

**Parvatibai Chowgule College of Arts and Science
(Autonomous)**

DEPARTMENT OF COMPUTER SCIENCE

COURSE STRUCTURE

THREE YEAR B.Sc. DEGREE COURSE IN COMPUTER SCIENCE

COMPONENT A:

| SEMESTER | CORE COMPULSORY | | CORE ELECTIVE | | | |
|----------|--|--|---------------|-----|-----|-----|
| I | COM-I.C-1 Mathematical foundation of Computer Science - I | COM-I.C-2 * Introduction to Programming | --- | --- | --- | --- |
| | | | | | | |
| II | COM-II.C-3 ** Object Oriented Programming | COM-II.C-4 * Data Structures | --- | --- | --- | --- |

| | | | | | | |
|-----|--|-----|--|--|--|--------------------------------------|
| | | | | | | |
| | | | | | | |
| III | COM-III.C-5 * Data Base Management Systems | --- | COM-III. E-1 Software Engineering | COM-III. E-2 Digital Logic Design | COM-III. E-3 Mathematical Foundation of Computer Science - II | COM-III. E-4 Web Designing |
| | | | | | | |
| IV | COM-IV.C-6 Computer Architecture and Organization | --- | COM-IV. E-5 Design & Analysis of Algorithms | COM-IV. E-6 Data Base Application Development | COM-IV. E-7 Server Side Programming | COM-IV. E-8 HCI |
| | | | | | | |
| V | COM-V.C-7 * Operating Systems | --- | COM-V. E-9 Embedded Systems | COM-V. E-10 Mobile Application Development | COM-V. E-11 Introduction to Data Science | COM- V. E-12 Software Testing |

| | | | | | | |
|----|---------------------------------------|-----|---|--|--|--|
| | | | | | | |
| | | | | | | |
| VI | COM-VI.C-8 * Computer Networks | --- | COM-VI. E-13 Network Security | COM-VI. E-14 Cloud Computing | COM-VI. E-15 Multimedia Techniques | COM-VI. E-16 Digital Marketing |
| | | | | | | |

Note: * Core Compulsory papers also offered for minor subject combination.

** Core Compulsory papers also offered for minor subject combination in 4th Semester.

COMPONENT B:

1. GENERAL COMPULSORY (GC-B) - a. Cyber Security
2. INTERDISCIPLINARY (GC-E) - a. Multimedia
b. Python Programming

B.Sc. Computer Science – Syllabus
(from Academic Year 2017-18)
Semester I

Paper Title: Mathematical Foundation of Computer Science I

Paper Code: COM-I.C-1

Marks: 75

Credits: 3

Prerequisite Courses : Nil

Course Objectives:

- To build mathematical foundations that are essential requirement in understanding various concepts related to computer science.

Learning Outcome:

On completion of the course students will learn the concepts of the following:

- Combination and permutation.
- Numbers systems and conversions
- Boolean Algebra and Logic
- Set, Relations and Functions
- Finite Machines

1. **Combinatory:** [8L]
Permutations; Combinations; Counting; Summation; generating functions; recurrence relations.
2. **Binary Number System:** [7L]
Decimal to binary conversion and vice versa, binary number representation (signed, 1's Complement and 2's complement) binary addition, subtraction, binary to octal, hexadecimal conversion and vice versa. Floating point representation.
3. **Boolean Algebra:** [7L]
Boolean functions, truth table, DeMorgan's theorem, logic gates, Realization of Boolean Function using logic gates, Simplification using Karnaugh map.
4. **Set, Relations and Functions:** [9L]
Venn diagram, set operations, relations and properties, closures, equivalence relations, Partial ordering, functions, function types, inverse of functions, composition of functions, recursive functions, growth of functions.
5. **Logic:** [6L]
Propositional logic, first order logic, mathematical induction, deduction, proof by contradiction, program correctness.

6. Grammars, Languages and Automation:

[8L]

Grammars and languages, finite automation of finite state machines, regular languages, regular expressions.

Text Book:

Rosen H. Kenneth, *Discrete Mathematics and its Applications*, Tata McGraw Hill, seventh edition, 2011.

Reference:

Sarkar Kumar Swapan, *A Textbook of Discrete Mathematics*, S Chand & Company, 2005.

Practical: Mathematical Foundation of Computer Science I

Credit: 1

Marks: 25

Programs using C Language: (Any one from the choices a), b), c))

- 1) Generate all permutations of n symbols, where $2 \leq n \leq 5$ is user defined.
- 2) a) Read an integer. Convert it into binary number.
b) Read a binary number. Convert it into decimal number.
c) Read a number. Convert it into normalized form.
d) Read a binary number. Convert it into octal number.
- 3) Read a string of decimal digits. Find the frequency distribution of digits.
- 4) Read binary string. Check the occurrence of the pattern 1001 in the string.
- 5) a) Read two binary numbers. Add them using 1's complement method.
b) Read two binary numbers. Add them using 2's complement method.
- 6) Read two integer numbers. Find their GCD using recursion.
- 7) a) Read the value of p . Find the p -th Fibonacci number from the following recurrence relation.
 $f(0) = 0, f(1) = 1, f(n) = f(n-1) + f(n-2), n \geq 2$
b) The Ackermann function is defined as follows:
$$A(m, n) = \begin{cases} n + 1 & \text{if } m = 0 \\ A(m - 1, 1) & \text{if } m > 0 \text{ and } n = 0 \\ A(m - 1, A(m, n - 1)) & \text{if } m > 0 \text{ and } n > 0. \end{cases}$$

Find $A(m, n)$ for given m, n .
- 8) Given two functions $f(x) = x^3 + 2x + 3$, and $g(x) = 3x^2/4 + 10$, find $f \circ g(x)$.
- 9) Read an expression. Check whether it has equal number of '(' and ')' brackets, equal Number of '{' and '}' brackets, and equal number of '[' and ']' brackets.
- 10) Consider a set consisting of n numbers, where n is user defined. Read a set. Check whether a given number is a member of the set.

- 11) a) Read two sets. Find their union.
b) Read two sets. Find their intersection.
- 12) Read an expression containing on parentheses (and). Check whether it is properly parenthesized.
- 13) Applications of set theory, finite state machines, matrices, Boolean algebra, gates, ...
- 14) Bit-wise operations using C

Paper Title: Introduction to Programming

Paper Code: COM-I.C-2

Marks: 75

Credits: 3

Prerequisite Courses : Nil

Course Objectives:

- To make the student understand the concept of basic computer algorithm and use the algorithm for various problem solving.
- To implement algorithms using high level programming language.
- To understand basic principles of structured programming – example C

Learning Outcomes:

On completion of the course students will learn the following:

- Designing algorithms for a given problem.
- Writing C Programs to implement the algorithms.

1. **Introduction to Computer Problem Solving:** [5L]
Algorithm, Flowchart, The Problem Solving Aspect, General problem solving strategies, Top-Down Design, Implementation of Algorithms, Efficiency of Algorithms, Recursive algorithms.
2. **Basic Algorithms:** [3L]
Exchanging the values, Summation of a set of numbers, factorial computation, generation of the Fibonacci series, reversing the digits of an integer, base conversion.
3. **Factoring Methods:** [2L]
Finding divisors of an integer, finding the Greatest Common Divisor of two integers, generating prime numbers, computing prime factors of an integers.
4. **Sorting and Searching algorithms** [2L]
Bubble sort, Insertion Sort, Sequential Search and Binary Search.

5. **Introduction to ‘C’:** [3L]
History, Structure of a C program, Keywords, Identifiers, variables, constants, data types, Arithmetic Operators & Expressions, Logical operators and Relational Operators, Precedence and Associativity rules.
6. **Conditions and Iterations:** [3L]
Conditions and Actions, Condition statement, Simple control statement (*if, if-else, switch*), Iterative control statements (*for, while, do-while*).
7. **Functions:** [5L]
What is a function, Advantages of functions, Standard library functions; User defined functions – declaration, definition, function call, parameter passing, return keyword. Scope of variables, Storage classes, Recursion.
8. **Arrays:** [4L]
One and Two dimensional arrays: Array declaration, initialization, accessing the values, passing arrays to functions.
9. **Pointers:** [4L]
Pointer declaration, initialization, Pointer arithmetic, Pointer to Pointer, Arrays and Pointers, Functions and Pointers – passing pointers to functions, function returning pointer, dynamic memory allocation.
10. **Strings:** [4L]
Declaration and initialization, standard library string functions, strings and pointers, array of strings.
11. **Structure and Union:** [4L]
Creating structures, accessing structure members, array of structures, passing structure to functions, nested structure, pointers and structures, union, difference between structures and unions.
12. **File Handling:** [4L]
FILE variable, file access modes, operations on files, random access to files, command line arguments.
13. **Preprocessing:** [2L]
Format of Preprocessor directive, File Inclusion directive, Macro substitution, conditional compilation.

Text Books:

1. Dromey R.G., *How to solve it by computer*, Prentice Hall of India, 2nd Edition, 2004.
2. Kanetkar Yeshwant, *Let us C*, BPB Publications, 13th Edition, 2012.
3. Behrouz Forouzan, Richard Gilberg, *Computer Science: A Structure Programming Approach using C*, Cengage Learning 3rd Edition, 2013.

Reference books:

1. Horowitz Ellis, Sahni Satraj, Sanguthevar Rajasekaran, *Fundamentals of Computer algorithm*, Orient Longman, 2nd Edition, 2008.
2. Gottfried Byron, *Programming with C*, Tata McGraw Hill, 3rd Edition, 2010.
3. Brain W. Kernighan and Dennis M Ritchie, *The C Programming Language*, Prentice Hall India, 2nd Edition, 1988.

Practical: Introduction to Programming**Credit: 1****Marks: 25**

Programs using C language that covers the following concepts:

1. Program to compute a given formula.
2. Conditions
 - if..else
 - nested if
3. Iterative Control Statements
 - for
 - while
 - do...while
4. Functions.
 - Standard Library functions
 - Call by Value
 - Call by reference
5. Recursive functions.
6. Arrays.
 - One Dimensional Arrays
 - Two Dimensional Arrays
7. Sorting
 - Bubble sort
 - Insertion sort
8. Searching.
 - Sequential search
 - Binary search
9. Pointers.
 - Arrays and Pointers
 - Function returning pointers
 - Dynamic memory allocation
10. Strings.
 - Standard Library string functions
 - Strings and Pointers
11. Array of Strings.

12. Structure.

- Array of structures
- Passing Structure to functions
- Nested structure
- Structure and Pointer

13. Union.

14. File Handling.

- Text file
- Binary file
- Random Access to a file
- Command Line arguments

14. C Preprocessing.

- Macro expansion
- Conditional compilation

Semester II

Paper Title: Object Oriented Programming

Paper Code: COM-II.C-3

Marks : 75

credits: 3

Prerequisite Courses : Nil

Course Objectives:

- To teach the basic concepts and techniques which form the object oriented programming paradigm
- To introduce object oriented programming (OOP) using Java.

Learning Outcomes:

- Understand the concept and underlying principles of Object-Oriented Programming.
- Understand how object-oriented concepts are incorporated into the Java programming language.
- Develop problem-solving and programming skills using the OOP concept.

Syllabus

Principles of OOP

(3 L)

Programming paradigms. Basic concepts in OOP. OOP: major principles - encapsulation, abstraction, inheritance, polymorphism. Benefits of OOP. Applications of OOP.

Introduction to Java

(8 L)

Basics of Java programming, Data types, Variables, Operators, Control structures including selection, Looping, Java methods, Overloading, java.Math class, Arrays in java.

Objects and Classes

(9 L)

Basics of objects and classes in java, Constructors, Finalizer, Visibility modifiers, Methods and objects, object serialization, Inbuilt classes like String, Character, StringBuffer, File, this reference.

Inheritance and Polymorphism

(9 L)

Inheritance in java, Super and sub class, Overriding, java.lang.Object class, Polymorphism, Dynamic binding, Generic programming, Casting objects, Instance of operator, Abstract class, Interface in java, Package in java, java.util package.

Event and GUI programming

(6 L)

Design patterns – what and why? It's classification. Introduce the Observer design pattern.

Event handling in java, Event types, Mouse and key events, GUI Basics, Panels, Frames, Layout Managers: Flow Layout, Border Layout, Grid Layout, GUI components like Buttons, Check Boxes, Radio Buttons, Labels, Text Fields, Text Areas, Combo Boxes, Lists, Scroll Bars, Sliders, Windows, Menus, Dialog Box, Applet and its life cycle.

Multithreading in java**(4 L)**

Multithreading in java, Thread life cycle and methods, Runnable interface, Thread synchronization.

Exception handling**(3 L)**

Exception handling – what and why? Try and catch block. Multiple catch blocks. Nested try, finally block, throw keyword, throws keyword. Custom Exception.

Introduction to the Collections Framework.**(3 L)****Text Book**

Paul Deitel & H. Deitel, , Java - How to Program, Prentice Hall Publications 9th Edition,

Suggested Reference material

1. Patrick Naughton, Herbert Schildt, 2000, Java 2 – The Complete ReferenceTMH publications 9th Edition
2. Patrick Naughton, 1997, The Java Handbook –TMH publications
3. E. Balaguruswamy 2009,Programming with Java – A Primer, TMH Publications , 4th Edition
4. D. Flanagan, 2004, "Java Examples in a Nutshell", O'Reilly. 4th Edition
5. Arnold, Gosling & Holmes, 2005 "The Java Programming Language", Addison-Wesley Professional,4th Edition.

Practical: Object Oriented Programming

Credit: 1

Marks: 25

Programming exercises to be assigned based on:

- 1) Classes and instances
- 2) Working with the java.Math class
- 3) Controls Structures (Loops, conditional statements)
- 4) Array Programming, Inheritance
- 5) Polymorphism, abstract classes and interfaces
- 6) Constructors/ Visibility modifiers
- 7) Utilising the java.util package
- 8) Event handling and GUI (2P)
- 9) Exception handling
- 10) Multithreading
- 11) Collections framework

Paper Title: Data Structures

Paper Code: COM-II.C-4

Marks: 75

Credits: 3

Prerequisite Courses: Introduction to Programming(COM-I.C-2)

Course Objectives:

To understand different methods of organizing data and efficiently implement different data structures.

Learning outcome:

On completion of the course student will learn:

- Different data structures like Stack, Queues, Linked Lists, Graphs and their applications.
- Implementation of data structures.

1. Introduction to data structures: [2L]

Concept, Data type, Data object, ADT, Need of Data Structure, Types of Data Structure

2. Algorithm analysis: [2L]

Algorithm – definition, characteristics, Space complexity, time complexity, Asymptotic notation (Big O)

3. Linked List: [7L]

Introduction to List, Implementation of List – static & dynamic representation, Types of Linked List, Operations on List, Applications of Linked List, polynomial manipulation, Generalized linked list – concept & representation.

4. Stacks: [7L]

Introduction, Representation-static & dynamic, Operations, Application - infix to postfix & prefix, postfix evaluation, Simulating recursion using stack.

5. Queues: [5L]

Introduction, Representation -static & dynamic, Operations, Circular queue, priority queue (with implementation), Concept of doubly ended queue.

6. Trees: [10L]

Concept & Terminologies, Binary tree, binary search tree, Representation – static & dynamic, Operations on BST – create, Insert, delete, traversals (preorder, inorder, postorder), counting leaf, non-leaf & total nodes, non recursive inorder traversal, Expression Tree. Introduction to AVL Trees, Multiway Search Trees, B Tree, B+ Tree.

7. Searching and Sorting : [6L]

use of various data structures for searching and sorting, selection sort, merge sort, quick sort, heap sort and hashing.

8. Graph: [6L]

Concept & terminologies, Graph Representation – Adjacency matrix, adjacency list, Traversals – BFS & DFS, Application of BFS, DFS – Shortest path, Backtracking.

Text Book:

Horowitz Ellis, Sahni Sartaj, 2008, *Fundamentals of Data Structures in C*, University Press, 2nd Edition.

Reference:

1. Langsam Yedidyah, Augenstein J. Moshe, Tenenbaum M. Aaron ,2009, *Data Structures using C*, Pearson Education, 2nd Edition.
2. Gilbeg Richard, Forouzan Behrouz, Data Structures: A Pseudocode Approach with C, Cengage Learning, 2nd Edition.
3. Michael Goodrich, Tamassia Roberto,2001, *Algorithm Design Foundations, Analysis and Internet Examples*. John Wiley and sons.

Practical: Data Structures**Credit:** 1**Marks:** 25

Programs using C language that covers the following concepts:

1. Stack: Static/Dynamic stack implementation.
2. Stack: infix to postfix.
3. Stack: Evaluation of Postfix expression.
4. Queues: Static and Dynamic Queue Implementation
5. Queues: Circular queue
6. List: Singly Linked List
7. List: Doubly Linked List
8. List: Circular Linked List
9. Linked List: Polynomial addition
10. Trees: Binary Search Tree: create, add, delete, display nodes.
11. Trees: BST traversal.
12. Graph: Representation of Graphs, Graph Traversals.
13. Graph: DFS, BFS.
14. Searching
15. Sorting

Semester III

Paper Title: Data Base Management Systems

Paper Code: COM-III.C-5

Marks: 75

Credits: 3

Prerequisites: -

--Nil--

Course Objectives:

It provides basic knowledge of a database management system. It helps to understand importance of ER diagram. It introduces SQL to query a database.

Learning outcome:

- On completion of the course students will learn Database concepts and structures. They will be able to explain terms related to database design and management. Students will understand data modeling and database development process.
- Students will be able to construct and normalize data models and implement the same using any Relational Database Management System.
- Students will become proficient in using database query language, i.e. SQL.

Syllabus

1. Overview of database management

[7L]

Data, information, database, database management system; Managing data; File systems versus a DBMS, advantages of a DBMS; Data abstraction, instances and schemas, data models; Data manipulation language, data definition language; Architecture of a DBMS; Users of a DBMS, database administrator

2. Database design and the ER model

[10L]

Design phases – conceptual design, logical design, physical design; ER model – entities, attributes, and relationships, mapping cardinalities, keys; ER diagrams – strong entities, weak entities, generalization, specialization, aggregation; Converting ER diagram to relational schemas

3. Relational model

[9L]

Relation, properties of relational model; Entities, integrity constraints, referential integrity constraints; Relational algebra – select, project, cross product, set operations, rename operation; Other relational operations – natural join, outer join

4. SQL

[10L]

Basic structure of SQL query – Create, select, where, from, rename operation; Set operations; Aggregate functions; Group by, having clauses; Nested queries; Views; Insert, delete, update.

5. Functional dependency and normalization

[6L]

Atomic domain, nested relation; Key, super key, primary key, candidate key; Functional dependency, axioms, closure of a set of attributes, closure of a set of functional dependencies; Purpose of normalization; 1NF; 2NF; 3NF; BCNF

6. Introduction to Transactions

[3L]

Transaction concept, Transaction state, ACID properties, Concurrent Transactions, Serializability.

Text Book:

1. A Silberschatz, H F Korth, S Sudarshan, *Database system concepts*, McGraw-Hill ,sixth Edition

Reference Books:

- Ramakrishnan, J Gehrke, “*Database management systems*”, McGraw-Hill , 3rd edition
R Elmasri, S B Navathe, “*Fundamentals of database Systems*”, Pearson Education , 5th Edition

Lab : Database Management Systems

Credit : 1

Marks : 25

List of Practicals

1. ER diagram
2. ER diagram with specialization/generalization and aggregation.
3. Converting ER diagram to Schemas
4. Converting ER diagram with generalization/specialization, aggregation into schema
5. Studying RDBMS
 - a. Understanding Client server architecture
 - b. Creating tables
6. SQL
7. SQL
8. Introduction of .NET Framework, Advantages of .Net Framework, Components of .NET Framework, Data type & Operators with examples.
8. Loops, Control Statements, Operators, Data Types
9. Controls : Label, Button, Textbox, Picture Box
10. Controls : Radio button, Checkbox, Timer Control, Scroll Bars
11. Controls : List box, Combo Box, Main Menu Dropdown list etc.
12. Working on Database
13. Working on Database

- 14. Normalization
- 15. Report Writing

Paper Title: Software Engineering

Paper Code: COM-III.E-I

Marks: 75

Credits: 3

Prerequisites: Nil

Course Objectives:

On completion of the course the students will develop specific skills and competencies to use various software engineering tools and methods to develop software for medical, industrial, military, communications, aerospace, business, scientific, and general computing applications.

Learning Outcomes:

- Gain knowledge of concepts & principles, methods and tools used in software engineering .
- Appreciate the role of software engineering in the software development industry.
- Be enabled to use various software engineering methods and tools employed during analysis, design, programming, testing and project management

Syllabus

SOFTWARE PROCESS:

[9L]

Introduction- What is software, Software myths from managers, ‘users and developers’ perspective, Software characteristics, Why engineering approach, definition(s) of software engineering.

Characteristics of software process.

Software Development Processes and Methodologies: waterfall, prototyping, iterative, spiral, unified process.

Benefits of iterative and incremental approach with emphasis on Unified process, CASE Tools, Agile methodologies.

PROJECT MANAGEMENT:

[8L]

Planning a Software Project – Cost estimation, Project Scheduling, Software configuration management plans, Quality Assurance plans, Project Monitoring plans and Risk Management. Techniques such as Interviewing, Requirement Workshop, brainstorming, prototyping. Characteristics of SRS

Design: [9L]

Design Principles.

Design Concepts: Abstraction, Refinement, Modularity, Software architecture, Control Hierarchy, Structural Partitioning, Data Structure, Software procedure, Information hiding.

Effective modular design: Functional Independence, Cohesion, Coupling.

Design heuristics for effective modularity.

Design Specification outline.

Design methods: Data design – Principles, Architectural design using transform flow, transaction flow.

Interface Design- Internal and External Interface design, User Interface Design.

Interface design guidelines.

Procedural design.

CODING: [5L]

Coding styles, standards, peer reviews, checklist, Halstead Metrics

TESTING: [5L]

Testing Fundamental, Functional Testing, Structural Testing, Testing Process and Metrics.

DOCUMENTATION and MAINTENANCE: [4L]

Need for Software Documentation. Types of documentation

Need for Maintenance; Types of Maintenance

REENGINEERING: [5L]

Business Process Reengineering, Software Reengineering, Reverse Engineering, Restructuring, Forward Engineering, The Economics of Reengineering

Text Book:

- Roger Pressman, Software Engineering: A Practitioners Approach, (6th Edition), McGraw Hill, 1997.

Reference Books :

1. An Integrated Approach to Software Engineering by Pankaj Jalote. Narosa Publishing House, 2nd Edition
2. Glenford J.Myers, “ The Art of Software Testing “, John Wiley & Sons, 1979.
3. Sommerville, Software Engineering, 7 th edition, Adison Wesley, 1996.

4. Thomas T. Barker, "Writing s/w documentation - a task oriented approach", Allyn & Bacon Series of Technical Communication , 1998.
5. Code Complete by Steve Mc Connell, published by Microsoft Press, ISBN 978-0-7356-1967-8 Second Edition (June 2004)

Laboratory: Software Engineering

Credit: 01

Marks: 25

List of suggested **PRACTICALS** (the numbers in brackets indicate number of practicals) :

- 1) Requirements Gathering Techniques[2]
- 2) SRS using IEEE format[2]
- 3) Cost and Effort Estimation [4]
- 4) I/O design [2]
- 5) Test Case Design [2]
- 6) Draw a Gantt Chart for a project[2]
- 7) Develop a mini project/ Case Study[1]

Paper Title: Digital Logic Design

Paper Code: COM-III. E-2

Marks: 75

Credits: 3

Course objectives:

To understand the basics of Boolean algebra and the operation of logic components, combinational, sequential circuits and design of digital systems.

Learning Outcome:

- Apply the principles of Boolean algebra to manipulate and minimize logic expressions.
- Use K-maps to minimize and optimize two-level logic functions.
- Understand the operation of latches, flip-flops, counters, registers, and register transfers.
- Analyze the operation of sequential circuits built with various flip-flops.
- Apply the above concepts to develop digital systems.

Syllabus

1. **Introduction to Number Systems and codes:** [3L]
Decimal, Binary, Hexadecimal, and Octal number systems; BCD Code, Gray Code, Excess-3 Code, ASCII code, Unicode.
2. **Boolean Algebra:** [8L]
Basic Boolean functions, Postulates and theorems of Boolean Algebra, logic gates, Sum-of-Products and Product-of-Sums forms of Boolean functions; Canonical and Standard forms. Simplification of Boolean Functions, Plotting of K-Maps, POS and SOP Simplification, NAND and NOR implementation.
3. **Combinational Logic:** [8L]
Design procedure for combinational logic circuits; design and analysis of Half Adder, Full Adder; their use in designing other combinational logic circuits; Analysis & Design of Encoders and Decoders; Multiplexer and demultiplexers; their use in designing combinational circuits.
4. **Sequential Circuits:** [9L]
SR, JK, T, D Flip-flops and Latches, their schematic symbols, Truth tables. Shift Registers - SISO, SIPO, PISO, PIPO, Bi-directional Shift Registers, Loading methods for Shift Registers.
5. **Sequential Circuit Design:** [8L]
Design procedure for sequential circuits using state diagrams, State Tables; State assignments and State minimization methods; Circuit implementation. Design and analysis of Counters, Synchronous Counters; Modulo Counters, Asynchronous, Ripple and Ring Counters; Application of Counters.
6. **D/A & A/D Converters:** [4L]
Digital to Analog converter – Introduction, Weighted-Resistor & R-2R Ladder, Analog to Digital Converter – Introduction, Successive Approximation method.
7. **Semiconductor memories:** [5L]

Memory organization and operation, expanding memory size, Classification and characteristics of memories, RAM, ROM, Synchronous DRAM (SDRAM), Double Data Rate SDRAM, Synchronous SRAM, DDR and QDR SRAM, Content Addressable Memory.

Text Book:

1. R.P. Jain , “Modern digital electronics” , 3rd edition , 12th reprint TMH Publication, 2007.

Reference Books:

1. D.P. Leach, A.P. Malvino, G.Saha, Digital Principles and Applications, 7th Edition, Mc. Graw Hill (SiE)
2. M. Morris Mano, Digital Logic and Computer Design.
3. Herbert Taub and Donald Schilling, Digital Integrated Electronics, McGraw-Hill.

Lab : Digital Logic Design

Credits : 1

Marks : 25

1. Introduction to digital ICs, specifications, verification of the truth tables of logic gates1.
2. Introduction to digital ICs, specifications, verification of the truth tables of logic gates2.
3. Implementation of the given Boolean function using logic gates in SOP form.
4. Implementation and verification of Decoder using logic gates.
5. Implementation of Multiplexer using logic gates.
6. Implementation and verification of De-multiplexer and Encoder using logic gates.
7. Implementation of Binary Adders.
8. Implementation of 4-bit parallel adder.
9. Verification of state tables of RS, JK, T and D flip-flops.
10. Implementation of Shift Registers using flip-flops.
11. Design and verification of the 4-bit synchronous counter1.
12. Design and verification of the 4-bit synchronous counter2.
13. Design and verification of the 4-bit asynchronous counter1.
14. Design and verification of the 4-bit asynchronous counter2.
15. Design and implementation of combinational logic using MUX and DEMUX ICs.

Paper Title: Mathematical Foundation of Computer Science - II

Paper Code: COM-III.E-3

Credits: 3

Marks: 75

Course Objectives:

The objectives of this paper are to build mathematical foundations in the areas namely graph theory and numerical analysis being closely related to topics of computer science.

Learning Outcomes:

On completion of the course, students will learn the following concepts: graph theory and numerical analysis

Syllabus

Graph Theory

1. Graphs, Subgraphs and Components [3L]

Graphs, subgraphs, and some special graphs; Graph properties; Paths, cycles, and components

2. Trees and Cycles [4L]

Trees; Spanning trees; Algorithms to find MST; Cycles; Generation of Trees and Cycles

3. Connectivity [3L]

Cut vertices, Cut edges and Blocks; Eccentricity sequences and Sets; Connectivity parameters

4. Planarity [4L]

Planar embeddings; Bridges; two characterization theorems

5. Eulerian graphs [3L]

Introduction; Eulerian and traversable graphs; Non-Eulerian graphs

6. Digraphs [4L]

Basic definitions; types of connectedness; Covers and bases; Connectivity; Acyclic digraphs

Numerical Analysis

1. Interpolation with Equal Intervals [6L]

Introduction; Various methods of interpolation; Various methods of curve fitting; Newton's method of forward interpolation formula; Newton's method of backward interpolation formula

2. Interpolation with Unequal Intervals [2L]

Introduction; Lagrange's formula

3. Numerical Integration [5L]

Introduction; General quadrature formula; Trapezoidal rule; Simpson's one-third rule; Simpson's three-eighth rule; Weddle's rule

4. Solutions to algebraic and transcendental equations [5L]

Introductions; Graphical method; Bisection method; Method of false position; Secant method; Newton-Raphson method

Linear Algebra [6L]

Adjoint, inverse of a matrix; Rank; Linear equations; Characteristics roots and vectors

Text Books:

1. K R Parthasarathy, Basic Graph Theory, Tata McGraw-Hill Publishing, 1994
2. B S Goel, S K Mittal, Numerical Analysis, Pragati Prakashan, 13th Edition, 1998
3. S.N. Iyengar, Matrices, Anmol Publications, 2010

Reference Books:

1. J Clark, D A Holton, A First Look at Graph Theory, World Scientific, 1991
2. P N Chatterjee, Numerical Analysis, Rajhans Prakashan Mandir, 3rd Edition, 1996
3. V. Krishnamurthy, Introduction to Linear Algebra, Affiliated East-West Press, First Edition, New Delhi, 1976

Lab : Mathematical Foundation of Computer Science - II

Credit: 1

Marks: 25

List of Experiments

Graph Theory

- 1) Read a graph, and check if it is connected.
- 2) Find the components of a graph.
- 3) Check the existence of cycle in a graph.
- 4) Read a weighted graph. Find the minimum spanning tree
- 5) Identify cut vertices and cut edges in a graph
- 6) Find maximum distance between two vertices in a graph
- 7) Read a digraph, and check if it is connected.
- 8) Find the strongly connected components of a digraph.

Numerical Analysis

1. Find the value of dependent variable using Newton's forward formula for a given value of independent variable.
2. Use Newton's backward formula to estimate a value
3. Estimate a value using Lagrange's formula
4. Integrate a function using Simpson's one-third rule
5. Apply Simpson's three-eighth rule to find the value of integration
6. Find the value of a root using method of false position
7. Estimate root of an equation by secant method
8. Apply Newton-Raphson method to estimate the root of a equation

Linear Algebra

- 1) Find the rank of a matrix
- 2) Find solutions of a system of equations

Paper Title: Web Designing

Paper Code: COM-III.E-4

Marks: 75

Credits : 3

Course objectives:

How to design good user interfaces covering important design principles such as learnability, visibility, error prevention, efficiency and graphic design

Learning Outcomes:

Implementation of user interfaces following design principles and using technologies such as HTML, CSS, JavaScript and JQuery.

Syllabus

Unit I : User Interface – Introduction, its importance, design principles – learnability, visibility, error prevention, efficiency, graphic design. Design Patterns for GUI – View tree, Listener, Widget, Model-View-Controller. Approaches to GUI programming – Procedural, Declarative, Direct Manipulation. Web UI – HTML, Javascript, JQuery.

[6L]

Unit II : Structure and Style with HTML and CSS

HTML

[6L]

Introduction. The development process, basic HTML, formatting and fonts, commenting code, color, hyperlink, lists, tables, images, simple HTML forms, web site structure, Meta tags, Character entities, frames and frame sets.

HTML5

[6L]

Introduction, New Elements, Canvas, SVG, Drag/Drop, Geolocation, Video, Audio, Input types, form elements, form attributes, semantic, web storage, app cache, web workers, SSE

CSS

[5L]

Introduction – Syntax, Id & Class, Backgrounds, Text, Fonts, Links, Lists, Tables. CSS Box Model – Border, Outline, Margin, Padding. Advanced - Grouping/Nesting, Dimension, Display, Positioning, Floating, Align, Pseudo-class, Pseudo-element, Navigation Bar, Image Gallery, Image Opacity, Image Sprites, Media Types, Attribute Selectors.

CSS3

[5L]

Introduction, Borders, Backgrounds, Gradients, Text Effects, Fonts, 2D Transforms, 3D Transforms, Transitions, Animations, Multiple Columns.

Unit 3 : Javascript

[10L]

Introduction - What is JavaScript, Understanding Events, JavaScript Example, External JavaScript. Basic Elements – Comment, Variable, Global Variable, Data Types, Operators, If Statement, Switch, Loop: for and while, Function. JavaScript Objects – objects, Array. Browser Object Model - Browser Objects, Window Object, Document Object – getElementById, getElementsByName, getElementsByTagName, innerHTML property, inner Text property. Validation- form validation, email validation.

Unit 4 : Introducing jQuery

[7L]

JQuery : Introduction - Syntax, Selectors, Events. Effects- Hide/Show, Fade, Slide, Animate, stop(), Callback, Chaining. HTML/CSS- Add, Remove, CSS Classes, css(), Dimensions, slider. Traversing – ancestors, descendants, siblings, filtering.

Reference books:

Elisabeth Robson, Eric Freeman, “Head First HTML and CSS”, O'Reilly

Ivan Bayross, “HTML 5 and CSS 3 Made Simple”, BPB publication

Kogent Learning Solutions Inc., “HTML5 Black Book: Covers CSS3, Javascript, XML, XHTML, Ajax, PHP and JQuery”, Pearson Education.

Steven M. Jacobs, Ben Shneiderman, “Designing the User Interface : Strategies for effective human-computer interaction”, 5th Edition, Pearson Education

Lab : Web Designing

Marks: 25

Credits: 1

List of Assignments: (the numbers in brackets indicate number of practicals) :

- 1) Case studies to review UI designs [2 P]
- 2) Create a HTML page with the following : [3 P]
 - a) title heading paragraph emphasis strong and image elements
 - b) complex HTML table
 - c) simple HTML Form covering major form elements
 - d) Embed Video in an HTML page
- 3) Using CSS do the following : [3 P]
 - a) Create a Navigation bar (with dropdown) with CSS
 - b) Create a CSS Grid
 - c) Create a CSS3 based button
 - d) make an image rounded shape
 - e) Create a CSS based sticky footer
 - f) Create CSS3 Corner Ribbon
 - g) Create CSS3 blurry text effect
 - h) Create CSS3 speech bubble shape
 - i) Create image cross fade with CSS3 transition
 - j) Set style for link hover active and visited states of hyperlink
- 4) Write JavaScript functions to : [4 P]
 - a) accept a string as a parameter and converts the first letter of each word of the string in upper case
 - b) check whether a given credit card number is valid or not.
 - c) check whether a given value is an valid url or not.
 - d) check whether a given email address is valid or not.

- e) print an integer with commas as thousands separators
- f) remove items from a dropdown list.

5)Use JQuery to :

[3 P]

- a) Disable buttons
- b) Make textbox read only
- c) Uncheck check boxes
- d) Confirm again
- e) Sort
- f) Switch rows and columns

A mini project combining all the technologies learnt using a front-end development framework such as bootstrap is recommended.

Semester IV

Paper Title: Computer Architecture and Organization

Paper Code: Com-IV. C-6

Marks: 75

Credits: 3

Objectives:

To have a thorough understanding of the basic structure and operation of a digital computer.

Learning Outcome:

- Understand the CPU organization and the instruction set of 8086 processor.
- Study the hierarchical memory system including cache memories.
- Study the different ways of communicating with I/O devices and standard I/O interfaces.

- Briefly study the different types of parallel processing.
- Design, execute and debug assembly language programs for 8086 processor.

Syllabus

1. **Computer System:** [3L]
Function and structure of a computer, Interconnection of components, Performance of a computer. Computer Architecture – Princeton (Von Neumann) and Harvard architecture.

2. **Processing Unit:** [6L]
Architecture of 8086 processor - Registers, ALU and Control unit, Data path in a CPU. Instruction cycle, Organization of a control unit – Block Diagram of Hardwired and Microprogrammed control unit.

3. **Representation of Instructions:** [10L]
Machine instructions characteristics, Types of operations-data transfer, arithmetic, logical, conversion, I/O, system control, transfer of control; 8086 Instruction Set and Assembly language: Addressing modes-immediate, direct, indirect, register, register indirect, displacement, stack. Instruction formats - instruction length, allocation of bits, variable length instructions, Instruction set architectures - CISC and RISC architectures.

4. **Memory Subsystem:** [10L]

Characteristics of memory system, the memory hierarchy, Semiconductor memories, Types of ROM & RAM, Cache memory unit - Concept of cache memory, Organization of a cache memory unit, Mapping methods, replacement algorithms, write policy, block size.

- 5. Input/Output Subsystem:** [8L]
General block diagram of External device & I/O module, Programmed I/O, Interrupt driven I/O, DMA, I/O channels and I/O processors. I/O interfaces – Serial port, Parallel port, PCI bus, SCSI bus, USB bus, Firewire and Infiniband.
- 6. Parallel Processing:** [8L]
Classifications, Introduction to pipeline processing: Instruction pipeline & Arithmetic pipeline, Introduction to Array & Vector processors, Introduction to Multiprocessors.

Text Book:

1. William Stallings, "Computer Organization and Architecture - Designing for performance", EEE, PHI, 9th Edition.

Reference Books:

1. M. Morris Mano, "Computer System Architecture", Pearson Education, 3rd Edition, 2008
 2. D. A. Patterson and J. L. Hennessy, "Computer Organization and Design – The Hardware/Software Interface", Morgan Kaufmann, 4th Edition.
 3. Douglas V. Hall, "Microprocessors and its Interfacing", McGraw Hill Education (India) Private Limited, 3rd Edition.
-
-

Lab : Computer Architecture and Organization

Credits : 1

Marks : 25

1. Study of Motherboard, Peripherals and the Computer System.
O.S. Installation (Dual Boot):

BIOS; Manage disk partitions: understand MBR-style partitions, (primary, extended, logical); list/create/delete partitions;

Manage logical volumes: create/remove physical volumes, create/delete logical volumes, Boot loader

Installation of drivers; updating software packages

2. DOS Commands, Tools for Computer Management (Disk Management, Disk Cleanup, Defragmentation, Performance Monitor, System Restore etc).

Assembly language programs for 8086 using MASM / compatible assembler or Simulator, either in Windows or Linux.

3. Study of addressing modes.
4. Programs for arithmetic operations1
5. Programs for arithmetic operations2
6. Programs for arithmetic operations3
7. Programs for data transfer operations
8. Programs for logical operations1
9. Programs for logical operations2
10. Programs code conversion1
11. Programs code conversion2
12. Programs on sorting
13. Programs on searching
14. DOS/BIOS – Programming1
15. DOS/BIOS – Programming2

Paper Title : Design and Analysis of Algorithms

Paper Code : COM-IV.E-5

Marks : 75

Credits : 3

Course Objectives:

- To study paradigms and approaches used to analyze and design algorithms and to appreciate the impact of algorithm design in practice.
- To ensure that students understand how the worst-case time complexity of an algorithm is defined, how asymptotic notation is used to provide a rough classification of algorithms and compare with one another, and how there are still some problems for which it is unknown whether there exist an efficient algorithm, and how to design efficient algorithms.

Learning Outcomes:

On completion of the course students will learn the following:

- To explain basic concepts related to the design and analysis of algorithms
- To describe classical algorithms and their complexity.
- To design and analyse selected algorithms.

Syllabus

1. Introduction

[8L]

What is an Algorithm?, Rules for writing Algorithms, Properties of Algorithms, Framework for design and analysis of algorithms(RAM model of computation),Recursive Algorithms, Space and Time Complexity by Tabular method(Performance Analysis).

2. Divide and Conquer

[7L]

Elements of Divide and Conquer Algorithms, QuickSort algorithm, Merge sort analysis, Strassen's algorithm for matrix multiplication, Analysis of Binary Search,The Maximum subarray Problem.

3. Dynamic programming

[8L]

General Method, caching v/s computation, Fibonacci numbers by recursion, Fibonacci numbers by caching, Fibonacci numbers by dynamic programming,Optimal Binary Search Tree,Rod Cutting Problem.

4. Greedy algorithms

[5L]

Elements of greedy strategy, Activity-selection problem, Job sequencing with deadlines.

Knapsack problem.

5. Basic Traversal and Search Technique

[7L]

Techniques for Binary Trees, Techniques for Graphs(Breadth First search and Traversal, Depth First Search and Traversal)

6. Graph Algorithms

[6L]

Elementary graph algorithms- Minimum spanning tree, growing a spanning tree, Kruskal and Prim algorithms.

7. Complexity Classes

[2L]

Introduction to polynomial time algorithms, NP, NP Complete, NP Hard

8. Introduction to Randomisation and approximation.

[2L]

Text books

Thomas H. Cormen, Charles E. Leiserson, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", IEEE, PHI, Third Edition

Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", Galgotia, 2nd Edition

Reference Books

A. Aho, J. Hopcroft, J. Ullman, "The Design and Analysis of Computer Algorithms", Pearson Education, Eighth Edition

Lab : Design and Analysis of Algorithms

Credit : 1

Marks : 25

Lab Assignments are to be done using a Programming Language for the following :

- 1 Program to find GCD of 2 numbers using Iterative approach and Recursive approach
- 2 Program for quickSort
- 3 Program to perform Binary Search using Recursive approach
- 4 Program to generate Fibonacci numbers using Dynamic Programming approach.
- 5 Program to implement Activity Selection Problem.
- 6 Program to implement job sequencing with Deadlines.
- 7 Program to implement Knapsack Problem
- 8 Program to implement Rod Cutting Problem.
- 9 Program to implement Binary Tree.
- 10 Program to implement Optimal Binary Search Tree.
- 11 Program to represent graph using matrix.
- 12 Program to represent graph using Linked List.
- 13 Program to implement BFS/DFS Traversal on graph.
- 14 Program to implement Kruskal's Algorithm
- 15 Program to implement Prim's Algorithm

Paper Title: Data Base Application Development

Paper Code: COM-IV.E-6

Marks: 75

Credits: 3

Prerequisites: - Data Base Management Systems

Course Objectives:

To provide advance database solutions.

Learning outcome:

On completion of the course student will learn:

- Advance SQL
- Concept of Big data.
- Developing a database application.

Syllabus

1. Advanced SQL [15L]

SQL data types and schemas, Integrity constraints, Authorization, Embedded SQL, Dynamic SQL, Triggers, Stored Procedures, views

2. Indexing and Hashing [7L]

Basic concepts, Ordered Indices, Dense and Sparse Indices. B and B+ trees Hashing – Static hashing, Dynamic Hashing, Extendable hashing, Comparison of Ordered Indexing and Hashing.

3. Transaction, Concurrency Control, Recovery System. [10L]

Transaction: Transaction concept, Transaction state, Implementation of Atomicity and Durability, concurrency. Serializability, conflict serializability.

Concurrency Control : Lock-Based Protocol

Recovery System: Failure Classification, Storage structure, Stable storage implementation, Recovery and Atomicity: Log-Based Recovery.

4. Introduction to Big data and NoSQL [13L]

Introduction to the Big Data problem. Current challenges, trends, and applications

Comparison between SQL and NOSQL Databases

Types and examples of NoSQL databases- Column, Document, Key-value, Graph, Multi-model

Introduction to Document type NoSQL database such as MongoDB. - Introduce concepts of collection and documents, Advantages, Data types, Projections, indexing, Sharding .

Text Book:

A Silberschatz, H F Korth, S Sudarshan, *Database system concepts*, McGraw-Hill ,sixth Edition

Reference Books :

Ramakrishan, J Gehrke, “*Database management systems*”, McGraw-Hill , 3rd edition

R Elmasri, S B Navathe, “*Fundamentals of database Systems*”, Pearson Education , 5th Edition

Kristina Chodorow *MongoDB : The Definitive Guide (English)* O’Reilly 2nd Edition

Lab : Database Application Development

Credits : 1

Marks : 25

Practical should cover following concepts

1. SQL Revision
2. Advance SQL- Dynamic SQL, Triggers
3. Advance SQL- Stored Procedures
4. Using ODBC API for insertion of record into database.
5. Using ODBC API for deletion of record.
6. Using ODBC API for modification of data.
7. Using ODBC API for data retrieval.
8. Installing and Creating a document using MongoDB concept
9. Performing Indexing using MongoDB
10. Performing aggregation functions on MongoDB
11. Implementation of Master-Slave approach.
12. Connection of MongoDB using Java
13. Insertion, modification, deletion using MongoDB
14. Data retrieval using MongoDB
15. Sharding using Java and MongoDB.

Paper Title: Server Side Programming

Paper Code: COM-IV.E-7

Marks: 75

Credits : 3

Course Prerequisite:

- **Object Oriented Programming**
- **Software Engineering**

Course Objectives:

- Provide an in depth understanding of object oriented approaches to software development, in particular to the analysis and design phases of the software life cycle.
 - Design and implement basic server-side scripts
 - Create data documents using XML
 - Create and manipulate databases using SQL and server side technologies
- Understand how rich internet applications are implemented using AJAX and XML/JSON.

Learning Outcomes:

- Understand the basic underlying concepts in World Wide Web: web server, 3-tier web applications, server side scripting and programming languages, rich internet applications, AJAX and web services
- Understand and apply supporting and emerging web technologies: access to databases, AJAX, rich client user interfaces.

Syllabus

OOAD and UML:

[12L]

OOAD: Definition; object oriented analysis; object oriented design and modelling; Assigning responsibilities.

UML: Main UML diagrams – Class diagram, sequence diagram, activity diagram, use case diagram. Use case model – use case diagram, use case descriptions, use case realization using sequence and activity diagrams. Supplementary requirements. Advanced use case model features.

Requirements: Functional and non-functional

System Design: Class diagram, sequence diagram, activity diagram, state chart diagram, deployment diagram. Brief introduction to other UML diagrams.

Server-side technologies:

[11 L]

Static vs. Dynamic web pages, Need for Server Side technologies, Multitier Web Architecture. Common Gateway Interface standard, server-side includes, server APIs, server-side scripting – working principles, and implicit objects. Database and file access. Comparison of Web servers.

Ajax-Enabled Rich Internet Applications with XML and JSON

[13 L]

AJAX – introduction, purpose, advantages and disadvantages. Key elements of AJAX – introduction to XML. XML processing with server sidescript. XSL, transforms and templates. The XMLHttpRequest object – methods and properties. Creating and using XMLHttpRequest objects. Using XSLT with AJAX.

JSON – Syntax, mixing literals, Array, object, encoding/decoding, JSON versus XML, server-side JSON tools.

Web Services:

[9 L]

Introduction, its role. Basics – Creating, Publishing, Testing and Describing a Web services (WSDL)-Consuming a web service. SOAP - introduction, requests and responses. Role of UDDI – accessing registries. REST based web services – building, deploying and consuming

Reference Books:

1. Martin Fowler, UML Distilled, 3rd Edition, Addison Wesley
2. Booch, Rumbaugh, Jacobson, The Unified Modelling Language User Guide, 2nd edition, Addison Wesley,
3. James Rumbaugh, Object Oriented Modelling and Design, Prentice Hall
4. Dana Moore, Edward Benson, Professional Rich Internet Applications: Ajax And Beyond (English), Wiley India.

5. Schmelzer, XML and Web Services Unleashed, Pearson India.

Lab : Server Side programming

Credit : 1

Marks : 25

List of suggested **PRACTICALS** (the numbers in brackets indicate number of practicals)

- 1) Perform OOAD of a given system using the following diagrams: [3 P]
 - a) use case
 - b) class
 - c) sequence
 - d) activity
- 2) Using server side programming and following OOAD principles develop a dynamic web application. [6 P]
- 3) Add AJAX and Web service(s) to the application. [3 P]

Paper Title: Human Computer Interface

Paper Code: COM-IV.E-8

Marks: 75

Credits: 3

Prerequisites: Nil

Course Objectives:

To study the different aspects of human computer interaction and the computer interface design concepts.

Learning Outcomes

- To understand the intricacies of human interaction with a computer System

- To understand the concept of a graphical user interface, and its design characteristics
- To recognize the human element its strengths and weakness for computer interaction
- To know the principles of good screen design and layouts
- To know the different navigation schemes on windows based interface; learn the different types of selection devices and components of a window based interface
- To know the different types of interaction devices and media

Syllabus

1. Introduction: Human-Computer Interaction, Evaluating Designs, The Birth of HCI. Importance of user Interface, Importance of good design, Benefits of good design, principles & heuristics of good design (4L)
2. Human interaction with computers, Importance of : Human characteristics, Human consideration, Human interaction speeds, Understanding business functions. User centred design- Need-finding: Participant Observation, Interviewing, Additional Need finding, contextual inquiry & persona. (6L)
3. Rapid Prototyping: story boarding. Paper Prototyping and Mockup, Video Prototyping, Creating and Comparing Alternatives (5L)
4. Direct Manipulation and Representations: various user interaction models- command, menu, Direct Manipulation. Mental Models. Heuristics (guidelines) for design. (7L)
5. Graphical Interface Design: Graphical user interface, standards such as Microsoft windows HCI guidelines, Windows: Navigation schemes selection of window; Selection of devices based and screen based controls, Components, Text and messages, Icons, Multimedia, Colors., controls, Help & error messages design. (8L)
6. Web user interface design – jessy James Garette five layers of user experience. (4L)
7. Heuristic Evaluation: Heuristic Evaluation — Why and How? (4L)
8. visualization, Amount of information, Focus and emphasis, Presentation information simply and meaningfully, Information retrieval on web, Statistical graphics (7L)

Text books:

1. Alan Cooper & Robert Reimann, About Face 2.0: The Essentials of Interaction Design, Wiley
2. Alan Dix, Janet Finlay, Gregory D. Abowd, and Russell Beale, Human-Computer Interaction (3rd Edition), Pearson, 2004.
3. Ben Shneiderman and Catherine Plaisant, Designing the User Interface: Strategies for Effective Human-Computer Interaction (5th Edition), 5th ed., Pearson Addison-Wesley, 2009
4. Donald A. Norman, The Design of Everyday Things, Basic Books, 2002

Lab : Human Computer Interface

Credit : 1

Marks : 25

Suggested list of practical (Numbers in brackets indicate number of practicals)

1. Paper Prototyping using templates (1)
2. Conducting survey interview and summarizing the result(1)
3. Persona- conducting contextual interview and developing persona(1)
4. GUI design- form design, menu design, help, error messages(2)
5. Web UI design- pages, navigation, controls, (Ajax) (2)
6. Report designs (2)
7. Visualization and info graphics (1)
8. Heuristic evaluation(2)
9. Story boarding (1)

Semester V

Paper Title: Operating Systems

Paper Code: COM-V.C-7

Marks: 75

Credits: 3

Prerequisite Courses :

- Introduction to Programming(COM-I.C-2)
- Data structures(COM-II.C-4)

Course Objectives:

This course aims at understanding functions of operating system. As part of the course students will study different aspects of operating system such as Memory management, CPU scheduling, Concurrency, Storage management etc.

Learning Outcomes:

On completion of the course

- Students will understand Memory Management
- Students will become familiar with different CPU scheduling algorithms
- They will learn about how concurrent transactions are handled by operating system.
- Students will be able to implement algorithms for CPU scheduling, memory management, Deadlocks etc.

Syllabus :

1. Introduction to Operating System:

[4L]

Basic elements of a computer system: Processor, Main Memory, I/O Modules, System Bus, Instruction Execution; Operating Systems: Definition, Operating system Structure, operating system operations, Relationship between Kernel, OS, and Hardware, Operating system services, System calls, Types of system calls, System programs.

2. Process Management:

[8L]

Process Definition, Process Control Block, Process States, Operations on Process; Interprocess communication, Threads and Microkernels: Definition, Multi-threading Model Process Scheduling, Scheduling Criteria, Scheduling Algorithms, Multi-Processor Scheduling;

3. Process Coordination

[8L]

Process Synchronization, Principles, Mutual Exclusion, The Critical-Section Problem, Petersons Solution, Semaphores, Monitors, Readers/Writers Problem; Classic Problems of Synchronization, Dining Philosopher's problem

Deadlocks- system models, Deadlock characterization , Deadlock Handling Methods, Prevention, Avoidance, Detection, Recovery From Deadlock

4. Memory Management:

[12L]

Introduction, Swapping, Contiguous Memory Allocation, Paging, Page Table, Segmentation

Virtual Memory: Introduction, Demand Paging, Page Replacement, Allocation of Frames, Thrashing

5. Storage Management

[5L]

File System, Concepts, File Organization and Access Methods, Directory and Disk Structure.

Secondary Storage Structure - Overview, disk structure, Disk attachment, Disk scheduling

6. Protection and Security

[3L]

System Protection :Goals of protection, Principles, domain of protection, Access Matrix, Implementation of Access Matrix.

7. Advanced Concepts:

[5L]

Distributed Operating System, Motivation, types of network based operating systems, Network structure, Network topology, Communication structure,

Mobile Operating System, Multimedia Systems – What is multimedia, compression, Requirements of multimedia kernels.

Text Book:

1. A. Silberchatz, Galvin, Gagne, 2008, Operating System Concepts, Wiley publication 8th Edition.

Reference Book:

William Stallings, Operating Systems: Internals and Design Principles, Prentice Hall, 6th Edition

Lab : Operating Systems

Credit : 1

Marks : 25

Suggested list of Practical

1. Demo/Review of Installing Linux / Windows Operating System, Partitioning and formatting disk, Installing applications device drivers, working with files, mounting file systems, checking system space, creating, modifying and deleting user accounts (1P)
2. Study of Basic commands of Linux. (1P)
3. Shell Programming in Unix/Linux, arithmetic operations, loops (1P)
4. Shell Programming – advanced (1P)
5. Menu Driven Shell scripting (1P)
6. Filters and Pipes in LINUX (1P)
7. Implementation of Inbuilt Linux/UNIX commands like cp, rename etc. (1P)
8. Implementation of CPU scheduling policies (3P)
9. Implementation of Memory allocation techniques: (2P)

10. Implementation of Banker's algorithm. (Resource Allocation Graph)

(1P)

Paper Title: Embedded Systems

Paper Code: COM-V.E-9

Marks: 75

Credits: 3

Prerequisite Courses:

- Digital Logic Design(COM-III.E-2)
- Knowledge of Programming

Course Objectives:

- To have a thorough understanding of Embedded Systems and their applications.

Learning Outcome:

After completion of the course students will be able to:

- Gain knowledge about the world of Embedded Systems, their characteristics and applications.
- Understand the function and use of Embedded System hardware and Interfacing I/O devices.
- Identify various sensors, actuators and their use.
- Understand the need for Real Time Operating Systems
- Develop Embedded Applications.

Syllabus :

1. Introduction:

[6L]

Introduction to Embedded Systems, Microprocessors and Micro-controllers. Components of Embedded System & its Classification, Characteristic of embedded system.

Introduction to embedded processor, Digital Signal Processor, Application Specific System Processor, Design Process in Embedded System, Design metrics, Steps in design process. Challenges in Embedded System design, Design Examples. Advances in Embedded Systems.

2. System hardware:

[10L]

System hardware, Interrupt structure and Applications, ARM Processor - Architecture, Programmer's model, Modes of operation, Interrupt, Handling Interrupts, Comparison of ARM7 & ARM9.

Components of Embedded Systems-Management of Power Supply, Clocking Unit, Real Time Clock and Timers, Reset Circuitry and Watchdog Timer. Processor and Memory Selection, Memory Map of Embedded System, Interfacing Processors, Memories and I/O – Analog vs Digital. Overview of Arduino, Intel Edison and Raspberry Pi boards.

3. Sensors and Actuators (Overview):

[12L]

Sensors / Transducers: Principles – Classification – Parameters – Characteristics – Environmental Parameters (EP) – Characterization.

Thermal Sensors: Introduction – Gas thermometric Sensors – Thermal Expansion Type Thermometric Sensors.

Magnetic sensors: Introduction – Sensors and the Principles Behind – Magneto-resistive Sensors, Semiconductor Magnetoresistors.

Smart Sensors: Introduction – Primary Sensors – Excitation – Amplification – Filters – Converters – Compensation – Information Coding/Processing - Data Communication – Standards for Smart Sensor Interface – The Automation.

Sensors Applications: Introduction – On-board Automobile Sensors (Automotive Sensors) – Home Appliance Sensors – Aerospace Sensors — Sensors for Manufacturing – Sensors for environmental Monitoring.

Actuators: Overview of Pneumatic and Hydraulic Actuation Systems, Mechanical Actuation Systems, Electrical Actuation Systems

4. I/O Interfacing and Communication:

[10L]

I/O interfacing and Communication Buses, Serial vs Parallel Communication, Serial Data Communication RS-232/UART.

I/O devices, ADC / DAC, Optical Devices such as LED / LCD Display devices, Opto-Isolator, Relays, DC motor, Stepper motor, Timers/Counters, Parallel ports - Device interfacing.

Serial communication Protocols - UART Protocols, I²C, CAN, USB & ZigBee – Protocol Architecture, Topology, Packets, Communication Cycle, Arbitration, Applications and comparison.

5. Internet of Things (IoT):

[3L]

Introduction to IoT, **M2M to IoT**-The Vision-Introduction, M2M towards IoT- the global context, **IoT Architectural Overview, Potential and Challenges.**

6. Real Time Operating System:

[4L]

Introduction to RTOS, architecture of kernel, task and task scheduler, interrupt service routines in RTOS Environment.

Text Books

1. Rajkamal, “Embedded Systems – Architecture, Programming and Design”, Tata McGraw Hill, Second Edition, 2008.
2. D. Patranabis, “Sensors and Actuators”, 2nd Ed., PHI, 2013.

Reference Book:

1. Dr. K. V. K. K. Prasad, “Embedded / Real Time System : Concepts, Design, & Programming – Black Book”, Dreamtech Press Publication.
2. David E Simon, “An Embedded Software Primer”, Pearson India, 1st Edition.
3. Tammy Noergaard, “Embedded Systems Architecture”, Elsevier
4. D. Patranabis, “Sensors and Transducers”, PHI Learning Private Limited.
5. Dr. K. V. K. K. Prasad, Gupta Dass, Verma, “Programming for Embedded system”, Wiley – Dreamtech India Pvt. Ltd.
6. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, “From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence”, 1st Edition, Academic Press, 2014.

Lab : Embedded Systems

Credit: 1

Marks: 25

Programs to be executed on some of the embedded boards like Arduino, Intel Edison, Raspberry Pi, Bolt, etc., that covers the following tasks:

- | | |
|-------------------------------------|------|
| 1) Interfacing sensors | (3P) |
| 2) Interfacing output devices | (1P) |
| 3) Interfacing input devices | (1P) |
| 4) Interfacing actuators | (1P) |
| 5) Building obstacle avoiding Robot | (1P) |
| 6) Line Following Robot | (1P) |
| 7) Programming with Raspberry Pi | (2P) |

- 8) Monitoring Data over Cloud (1P)
- 9) Building Web app to control devices. (1P)
- 10) Mini Project

Paper Title: Mobile Application Development

Paper Code: COM-V.E-10

Marks: 75

Credits: 3

Prerequisite Courses :

- Object Oriented Programming(COM-II.C-3)
- Web Designing(COM-III.E-4)

Course Objective:

Students learn how to develop applications for mobile devices, including smart phones and tablets. Students are also introduced to the current mobile platforms, mobile application development environments and mobile device input methods. Students will design and build a variety of apps on a popular platform throughout the course to reinforce learning and to develop real competency.

Learning Outcome :

Upon successful completion of the course, the student will demonstrate the ability to:

- 8) Explain mobile devices, including their capabilities and limitations.
- 9) Review current mobile platforms and their architectures.
- 10) Develop mobile applications on a popular mobile platform.
- 11) Evaluate development with another mobile platform.

Syllabus :

Introduction to mobile devices

(3 L)

Mobile devices vs. desktop devices, Why we Need Mobile App, Different Kinds of Mobile Apps, ARM and intel architectures, Power Management, Screen resolution, Touch interfaces, Application

deployment - App Store, Google Play, Windows Store, Development environments – Android Studio, PhoneGAP, Native vs. web applications.

Review of HTML5/JS/CSS3 (2 L)

Quick recap of technologies, Mobile-specific enhancements, Browser-detection, Touch interfaces, Geolocation, Screen orientation, Mobile browser “interpretations” (Chrome/IE).

Mobile OS Architectures (3 L)

Comparing and Contrasting architectures of Android, iOS and Windows, Underlying OS(Darwin vs. Linux vs. Windows), Kernel structure and native level programming, Runtime (Objective-C vs. Dalvik vs. WinRT), Approaches to power management, Security.

Android overview (2 L)

Introduction to Android.Overview of android stack, Introduction to OS layers, Android features. Linux Kernel, Libraries, Android Runtime, Application Framework, Dalvik VM

Android Components – Introduction (3 L)

Activities, Services, Broadcast Receivers, Content Providers.

Building UI with Activities (4 L)

Activities, Views, layouts and Common UI components, Creating UI through code and XML, Activity life cycle, Intents, Communicating data among Activities.

Advanced UI (5 L)

Selection components (GridView, ListView, Spinner), Adapters, Custom Adapters, Menus,

Toast, Custom Toast, Dialogs, Status bar Notifications.

Multithreading (4 L)

Using Java Multithreading classes, AsyncTask, Handler, Post.

Intent, Intent Filters and Broadcast Receivers (4 L)

Role of filters, Intent-matching rules, Filters in your manifest, Filters in dynamic Broadcast Receivers, Creating Broadcast receiver, Receiving System Broadcast, Understanding Broadcast action, category and data, Sending Broadcast.

Data Storage (5 L)

Shared Preferences, Android File System, Internal storage, External storage. SQLite Introducing SQLite, SQLiteOpenHelper and creating a database, Opening and closing a database, Working with cursors, inserts, updates, and deletes.

Content Providers (5 L)

Accessing built in content providers, Content provider MIME types, Searching for content, Adding, changing, and removing content, Creating content provider, Working with content files.

Services (5 L)

Overview of services in Android, Implementing a Service, Service lifecycle, Inter Process Communication (AIDL Services).

Web Services and WebView - Consuming web services, Receiving HTTP Response (XML, JSON), Parsing JSON and XML, Using WebView.

Reference books:

1. Beginning Android 4 Development, Wei-Ming Lee(John Wiley & Sons)
2. Pro Android 4 ; Satya Komateneni, Dave MacLean (Apress)
3. Hello Android - Introducing Google's Mobile Development platform - Ed Brunette (The Pragmatic Bookshelf)
4. Android Apps with Eclipse 1st Edition, Onur Cinar(Apress)
5. Android- A Programmer'S Guide, Dimarzio, J.F.(Tata McGraw Hill)

Web References:

1. <http://developer.android.com/index.html>
2. <http://www.appinventor.org/>

Lab : Mobile Application Development

Credit: 1

Marks: 25

List of practicals

1. Getting Started with Android – Installing the Development Environment, Configuring Android Stack (1P)
2. Creating the First Android Application - Creating a Simple Android Project, Debugging Application through DDMS. setting up environment. AVD Creation, Executing Project on Android Screen. (1P)
3. Android application development - Use of GUI components to implement a simple application such as a Calculator. (1P)
4. Review the earlier application making use of the advanced UI components. (1P)
5. Implementing Data storage application - an application to make Insert , update , Delete and retrieve operation on the database. (2P)

6. Understanding content providers and permissions: Read phonebook contacts using content providers and display them suitably. (1P)
7. Optimizing your app performance with Services/Multithreading/Multiprocessing (2P)
8. Course Project (3P)

Paper Title: Introduction to Data Science
Paper Code: COM-V.E-11
Marks: 75
Credits: 3

Prerequisite courses:

Students are expected to have basic knowledge of algorithms and reasonable programming experience, and some familiarity with basic linear algebra and basic probability and statistics.

Course Objectives:

Become familiar with methods of data science and their practical usefulness.

Learning outcomes:

At the conclusion of the course, students should be able to:

- Describe what Data Science is and the skill sets needed to be a data scientist.
- Explain in basic terms what Statistical Inference means. Identify probability distributions commonly used as foundations for statistical modeling. Fit a model to data.
- Use R to carry out basic statistical modeling and analysis.

Syllabus :

What is Data Science?,Big Data and Data Science hype -and getting past the hype,
Why now? –Datafication,Current landscape of perspectives,Skill sets needed

Statistical Inference [6L]

Populations and samples,Statistical modeling, probability distributions, fitting a model,
Intro to R

Exploratory Data Analysis and the Data Science Process [6L]

Basic tools (plots, graphs and summary statistics) of EDA,Philosophy of EDA,
The Data Science Process, Case Study: RealDirect (online real estate firm)

Three Basic Machine Learning Algorithms [6L]

Linear Regression,k-Nearest Neighbors (k-NN),k-means

Feature Generation and Feature Selection (Extracting Meaning From Data) [6L]

Motivating application: user (customer) retention,Feature Generation (brainstorming, role of domain
expertise, and place for imagination) , Feature Selection algorithms ,Filters; Wrappers; Decision
Trees; Random Forests

Mining Social-Network Graphs [6L]

Social networks as graphs,Clustering of graphs,Direct discovery of communities in graphs,
Partitioning of graphs, Neighborhood properties in graphs

Data Visualization [6L]

Basic principles, ideas and tools for data visualization,Examples of inspiring (industry) projects,
Exercise: create your own visualization of a complex dataset

Data Science and Ethical Issues [5L]

Discussions on privacy, security, ethics, A look back at Data Science,Next-generation data scientists

Text Book:

Cathy O’Neil and Rachel Schutt. Doing Data Science, Straight Talk From The Frontline,
O’Reilly, 2014.

References Books :

- Jure Leskovek, Anand Rajaraman and Jeffrey Ullman, Mining of Massive Datasets v2.1,

Cambridge University Press, 2014 (free online)

- Kevin P. Murphy, Machine Learning: A Probabilistic Perspective, ISBN 0262018020, 2013.
- Foster Provost and Tom Fawcett, Data Science for Business: What You Need to Know about Data Mining and Data-analytic Thinking, ISBN 1449361323, 2013.
- Trevor Hastie, Robert Tibshirani and Jerome Friedman. Elements of Statistical Learning, Second Edition, ISBN 0387952845, 2009 (free online)
- Avrim Blum, John Hopcroft and Ravindran Kannan. Foundations of Data Science.
(Note: this is a book currently being written by the three authors. The authors have made the first draft of their notes for the book available online. The material is intended for a modern theoretical course in computer science.)
- Mohammed J. Zaki and Wagner Miera Jr, Data Mining and Analysis: Fundamental Concepts and Algorithms, Cambridge University Press, 2014.
- Jiawei Han, Micheline Kamber and Jian Pei, Data Mining: Concepts and Techniques, Third Edition, ISBN 0123814790, 2011.

Lab : Introduction to Data Science

Credit: 1

Marks: 25

List of practicals

- | | |
|---|------|
| 1: Implementation of probability distribution | [1P] |
| 2: Sampling and re-sampling. | [1P] |
| 3: Linear Models | [1P] |
| 4: K-Nearest neighbour | [2P] |
| 5: K-Means | [1P] |
| 6: Feature Selection Algorithm | [2P] |
| 7: Filters and Wrappers | [1P] |
| 8: Decision Trees | [1P] |

All the experiments will be implemented using Excel /R-Tool/ or equivalent.

Paper Title: Software Testing

Paper Code: COM-V.E-12

Marks: 75

Credits: 3

Prerequisite courses: Introduction to Programming (COM-I.C-2)

Course Objectives:

- To study fundamental concepts in software testing, including software testing objectives, process, criteria, strategies, and methods.
- To discuss various software testing issues and solutions in software unit test; integration, regression, and system testing.
- To learn how to plan a test project, design test cases and data, conduct testing operations, manage software problems and defects, generate a testing report.

Learning Outcomes:

On Completion of this course the student will:

- Have an ability to understand and identify various software testing problems, and solve these problems by designing and selecting software test models, criteria, strategies, and methods.
- Have an ability to use software testing methods and modern software testing tools for their testing projects.

Syllabus :

Software testing principles - Software Testing- Need for testing, Psychology of testing, Testing economics, SDLC and Testing, Verification & Validation. Quality Assurance, Quality Control

(3L)

Testing strategies and types - White box testing techniques - Statement coverage, Branch Coverage , Condition coverage, Decision/Condition coverage , Multiple condition coverage, Dataflow coverage, Automated code coverage analysis, Inspections, Walkthroughs Code Review

(5L)

Black box testing techniques - Boundary value analysis,Robustness testing ,Equivalence partitioning,Syntax testing,Finite state testing,Levels of testing,Unit, Integration and System Testing,Compatibility Testing,Domain Testing,Adhoc Testing ,Use of Requirement,Traceability Matrix (6L)

Integration Testing Waterfall - Top-down ,Bottom up ,Big bang,Sandwich (3L)

System and Performance Testing - Types of system testing ,Functional and non-functional testing Acceptance Testing ,Setting entry and exit criteria for phases and typical product release scenarios ,Basic factors governing performance testing, Methodology for performance testing ,Tools for performance testing (4L)

Regression Testing - Purpose ,Timing, Choice of tests ,Smoke tests ,Best practices (3L)

Internationalization and Localization testing - Preliminary concepts,Adhoc testing,Pair testing, Extreme testing, Agile testing, Exploratory testing,Defect seeding (3L)

Usability Testing - Factors in usability testing ,Aesthetics testing ,Accessibility testing ,Tools for usability testing (3L)

Testing object oriented software - Definitions and Challenge differences from testing non-OO Software,Class testing strategies Class Modality,State-based Testing,Message SequenceSpecification (4L)

People and organizational issues in testing - Common people issues and myths in testing, Providing career paths in testing,Organizational structures for testing teams,Geographically distributed testing teams and success factors (6L)

Test Management and Automation- Test Planning,Test Management,Test Process,Test Reporting,Test Automation,Factors to consider in automation,Challenges in test automation,Test Metrics,Product Metrics,Process Metrics,Progress Metrics
Use of metrics in ascertaining product release **(5L)**

References:

1. Software Testing- Principles and Practices ,Srinivasan Desikan and Gopalaswamy Ramesh , Pearson Publication
2. Integrated Approach to Software Engineering , Pankaj Jalote, Narosa Publishing House
3. Software Engineering – A Practitioners Approach, Roger Pressman, McGraw Hill Publication

Lab: Software Testing

Credit: 01

Marks: 25

List of suggested **PRACTICALS** using any testing tool such as Selenium or equivalent :

1. Planning Test Cases (1P)
2. Generating Test Cases/Test Suite (2P)
3. Enhancing Tests (3P)
4. Debugging Tests (2P)
5. Running Tests (2P)
6. Analyzing Results (1P)
7. Reporting Defects (1P)

Semester VI

Paper Title: Computer Networks

Paper Code: COM-VI. C-8

Marks: 75

Credits: 3

Prerequisite Courses :

- Introduction to Programming (COM-I.C-2)
- Object Oriented Programming (COM-II.C-3)

Course Objectives:

- To understand the basic concepts of Computer Networking
- Be familiar with the components required to build and design different types of networks.

Learning outcome:

- Gain Knowledge of the Reference models
- Understand basic concepts of data transmission over wired medium Compare various routing, transport protocols and Identify suitable protocol for a given network.
- Able to design the basic Computer network and maintain the networks
- Develop client server programs for different applications

Syllabus :

1. Introduction

[8L]

Basics of Computer Networks, Classification: transmission technology, scale; Applications; Data Communications: data, signal, bandwidth, bit interval and bit rate, Modes of Communication.

Layered network architecture, Networks models: OSI model, TCP / IP protocol suite; Guided and Unguided Transmission media, Multiplexing: FDM, TDM. Switching: Circuit switching, message switching, Packet Switching.

2. Data link layer

[12L]

Data link control: Framing: Character Count, Character Stuffing, Bit Stuffing; , Error Detection and correction, Flow and error control, HDLC; Multiple access: Random access – Controlled access , ALHOA, CSMA, CSMA/CD and CSMA/CA; Ethernet : IEEE standards, standard Ethernet, Fast

Ethernet, Gigabit Ethernet; Connecting devices: repeater/hub, bridge, router and gateway,
Backbone networks - Virtual LANS

3. Network layer **[14L]**

Functions of Network layer; Network Service types: Virtual Circuits, Datagrams; Logical addressing: IPv4, private and public IP addressing, special IP addresses, subnetting, IPV6 addressing
Internet Protocol: Internetworking: IPv4, Fragmentation and reassembly , Address mapping : ARP, RARP, BOOTP, DHCP, ICMP . Routing: classification of routing, Shortest path routing, Distance Vector routing, Link State routing;

4. Transport layer and Application layer **[9L]**

Process-to-Process delivery: User Datagram Protocol (UDP), Transmission Control Protocol (TCP), Quality of services (QoS); Application Layer: Domain Name System (DNS) , E-mail, FTP, HTTP.

5. Wireless Networks **[2L]**

Basics of wireless networking.

TEXT BOOK:

1. Andrew S. Tanenbaum, David J. Wetherall “Computer Networks”, Prentice-Hall, 5th Edition.

REFERENCES:

1. Behrouz A. Forouzan, “Data communication and Networking”, Tata McGraw – Hill, 2011, 4th Edition.
2. James F. Kurose, Keith W. Ross, “Computer Networking – A Top-Down Approach Featuring the Internet”, Pearson Education, 2009, 5th Edition,
3. Nader. F. Mir, “Computer and Communication Networks”, Pearson Prentice Hall Publishers, 2010.
4. Ying-Dar Lin, Ren-Hung Hwang, Fred Baker, “Computer Networks: An Open Source Approach”, Mc Graw Hill Publisher, 2011.
5. Larry L. Peterson, Bruce S. Davie, “Computer Networks: A Systems Approach”, Morgan Kaufmann Publishers, 2011, 5th Edition

Lab : Computer Networks

Credits : 1

Marks : 25

List of Practicals

1. Installing virtual machines, Ethernet cabling
2. Study of network commands ping, ipconfig, netstat, traceroute
3. Setting up of LAN Network
4. IP address manipulation -Extract network id and Host id given netmask
5. Mini Project / Packet capture tool/ packet generator tool
6. UDP Socket programming (c/c++/Java/ perl)
7. TCP Socket programming -I
8. Configuring VLAN (DLink Switch)/ TCP Socket programming –II
9. Configuring routing tables
10. Configuring DHCP server/client
11. Mini Project / Simulation of IP fragmentation
12. Mini Project/Configuring E-Mail/DNS

Paper Title: Network Security

Paper Code: COM-VI. E-13

Marks: 75

Credits: 3

Prerequisite Courses:

Introduction to Programming(COM-I.C-2)/Object Oriented Programming(COM-II.C-3)

Course Objectives:

- To understand the theory and concepts of Network Security

Learning outcome:

- Gain Knowledge of the threats, vulnerabilities and system risks
- Understand cryptography, ciphers and encryption algorithms
- Know about viruses, Trojan horses, worms, program flaws and the defenses against them

Syllabus :

1. Concepts of Security & Classical Encryption Techniques

[8L]

Introduction, The need for security, Security Approaches, Security Attacks, Security Services, Security Mechanisms, A Model for Network Security. Symmetric Cipher Models – Substitution techniques, Transposition techniques, Steganography, Block Cipher Operation, Electronic Code Book, Cipher Block Chaining, Block Cipher Principles, The Data Encryption Standard

2. Public Key Cryptography and Cryptographic Hash Functions [8L]

Introduction To Number Theory, Modular Arithmetic, Prime Numbers, Euler's Totient Function, Principles of Public Key Cryptosystems, The RSA Algorithm, Other Public key cryptosystems, Diffie Hellman Key Exchange.

Applications of Cryptographic Hash Functions, Two Simple Hash Functions, Hash Functions Based on Cipher Block Chaining, MD5 Message Digest Algorithm, Secure Hash Algorithm SHA 512.

3. Message Authentication Codes and Digital Signatures [8L]

Message Authentication Requirements – Message Authentication Functions –Requirements for Security of MACs, MACs Based on Hash Functions, HMAC, MACs Based on Block Ciphers, Data Authentication Algorithm. Digital Signatures, Digital Signature Standard.

4. Key Management & Distribution And User Authentication [8L]

Introduction, Digital Certificate, Private key Management, The PKIX Model, Public key cryptographic standards, XML, PKI and security

5. Program Security [7L]

Flaws, Malicious code: viruses, Trojan horses, worms, Program flaws: buffer overflows, time-of-check to time-of-use flaws, incomplete mediation. Defenses: Software development controls, Testing techniques.

6. Firewall and Virtual Private Network [6L]

Introduction to network security techniques: IP Security, firewalls, virtual private networks.

TEXT BOOKS:

1. William Stallings, —Cryptography and Network Security – Principles and Practicesl, Prentice Hall of India, Fifth Edition
2. Kahate Atul, “Cryptography and Network Security” Tata McGraw-Hill.
3. Charles P. Pfleeger and Shari L. Fleeger, —Security in Computingl. Prentice-Hall.2003, (3rd edition)
4. Menezes A. J., P.C. Van Oorschot and S.A. Vanstone, “Handbook of Applied Cryptography”

Lab : Network Security

Credits : 1

Marks : 25

List of Practicals

1. Implementation of Caesar Cipher
2. Implementation of One-Time Pad
3. Implementation of Playfair Cipher
4. Implementation of Hill Cipher
5. Implementation of Data Encryption Standard Algorithm
6. Implementation of Image Steganography
7. Implementation of RSA Algorithm
8. Implementation of Digital Signatures using RSA Algorithm
9. Design Network protocol analyzer tool to analyse network traffic.
10. Mini Project/ Case Study

Paper Title: Cloud Computing

Paper Code: COM-VI. E-14

Marks: 75

Credits: 3

Prerequisite Courses:

Operating Systems(COM-V.C-7)

Course Objectives:

- To make students understand the key elements of cloud computing.
- To understand the difference between deploying applications on the cloud and the local infrastructure.
- To understand various cloud service models.

Learning Outcomes:

On completion of the course students will be able to:

- Explain the core concepts of the cloud computing paradigm.
- Characterize the different cloud services ie. Infrastructure, Platform and Software as a Service (IaaS, PaaS, SaaS).

Syllabus :

- 1. Overview of Computing Paradigm [7L]**
Recent trends in Computing: Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Cloud Computing.
- 2. Introduction to cloud computing: [7L]**
Cloud Computing definition, History of Cloud Computing, How Cloud Computing Works, Benefits and challenges of cloud computing, Issues for Cloud Computing.
- 3. Cloud Computing Architecture [10L]**
Comparison with traditional computing architecture (client/server), Cloud Computing Service Models, Deployment Models- Public cloud, Private cloud, Hybrid cloud and Community cloud, Key drivers to adopting cloud, Impact of cloud on users, Governance in the cloud.
- 4. Infrastructure as a Service(IaaS) [7L]**
Introduction to IaaS: IaaS definition, Introduction to virtualization, Different approaches to virtualization, Hypervisors, Machine Image, Virtual Machine(VM). Resource Virtualization: Server, Storage, Network. Examples: Amazon EC2.

5. Platform as a Service(PaaS) [7L]
Introduction to PaaS: What is PaaS, Service Oriented Architecture (SOA).Cloud Platform and Management: Computation, Storage, Examples: Google App Engine, Microsoft Azure, Salesforce.com.

6. Software as a Service(PaaS) [7L]
Introduction to SaaS, Web services, Web 2.0, Web OS, Introduction to MapReduce, Case Study on SaaS.

Text Books:

1. Tim mather, subrakumarswamryandsharhedLatif, “Cloud Computing Security and Privacy”, O’Reilly publication.
2. Richard Hill, Laurie Hirsch, Peter Lake, SiavashMoshiri, “Guide to Cloud Computing Principles and Practices”, Springer, 2013.
3. RajkumarBuyya, James Broberg, Andrzej M. Goscinski, “Cloud Computing: Principles and Paradigms”, Wile,2011.
4. Nikos Antonopoulos, Lee Gillam“Cloud Computing: Principles, Systems and Applications”, Springer, 2012.
5. Ronald L. Krutz, Russell Dean Vines,“Cloud Security: A Comprehensive Guide to Secure Cloud Computing”, Wiley-India, 2010

Lab : Cloud Computing

Credit : 1

Marks : 25

1. Create virtual machines that access different programs on same platform. [1P]
2. Create virtual machines that access different programs on different platforms. [1P]
3. Install a C compiler in the virtual machine and execute a sample program. [1P]
4. Working on tools used in cloud computing online- [2P]
 - a) Storage
 - b) Sharing of data
 - c) Manage your calendar, to-do lists,
 - d) A document editing tool

5. Working with any cloud service to make spreadsheet and notes and collaborate online in real time and chat with other collaborators. [1P]
6. Exploring Public Cloud. [1P]
7. Exploring Cloud IDE's. [1P]
8. Installation and Working of Google App Engine [1P]
9. Web Service deployment and usage over cloud. [2P]

Paper Title: Multimedia Techniques

Paper Code: COM-VI. E-15

Marks: 75

Credits: 3

Prerequisite Courses: Nil

Course Objectives:

On completion of the course the students will develop specific skills and competencies by making them proficient in Designing Graphical Images, Audio and Video Capture and Editing using Software tools

Learning Outcomes:

On Completion of this course the student will:

- Study Multimedia Concepts
- Develop their Creativity and publish a self-contained multimedia Application using multimedia authoring tool in various application areas.

Syllabus :

1. Introduction to Multimedia:

[8L]

Commonly used terms associated with multimedia like CDROM, Storyboard, Script and Authoring tools.

Stages of a Multimedia Project-Planning and Costing, Designing and Producing, Testing and Delivering.

The Multimedia team and their roles- Project Manager, Writer, Video specialist, Audio specialist and Multimedia programmer.

Multimedia Software. Multimedia Hardware.

Social & Ethical considerations, Digital Representations & Standards.

2. Introduction to Computer Graphics: [8L]

Vector graphics fundamentals, shapes, transforms and filters,
Bitmapped graphics: resolution, image compression, manipulation, Geometrical transformations

3. Text and Layout: [5L]

Text in graphics, character set, fonts, layout

4. Sound: [8L]

Basic Sound Concepts, Digitising and processing sound, Music, Speech, Compression, formats, MIDI and Digital Audio

5. Color Science and Color Models: [8L]

Human vision, Camera systems, Gamma correction, Color matching, different Color models – RGB, CYMK, Transformations among color model

6. Video: [8L]

Digitising video, streamed video, video standards, compression: mpeg, dv, codec comparison, introduction to Animation: captured, sprite, key frame, web, 3-D. Virtual reality: VRML

Text Book:

1. Nigel Chapman, Jenny Chapman; Digital Multimedia; Wiley India Edition, 2nd Edition
2. Ze-Nian Li & Mark S Drew; Fundamentals of Multimedia; Pearson Education International Edition
3. Vaughan, Tay; Multimedia: Making it Work; Tata McGraw-Hill, 3rd edition
4. Jeffcoate, Judith; Multimedia in Practice, Technology and Applications, PHI

Lab: MULTIMEDIA TECHNIQUES

Credit: 01

Marks: 25

List of suggested **PRACTICALS** (the numbers in brackets indicate number of practicals):

Practical can be done using Proprietary or FOSS for Text, Image, Audio and Video Editing. For example Scribus, GIMP, Audacity, Movie maker, Openshot, etc.

1. Design a Brochure for a given product, give details. Learn about different Image file Formats (2P)
2. Design a Poster with given information and learn about Image compression (2P)
3. Learn to prepare Images for Print, Web and Video. (1P)
4. Edit the Sound file and Learn about Effects and Filters of sound (3P)
5. Record Your voice and learn about Audio Compression. (1P)
6. Record an Audio Program and Learn about streaming an audio content. (1P)
7. Learn about Video editing – Prepare video with rough cut (2P)
8. Prepare Video content with title and special effects. (1P)
9. Record Video content and learn about video compressions. (1P)
10. Prepare Video content for streaming. (1P)

Paper Title: Digital Marketing

Paper Code : COM-VI.E-16

Marks: 75

Credits: 3

Prerequisite Courses: Web Designing(COM-III.E-4)

Course Objectives:

- To study various online Marketing Strategies.
- Analyze and research Internet to improve the quality and marketability of the Websites.

Learning Outcomes:

On completion of the course students will learn the following:

- Optimize the website for various search engines.

- Market the company/product using Search Engine and Social Media.
- Analyze the Web for improving the marketing strategy.

Syllabus :

I. Search Engine Optimisation (SEO): [10L]

Introduction to Online Search; Function of Search Engines Google Page Rank; Introduction to Search Engine Optimisation; Building Accessible Site; Keyword Research and Optimisation; Link Building Strategies; Useful Tools for SEO; The Past, Present and Future of SEO.

II. Search Engine Marketing (SEM): [9L]

Introduction to Internet and Search Engine Marketing; Google Adwords; Adwords Account Structure; Navigating in Google Adwords; Working with Keywords; Creating Ads in Google Adwords; Creating and Managing your First Ad Campaign; Adwords Reporting and Account Performance Reports.

III. Social Media Marketing (SMM): [9L]

Introduction to the World of SMM; Why Social Media?; Getting Started with Social Media; Building Relationships via Facebook, Twitter, LinkedIn, YouTube; Handling Positive and Negative Comments; Social Media Content Base Creation.

IV. Email Marketing: [5L]

Importance of Email marketing; Email Marketing Software's; Subscriber List; Email Marketing Campaign; Newsletters; Measuring the results.

V. WEB Analytics: [9L]

Web Analytics and Intelligence Tools; Basic Metrics Demystified; Introduction to Google Analytics; Goals and Actionable Insights; Data Management; Social Media Analytics; Social Media Goals and KPI's; Tools for Social Media Analytics.

VI. Marketing Automation: [3L]

Introduction to Marketing Automation; Advantages of using Marketing Automation Software; Issues with Marketing Automation.

Text Books:

- Damian Ryan, 2014 “*Understanding Digital Marketing: Marketing Strategies for Engaging the Digital Generation*”, Kogan Page Publisher, 3rd Edition.

Reference Books:

- Calvin Jones and Damian Ryan, 2012 “*The Best Digital Marketing Campaigns in the World:*
- Nick Smith, 2013 , “*Successful SEO and Search Marketing in a Week*”, Teach Yourself Publisher,.
- Lee Odden,2012,“*Optimize: How to Attract and Engage More Customers by Integrating SEO, Social Media, and Content Marketing*”, Wiley Publishing, 1st Edition.
- [Avinash Kaushik](#), 2013, “*Web Analytics 2.0: The Art of Online Accountability & Science of Customer Centricity (Sybex)*”, Wiley Publishing, 2nd edition .

Practical: Digital Marketing

Credit: 1

Marks: 25

1. Using Search Engine Optimization tools (like google & bing search console, hubspot, webceo, google page speed)
(3P)
2. Using Search Engine Marketing tools (like google adwords, google adwords certifications, search, display, remarketing formats, facebook marketing, linkedin advertising)
(3P)
3. Using Social Media Marketing tools (like hootsuite, buffer, sproutsocial, klear, twitonomy, socialmention, google alerts, mention)
(2P)
4. Using Email Marketing tools (like mailchimp, campaign monitor, mailgun, mandrill, phplist, amazon ses)
(2P)
5. Using Web Analytics tools (like google analytics, compete.com, crazyegg, facebook insights, twitter insights)
(3P)