in R
3. Write a program to illustrate data types in R
4. Write a program to illustrate creating and naming a vector in R
5. Write a program to illustrate create a matrix and naming matrix in R
6. Write a program to illustrate Add column and Add a Row in Matrix in R
7. Write a program to illustrate Selection of elements in Matrixes in R
8. Write a program to illustrate Performing Arithmetic of Matrices
9. Write a program to illustrate Factors in R
10. Case study of why you need to use a Factor in R
11. Write a program to illustrate Ordered Factors in R
12. Write a program to illustrate Data Frame Selection of elements in a Data frame
13. Write a program to illustrate Sorting a Data frame
14. Write a program to illustrate List ? Why would you need a List
15. Write a program to illustrate Adding more elements into a List
16. Write a program to illustrate if-else-else if in R
17. Write a Program to illustrate While and For loops in R
18. Write a program to illustrate Logical & and Logical | operators in R.
19. Write a program to illustrate Functions in Quick sort implementation in R
20. Write a program to illustrate Function inside function in R
21. Write a program to illustrate to create graphs and usage of plot() function in R
22. Write a program to illustrate Customising and Saving to Graphs in R.
23. Write a program to illustrate some built in Mathematical Functions



Devineni Venkata Ramana & Dr. Hima Sekhar

**MIC COLLEGE OF TECHNOLOGY**  
**(Autonomous)**

---

# **SEMESTER IV**

  

# **SYLLABUS**





## **WEB TECHNOLOGIES**

**Lecture : 3**

**Practical : 0**

**Internal Marks:30**

**Credits : 3**

**Tutorial : 0**

**External Marks:70**

### **COURSE OUTCOMES:**

1. Describe the concepts of WWW including browser and HTTP protocol.
2. List the various HTML tags and use them to develop the user friendly web pages.
3. Define the CSS with its types and use them to provide the styles to the web pages at various levels.
4. Develop the modern web pages using the HTML and CSS features with different layouts as per need of applications.
5. Use the JavaScript to develop the dynamic web pages.
6. Use server side scripting with PHP to generate the web pages dynamically using the database connectivity.
7. Develop the modern Web applications using the client and server side technologies and the web design fundamentals.

### **UNIT-I:**

Introduction : Concept of WWW, Internet and WWW, HTTP Protocol :Request and Response, Web browser and Web servers, Features of Web Web Design: Concepts of effective web design, Web design issues including Browser, Bandwidth and Cache, Display resolution, Look and Feel of the Website, Page Layout and linking, User centric design, Sitemap, Planning and publishing website, Designing effective Navigation.

### **UNIT-II:**

HTML : Basics of HTML, formatting and fonts, commenting code, color, hyperlink, lists, tables, images, forms, XHTML, Meta tags, Character entities, frames and frame sets, Browser architecture and Website structure. Overview and features of HTML5

Style sheets : Need for CSS, introduction to CSS, basic syntax and structure, using CSS, background images, colors and properties, manipulating texts, using fonts, borders and boxes, margins, padding lists, positioning using CSS, CSS2, Overview and features of CSS3.

JavaScript: Client side scripting with JavaScript, variables, functions, conditions, loops and repetition, Pop up boxes, Advance JavaScript:

JavaScript and objects, JavaScript own objects, the DOM and web.

### **UNIT-III:**

XML: Introduction to XML, uses of XML, simple XML, XML key components, DTD and



Devineni Venkata Ramana & Dr. Hima Sekhar

# MIC COLLEGE OF TECHNOLOGY (Autonomous)

---

Schemas, Using XML with application. Transforming XML using XSL and XSLT

## **UNIT-IV:**

PHP : Introduction and basic syntax of PHP, decision and looping with examples, PHP and HTML, Arrays, Functions, Browser control and detection, string, Form processing, Files, Advance Features: Cookies and Sessions, Object Oriented Programming with PHP.

## **UNIT-V:**

PHP and MySQL: Basic commands with PHP examples, Connection to Server, creating database, selecting a database, listing database, listing table names, creating a table, inserting data, altering tables, queries, deleting database, deleting data and tables, PHPmyadmin and database bugs

## **TEXT BOOKS:**

1. Developing Web Applications, Ralph Moseley and M. T. Savaliya, Wiley-India
2. Web Technologies, Black Book, dreamtech Press
3. HTML 5, Black Book, dreamtech Press

## **REFERENCE BOOKS:**

1. Web Design, Joel Sklar, Cengage Learning
2. Developing Web Applications in PHP and AJAX, Harwani, McGrawHill
3. Internet and World Wide Web How to program, P.J. Deitel & H.M. Deitel, Pearson



## **PYTHON PROGRAMMING**

**Lecture : 3**

**Practical : 0**

**Internal Marks:30**

**Credits : 3**

**Tutorial : 0**

**External Marks:70**

**OBJECTIVES:** The objective of this course is to make students:

1. To define Python functions and call them.
2. To develop Python programs with conditionals and loops.
3. To use Python data structures –, lists, tuples, dictionaries.
4. To learn object- oriented programming concepts.

**Outcomes:** At the end of the subject, students will be able to:

1. Structure simple Python programs for solving problems.
2. Decompose a Python program into functions.
3. Represent compound data using Python lists, tuples, and dictionaries.
4. Read and write data from/to files in Python Programs.
5. To build software for real needs.

### **UNIT – I:**

**Introduction:** History of Python, Need of Python Programming, Application Basics of Python, Installing Python in Windows, Programming Using the REPL(Shell), Running Python Scripts, Variables, Assignment, Keywords, Comments, Input-Output, Indentation.

### **UNIT – II:**

**Types, Operators and Expressions:** Types - Integers, Strings, Booleans; Operators- Arithmetic Operators, Comparison (Relational) Operators, Assignment Operators, Logical Operators, Bitwise Operators, Membership Operators, Identity Operators, Expressions and order of evaluations Control Flow- if, if-elif-else, for, while, break, continue, pass

### **UNIT – III:**

**Data Structures** Lists – Introduction, Accessing, Operations, Slicing, Methods; Tuples, Sets, Dictionaries, Sequences. Comprehensions.

**Functions** - Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments,



Default Arguments, Variable-length arguments, Anonymous Functions, Fruitful Functions (Function Returning Values), Scope of the Variables in a Function - Global and Local Variables.

#### **UNIT – IV:**

**Modules:** Creating modules, import statement, from. Import statement, name spacing,

**Python packages,** Introduction to PIP, Installing Packages via PIP, Using Python Packages

**Files:** Persistence, reading and writing, format operator, filenames and paths, catching exceptions, databases, pipes, writing modules.

**Case Study :Brief Tour of the following Standard Packages-** System, Date and Time, Math, Random, Numpy, Matplotlib, Multithreading, Internet Access and Turtle Graphics

#### **UNIT – V:**

**Object Oriented Programming OOP in Python:** Classes, 'self variable', Methods, Constructor Method, Inheritance, Overriding Methods, Data hiding,

**Error and Exceptions:** Difference between an error and Exception, Handling Exception, try except block, Raising Exceptions, User Defined Exceptions

#### **TEXT BOOKS**

1. Learning Python, Mark Lutz, Orielly
2. Guido van Rossum and Fred L. Drake Jr, —An Introduction to Python – Revised and updated for Python 3.2, Network Theory Ltd., 2011.
3. John V Guttag, —Introduction to Computation and Programming Using Python“, Revised and expanded Edition, MIT Press , 2013.

#### **Reference Books:**

1. Think Python, Allen Downey, Green Tea Press
2. Core Python Programming, W.Chun, Pearson.
3. Introduction to Python, Kenneth A. Lambert, Cengage
4. “Python in easy steps In Easy Steps”, Mike MC Grath, illustrated edition, In easy steps 2013 publishers.
5. Professional Python Frameworks: Web 2.0 Programming, Dana Moore, Raymond Budd, William Wright, Wrox Publication, ISBN: 978-0-470-13809-0, October 2007.



**DATA WAREHOUSING AND DATA MINING**

<b>Lecture : 3</b>	<b>Practical : 0</b>	<b>Internal Marks:30</b>
<b>Credits : 3</b>	<b>Tutorial : 0</b>	<b>External Marks:70</b>

**COURSE OUTCOMES:**

- 1) Understand Data Warehouse fundamentals, Data Mining Principles
- 2) Design data warehouse with dimensional modeling and apply OLAP operations.
- 3) Identify appropriate data mining algorithms to solve real world problems
- 4) Compare and evaluate different data mining techniques like classification, prediction, clustering and association rule mining

**COURSE OBJECTIVES:**

- 1) Identify the scope and necessity of Data Mining & Warehousing for the society.
- 2) Describe the design of Data Warehousing so that it can be able to solve the root problems.
- 3) To understand various tools of Data Mining and their techniques to solve the real time problems.
- 4) To develop ability to design various algorithms based on data mining tools.
- 5) To develop further interest in research and design of new Data Mining Techniques.

**Unit – I Warehouse:** What is it, Who Need It, and Why?, Things to Consider, Managing the Data Warehouse, Data Warehouse Design Methodology, Data Marts and Star Schema Design, Fundamentals of ETL Architecture, Partitioning Data, Indexing Data.

**Unit – II Data mining:** Introduction, Data mining on What Kind of Data, Data Mining Functionalities, Classification of Data Mining Systems, Major Issues in Data Mining. **Data Preprocessing:** Why Preprocess the Data? Data Cleaning, Data Integration, Data Transformation, Data Reduction, Data Discretization.

**Unit – III Classification and Prediction:** Introduction, Classification by Decision Tree Induction, Bayesian Classification. Other Classification Methods, Classification by Back Propagation, Prediction, Classifier Accuracy.

**Unit – IV Association Analysis:** Mining Association Rules in large databases - Association Rule Mining, Mining Single-Dimensional Boolean Association Rules from Transactional Databases, Mining Multi-Dimensional Association Rules from Relational Databases and Data Warehouses.



**Unit – V Cluster analysis** – Introduction, Types of Data in Cluster Analysis, A Categorization of Major Clustering Methods, Partitioning Methods, Hierarchical Methods, Density Based Methods: DBSCAN, Grid-Based method: STING, Model Based Clustering Method: Statistical Approach, Outlier Analysis.

**Prescribed Books:**

- 1) Michael Corey, Michael Abbey, Ian Abramson, Ben Taub, “Oracle 8i Data Warehousing”, TMH (Unit-I)
- 2) Jiawei Han Micheline Kamber, “Data mining & Techniques”, Morgan Kaufmann Publishers (Units-II-V)
- 3) Pang-Ning Tan, Michael Steinbach, Vipin Kumar, “Introduction to Data Mining”, Pearson Publications. (Units-III-V)

**Reference Books:**

- 1) S.N.Sivanandam, S.Sumathi, “Data Mining – Concepts, Tasks and Techniques”, Thomson (2006).
- 2) Ralph Kimball, “The Data Warehousing Toolkit”, Wiley.
- 3) Margaret H. Dunham, “Data mining - Introductory and advanced topics”, Pearson Education.
- 4) D.Hand, H. Mannila and P.Smyth, “Principles of Data mining”, PHI (2001).



# MIC COLLEGE OF TECHNOLOGY (Autonomous)

## E COMMERCE

<b>Lecture : 3</b>	<b>Practical : 0</b>	<b>Internal Marks:30</b>
<b>Credits : 3</b>	<b>Tutorial : 0</b>	<b>External Marks:70</b>

### **UNIT-I: Electronic Commerce**

Introduction, Anatomy of E Commerce Applications, E Commerce Consumer , Applications, E Commerce organization Applications, Consumer oriented E commerce, Mercantile Process Models

### **UNIT-II: Electronic Payment Systems**

Introduction, Digital Token Based Systems, Smart Cards, Credit Cards, Risks in Electronic Payment Systems, Inter Organizational Commerce, EDI- EDI Implementation, Value Added Networks

### **UNIT-III: Inter Organizational Commerce**

Introduction to Inter Organizational Commerce, Workflow, Automation, Customization , Internal Commerce, Supply Chain Management

### **UNIT-IV: Corporate Digital Library**

Document Library, Digital Document Types, Corporate Data Warehouses, Advertising and Marketing, Information Bases Marketing, Advertising on Internet, Online Marketing Process, Market Research

### **UNIT-V: Consumer Search and Resource Discovery& Multimedia**

Introduction, Information Search and Retrieval, Commerce Catalogues, Information Filtering, Key Multimedia Concepts, Digital Video and Electronic Commerce, Desktop Video Processing

### **TEXT BOOKS:**

1. Frontiers of electronic commerce – Kalakata, Whinston, Pearson

### **REFERNCE BOOKS:**

1. E-Commerce fundamentals and applications Hendry Chan, Raymond Lee, Tharam Dillon, Ellizabeth Chang, John Wiley.
2. E-Commerce, S.Jaiswal – Galgotia.
3. E-Commerce, Efrain Turbon, Jae Lee, David King, H.Michael Chang.
4. Electronic Commerce – Gary P.Schneider – Thomson.
5. E-Commerce – Business, Technology, Society, Kenneth C.Taudon, Carol Guyerico Tra



**CRYPTOGRAPHY AND NETWORK SECURITY**

**(Elective-I)**

**Lecture : 3**

**Credits : 3**

**Practical : 0**

**Tutorial : 0**

**Internal Marks:30**

**External Marks:70**

**COURSE OBJECTIVES:**

At the end of the course the student is able to understand

- Basics of Information Security
- How to provide security with Symmetric Encryption Methods
- How to provide security with Asymmetric Encryption Methods
- About E-Mail Privacy
- Web Security
- Basic principles of SNMP
- About Firewalls
- 

**COURSE OUTCOMES:**

1. Be able to individually reason about software security problems and protection techniques on both an abstract and a more technically advanced level.
2. Be able to individually explain how software exploitation techniques, used by adversaries, function and how to protect against them.

**UNIT I : Classical Encryption Techniques**

Introduction: Security attacks, services & mechanisms, Symmetric Cipher Model, Substitution Techniques, Transportation Techniques, Cyber threats and their defense( Phishing Defensive measures, web based attacks, SQL injection & Defense techniques)(TEXT BOOK 2), Buffer overflow & format string vulnerabilities, TCP session hijacking(ARP attacks, route table modification) UDP hijacking ( man-in-the-middle attacks)(TEXT BOOK 3).

**UNIT II: Block Ciphers & Symmetric Key Cryptography**

Traditional Block Cipher Structure, DES, Block Cipher Design Principles, AES-Structure, Transformation functions, Key Expansion, Blowfish, CAST-128, IDEA, Block Cipher Modes of Operations

**UNIT III: Number Theory & Asymmetric Key Cryptography**

Number Theory: Prime and Relatively Prime Numbers, Modular Arithmetic, Fermat's and Euler's Theorems, The Chinese Remainder theorem, Discrete logarithms.

Public Key Cryptography: Principles, public key cryptography algorithms, RSA Algorithms, Diffie Hellman Key Exchange, Elgamal encryption & decryption, Elliptic Curve Cryptography.





# MIC COLLEGE OF TECHNOLOGY

## (Autonomous)

---

### **UNIT IV: Cryptographic Hash Functions & Digital Signatures, IP Security**

Application of Cryptographic hash Functions, Requirements & Security, Secure Hash Algorithm, Message Authentication Functions, Requirements & Security, HMAC & CMAC. Digital Signatures, NIST Digital Signature Algorithm. Key management & distribution.

IP Security: IP Security Overview, IP Security Architecture, Authentication Header, Encapsulating Security Payload, Combining Security Associations and Key Management.

### **UNIT V: User Authentication, Transport Layer Security & Email Security, Intrusion detection systems**

User Authentication: Remote user authentication principles, Kerberos

Transport Level Security: Web Security Requirements, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Shell(SSH)

Electronic Mail Security: Pretty Good Privacy (PGP) and S/MIME.

Intrusion detection: Overview, Approaches for IDS/IPS, Signature based IDS, Host based IDS/IPS. (TEXT BOOK 2)

### **TEXT BOOKS:**

1. Cryptography & Network Security: Principles and Practices, William Stallings, PEA, Sixth edition.
2. Introduction to Computer Networks & Cyber Security, Chwan Hwa Wu, J.David Irwin, CRC press
3. Hack Proofing your Network, Russell, Kaminsky, Forest Puppy, Wiley Dreamtech.

### **REFERENCE BOOKS:**

1. Everyday Cryptography, Fundamental Principles & Applications, Keith Martin, Oxford
2. Network Security & Cryptography, Bernard Menezes, Cengage, 2010



**Distributed Systems**  
**(Elective-I)**

**Lecture : 3**  
**Credits : 3**

**Practical : 0**  
**Tutorial : 0**

**Internal Marks:30**  
**External Marks:70**

**COURSE OUTCOMES:**

The student should be made to:

- Understand foundations of Distributed Systems.
- Understand about various issues in Inter-process communication.
- Introduce the idea of peer to peer services and file system.
- Understand in detail the system level and support required for distributed system.
- Understand the issues involved in distributed transactions & concurrency control.

**UNIT I**

**Examples of Distributed Systems**–Trends in Distributed Systems, Focus on resource sharing, Challenges, Case study: World Wide Web.

**System Models** – Physical Model, Architectural Model, Fundamental Model.

**UNIT II**

**Interprocess Communication:** Introduction, The API for the Internet Protocols, The Characteristics of Interprocess communication. Sockets, UDP Datagram Communication, TCP Stream Communication; External Data Representation and Marshalling;

**Multicast Communication** – IP Multicast, Reliability & ordering of multicast; Network virtualization: Overlay networks – Example.

**UNIT III**

**Peer-to-peer Systems** – Introduction, Napster and its legacy, Peer-to-peer, Middleware, Routing overlays, Overlay case studies: Pastry, Tapestry.

**Distributed File Systems** – Introduction, File service architecture, Network File system, Andrew File system, Enhancements and further developments.

**UNIT IV**

**Distributed Objects and Remote Invocation:** Introduction, Communication between Distributed Objects- Object Model, Distributed Object Model; Design Issues for RMI, Implementation of RMI, Distributed Garbage Collection; Remote Procedure Call, Events and Notifications, Case Study: JAVARMI

**Operating System Support:** Introduction, the Operating System Layer, Protection, Processes and Threads -Address Space, Creation of a New Process, Threads.

**UNIT V**

**Transactions and Concurrency Control**– Transactions -Nested transactions, Locks, Optimistic concurrency control, Timestamp ordering, Comparison of methods for concurrency control,



Devineni Venkata Ramana & Dr. Hima Sekhar

# MIC COLLEGE OF TECHNOLOGY (Autonomous)

---

**Distributed Transactions** - Flat and nested distributed transactions, Atomic Commit protocols, Concurrency Control in Distributed Transactions, Distributed deadlocks, Transaction recovery.

**TEXT BOOKS:**

1. George Coulouris, Jean Dollimore, Tim Kindberg, "Distributed Systems- Concepts and Design", Fourth Edition, Pearson Publication.
2. Ajay D Kshemkalyani, Mukesh Sigal, "Distributed Computing, Principles, Algorithms and Systems", Cambridge.



## **ARTIFICIAL INTELLIGENCE**

### **(Elective-I)**

**Lecture : 3**  
**Credits : 3**

**Practical : 0**  
**Tutorial : 0**

**Internal Marks:30**  
**External Marks:70**

### **COURSE OBJECTIVES:**

- To learn the difference between optimal reasoning vs human like reasoning
- To understand the notions of state space representation, exhaustive search, heuristic search along with the time and space complexities
- To learn different knowledge representation techniques
- To understand the applications of AI: namely Game Playing, Theorem Proving, Expert Systems, Machine Learning and Natural Language Processing

### **UNIT - I**

Introduction, History, Intelligent Systems, Foundations of AI, Sub areas of AI, Applications. Problem Solving - State-Space Search and Control Strategies: Introduction, General Problem Solving, Characteristics of Problem, Exhaustive Searches, Heuristic Search Techniques, Iterative-Deepening A\*, Constraint Satisfaction. Game Playing, Bounded Look-ahead Strategy and use of Evaluation Functions, Alpha-Beta Pruning

### **UNIT - II**

Logic Concepts and Logic Programming: Introduction, Propositional Calculus, Propositional Logic, Natural Deduction System, Axiomatic System, Semantic Tableau System in Propositional Logic, Resolution Refutation in Propositional Logic, Predicate Logic, Logic Programming.

Knowledge Representation: Introduction, Approaches to Knowledge Representation, Knowledge Representation using Semantic Network, Extended Semantic Networks for KR.

### **UNIT - III**

Expert System and Applications: Introduction, Phases in Building Expert Systems, Expert System Architecture, Expert Systems Vs Traditional Systems, Truth Maintenance Systems, Application of Expert Systems, List of Shells and Tools.

Uncertainty Measure - Probability Theory: Introduction, Probability Theory, Bayesian Belief

Networks, Certainty Factor Theory, Dempster-Shafer Theory.

### **UNIT-IV**

Machine-Learning Paradigms: Introduction. Machine Learning Systems, Supervised and Unsupervised Learning. Inductive Learning. Learning Decision Trees, Deductive Learning. Clustering, Support Vector Machines.

### **UNIT-V**

Artificial Neural Networks: Introduction, Artificial Neural Networks, Single-Layer



Devineni Venkata Ramana & Dr. Hima Sekhar

# MIC COLLEGE OF TECHNOLOGY (Autonomous)

---

Feed-Forward Networks, Multi-Layer Feed-Forward Networks, Radial-Basis Function Networks, Design Issues of Artificial Neural Networks, Recurrent Networks.

## **TEXT BOOKS:**

1. Saroj Kaushik. Artificial Intelligence. Cengage Learning. 2012. Russell, Norvig: Artificial intelligence, A Modern Approach, Pearson Education, Second Edition. 2004

## **REFERENCE BOOK:**

1. Rich, Knight, Nair: Artificial intelligence, Tata McGraw Hill, Third Edition 2009.
2. Introduction to Artificial Intelligence by Eugene Charniak, Pearson.
3. Introduction to Artificial Intelligence and expert systems Dan W.Patterson. PHI.
4. Artificial Intelligence by George Fluger Pearson fifth edition.



**PYTHON PROGRAMMING LAB**

**Lecture : 0**  
**Credits : 2**

**Practical : 4**  
**Tutorial : 0**

**Internal Marks:40**  
**External Marks:60**

**Exercise 1 - Basics**

- a) Running instructions in Interactive interpreter and a Python Script
- b) Write a program to purposefully raise Indentation Error and Correct it

**Exercise 2 - Operations**

- a) Write a python program to compute distance between two points taking input from the user (Pythagorean Theorem)
- b) Write a program add.py that takes 2 numbers as command line arguments and prints its sum.
- c) Write a program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon user's choice.

**Exercise - 3 Control Flow**

- a) Write a Program for checking whether the given number is a even number or not.
- b) WAP to calculate total marks, percentage and grade of a student. Marks obtained in each of the three subjects are to be input by the user. Assign grades according to the following criteria :
  - Grade A: Percentage  $\geq 80$
  - Grade B: Percentage  $\geq 70$  and  $< 80$
  - Grade C: Percentage  $\geq 60$  and  $< 70$
  - Grade D: Percentage  $\geq 40$  and  $< 60$
  - Grade E: Percentage  $< 40$
- c) Using a for loop, write a program that prints out the decimal equivalents of  $1/2$ ,  $1/3$ ,  $1/4$ ,  $\dots$ ,  $1/10$
- d) Write a program using a for loop that loops over a sequence. What is sequence ?
- e) Write a program using a while loop that asks the user for a number, and prints a countdown from that number to zero.



#### **Exercise 4 - Control Flow - Continued**

- a) Find the sum of all the primes below two million. Each new term in the Fibonacci sequence is generated by adding the previous two terms. By starting with 1 and 2, the first 10 terms will be: 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...
- b) By considering the terms in the Fibonacci sequence, find the sum of the even-valued terms.

#### **Exercise - 5 - DS**

- a) Write a Python program using string slices and methods.
- b) Implement a Python program using Lists and perform various operations.
- c) Write a Python program to implement Dictionaries.

#### **Exercise - 6 Files**

- a) Write a program to print each line of a file in reverse order.
- b) Write a program to compute the number of characters, words and lines in a file.
- c) Write a program to count frequency of characters in a given file. Can you use character frequency to tell whether the given file is a Python program file, C program file or a text file?

#### **Exercise – 7 Functions**

- a) Write a function `ball_collide` that takes two balls as parameters and computes if they are colliding. Your function should return a Boolean representing whether or not the balls are colliding.

Hint: Represent a ball on a plane as a tuple of  $(x, y, r)$ ,  $r$  being the radius. If (distance between two balls centers)  $\leq$  (sum of their radii) then (they are colliding)

- b) Write a function `cumulative_product` to compute cumulative product of a list of numbers.

#### **Exercise - 8 Functions - Continued**

- a) Write a python program to display the powers of 2 using anonymous function.
- b) Write function to compute gcd, lcm of two numbers. Each function shouldn't exceed one line.
- c) Write a Python program to implement
  - a) Linear search
  - b) Binary search using functions.

#### **Exercise 10 – Numpy Module**



Devineni Venkata Ramana & Dr. Hima Sekhar  
**MIC COLLEGE OF TECHNOLOGY**  
**(Autonomous)**

---

- a) Write a program to perform addition of two square matrices
- b) Write a program to perform multiplication of two square matrices

**Exercise - 11 – Internet Access Module**

- a) Write a script that imports requests and fetch content from the page. Eg. (Wiki)
- b) Write a simple script that serves a simple HTTP Response and a simple HTML Page

**Exercise -12 Visualization Modules**

- a) WAP to read n integers and display them as a histogram.
- b) WAP to display sine, cosine, polynomial and exponential curves.
- c) WAP to plot a graph of people with pulse rate p vs. height h. The values of p and h are to be entered by the user.
- d) Develop a Python program that implements the Turtle module.

**Exercise - 13 OOP**

- a) Write a python program to demonstrate Constructors.
- b) Write a python program to demonstrate Inheritance
- c) Write a python program to handle Runtime Exceptions.





Devineni Venkata Ramana & Dr. Hima Sekhar  
**MIC COLLEGE OF TECHNOLOGY**  
**(Autonomous)**

---

**WEB TECHNOLOGIES LAB**

**Lecture : 0**  
**Credits : 2**

**Practical : 4**  
**Tutorial : 0**

**Internal Marks:40**  
**External Marks:60**

1. Design multiple webpages with navigation facility by utilizing all the html elements in Unit II
2. Design multiple page website with different types of styling like Inline Internal and External. Note: By the end of this task student should be clear on cascading.
3. Design a webpage to perform client side actions like validations and all using JavaScript.
4. Example on usage of DTD.
5. Example on usage of Schema.
6. Different examples using XSL and XSLT.
7. Develop a website to use PHP Cookies for Logins of Different users
8. Develop a website to use PHP MYSQL for Logins of Different users
9. Example using PHP Arrays All types.
10. Example using PHP Sessions and use them in handling a website with login system.
11. Develop an attractive Web site for an event to be organized in your institute.
12. Develop a Web based application to manage the Visiting Cards which allows user to add new cards, delete the cards, update the cards etc.
13. Develop a web based application for online purchasing of products with payment facility



**UNIFIED MODELING LAB**

**Lecture : 0**  
**Credits : 2**

**Practical : 4**  
**Tutorial : 0**

**Internal Marks:40**  
**External Marks:60**

**OBJECTIVES:**

- Construct UML diagrams for static view and dynamic view of the system.
- Generate creational patterns by applicable patterns for given context.
- Create refined model for given Scenario using structural patterns.
- Construct behavioral patterns for given applications.

Case Study 1: ATM

Case Study 2: Library Management System

Case Study 3: Railway Reservation System

Case Study 4: Online Shopping

**Week 1:**

**Familiarization with Rational Rose or Umbrello**

**For each case study:**

**Week 2, 3 & 4:**

**For each case study:**

- a) Identify and analyze events
- b) Identify Use cases
- c) Develop event table
- d) Identify & analyze domain classes
- e) Represent use cases and a domain class diagram using Rational Rose
- f) Develop CRUD matrix to represent relationships between use cases and problem domain classes

**Week 5 & 6:**

**• For each case study:**

- a) Develop Use case diagrams
- b) Develop elaborate Use case descriptions & scenarios
- c) Develop system sequence diagrams

**Week 7, 8, 9 & 10:**

**For each case study:**

- a) Develop high-level sequence diagrams for each use case
- b) Develop Detailed Sequence Diagrams / Communication diagrams for each use case showing interactions among all the three-layer objects
- c) Develop detailed design class model (use GRASP patterns for responsibility assignment)
- d) Develop three-layer package diagrams for each case study



Devineni Venkata Ramana & Dr. Hima Sekhar  
**MIC COLLEGE OF TECHNOLOGY**  
**(Autonomous)**

---

**Week 11 & 12:**

**For each case study:**

- a) Develop Use case Packages
- b) Develop component diagrams
- c) Identify relationships between use cases and represent them
- d) Refine domain class model by showing all the associations among classes

**Week 13**

**For each case study:**

- a) Develop sample diagrams for other UML diagrams - state chart diagrams, activity diagrams and deployment diagrams

**OUTCOMES:**

- Understand the Case studies and design the Model.
- Understand how design patterns solve design problems.
- Develop design solutions using creational patterns.
- Construct design solutions by using structural and behavioral patterns