**Proposed Scheme**
**for**

# Bachelor of Technology
## (Computer Sc. & Engg.)

**wef**
**2016-17 batch**

**Department of Computer Sc. & Engineering**
**Guru Jambheshwar University of Sc. & Tech.**
**HISAR**

# B.TECH. (CSE) --- CREDITS

| | Semester | Credits |
|---|---|---|
| 1. | **I** | 25 |
| 2. | **II** | 25 |
| 3. | **III** | 23.5 |
| 4. | **IV** | 23.5 |
| 5. | **V** | 25.5 |
| 6. | **VI** | 25.5 |
| 7. | **VII** | 27.5 |
| 8. | **VIII** | 24.5 |
| **Total Credits** | 200 | |

# B.Tech. (**CSE**) --- **II Year**

| **Semester- 3** | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **S. No.** | **Subject Area** | **Course Code** | **Subject** | **Teaching Schedule L** | **T** | **P** | **Total Hours** | **Credits** | **Duration of Exam (Hours)** |
| 1 | BS-6 | MAT-201-L | Mathematics III | 3 | 1 | - | 4 | 3.5 | 3 |
| 2 | PC-1 | CSE-201-L | Data Structures & Algorithms | 3 | 1 | - | 4 | 3.5 | 3 |
| 3 | PC-2 | CSE-203-L | Discrete Structures | 3 | 1 | - | 4 | 3.5 | 3 |
| 4 | PC-3 | CSE-205-L | Object Oriented Programming using C++ | 3 | 1 | - | 4 | 3.5 | 3 |
| 5 | ES-6 | CSE-207-L | Digital Electronics | 3 | 1 | - | 4 | 3.5 | 3 |
| 6 | ES-7 | EVS-201-L | Environmental Studies | 3 | - | - | 3 | 3 | 3 |
| 7 | PC-1 | CSE-201-P | Data Structures & Algorithms Lab | - | - | 2 | 2 | 1 | 3 |
| 8 | PC-3 | CSE-205-P | OOPS using C++Lab | - | - | 2 | 2 | 1 | 3 |
| 9 | ES-6 | CSE-207-P | Digital Electronics Lab | - | - | 2 | 2 | 1 | 3 |
| 10 | MC-03 (Non-credit) | CSE-209-P | Skills and Innovation Lab. | - | - | 3 | 3 | 2 units | 3 |
| **TOTAL** | | | | 18 | 5 | **6** | **32** | 23.5 Credits | |

# B.Tech. (**CSE) --- II Year**

| | | **Semester- 4** | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Sl. No** | **Subject area** | **Course Code** | **Subject** | **Teaching Schedule L** | **T** | **P** | **Total Hours** | **Credits** | **Duration of Exam (Hours)** |
| 1 | PC-4 | CSE-202-L | Data Base Management Systems | 3 | 1 | - | 4 | 3.5 | 3 |
| 2 | PC-5 | CSE-204-L | Principles of Software Engineering | 3 | 1 | - | 4 | 3.5 | 3 |
| 3 | PC-6 | CSE-206-L | Computer Networks | 3 | 1 | - | 4 | 3.5 | 3 |
| 4 | PC-7 | CSE-208-L | Java Programming | 3 | 1 | - | 4 | 3.5 | 3 |
| 5 | PC-8 | CSE-210-L | Computer Architecture & Organization | 3 | 1 | - | 4 | 3.5 | 3 |
| 6 | HS-4 | HUM-201-L | Fundamentals of Management | 3 | - | - | 3 | 3 | 3 |
| 7 | PC-4 | CSE-202-P | Data Base Management Systems Lab. | - | - | 2 | 2 | 1 | 3 |
| 8 | PC-6 | CSE-206-P | PC & Computer Network Lab | - | - | 2 | 2 | 1 | 3 |
| 9 | PC-7 | CSE-208-P | Java Programming Lab. | - | - | 2 | 2 | 1 | 3 |
| 10 | MC-04 (Non-credit) | PSY-201-L | Personality Development | 2 | 1 | - | 3 | 2 Units | 3 |
| | | **TOTAL** | | **18** | **5** | **6** | **32** | 23.5 Credits | |

# B.Tech. (CSE) --- III Year

| | | Semester- 5 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **S. No.** | **Subject Area** | **Course Code** | **Subject** | **Teaching Schedule L** | **T** | **P** | **Total Hours** | **Credits** | **Duration of Exam (Hours)** |
| 1 | PC-9 | CSE-301-L | Principles of Operating System | 3 | 1 | - | 4 | 3.5 | 3 |
| 2 | ES-8 | CSE-303-L | Microprocessor & Interfacing | 3 | 1 | - | 4 | 3.5 | 3 |
| 3 | PC-10 | CSE-305-L | High Speed Networks Technologies | 3 | 1 | - | 4 | 3.5 | 3 |
| 4 | PC-11 | CSE-307-L | .NET Technologies | 3 | 1 | - | 4 | 3.5 | 3 |
| 5 | PC-12 | CSE-309-L | Web Development | 3 | 1 | - | 4 | 3.5 | 3 |
| 6 | OE-1 | OE-1 | Open Elective-1 | 4 | - | - | 4 | 4 | 3 |
| 7 | PC-9 | CSE-301-P | OS lab | - | - | 2 | 2 | 1 | 3 |
| 8 | ES-8 | CSE-303-P | Microprocessor & Interfacing Lab. | - | - | 2 | 2 | 1 | 3 |
| 9 | PC-11 | CSE-307-P | .NET Lab | - | - | 2 | 2 | 1 | 3 |
| 10 | PC-12 | CSE-309-P | Web Development Lab | - | - | 2 | 2 | 1 | 3 |
| | | **TOTAL** | | 19 | 5 | 8 | 32 | 25.5 Credits | |

# B.Tech. (CSE) --- III Year

| | | Semester- 6 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **S. No.** | **Subject Area** | **Course Code** | **Subject** | **Teaching Schedule L** | **T** | **P** | **Total Hours** | **Credits** | **Duration of Exam (Hours)** |
| 1 | PC-13 | CSE-302-L | Computer Graphics | 3 | 1 | - | 4 | 3.5 | 3 |
| 2 | PC-14 | CSE-304-L | Intelligent Systems | 3 | 1 | - | 4 | 3.5 | 3 |
| 3 | PC-15 | CSE-306-L | Android Programming | 3 | 1 | - | 4 | 3.5 | 3 |
| 4 | PC-16 | CSE-308-L | Analysis & Design of Algorithms | 3 | 1 | - | 4 | 3.5 | 3 |
| 5 | PC-17 | CSE-310-L | Theory of Automata & Computation | 3 | 1 | - | 4 | 3.5 | 3 |
| 6 | OE-2 | OE-2 | Open Elective-2 | 4 | - | - | 4 | 4 | 3 |
| 7 | PC-13 | CSE-302-P | Computer Graphics Lab. | - | - | 2 | 2 | 1 | 3 |
| 8 | PC-14 | CSE-304-P | Intelligent Systems Lab. | - | - | 2 | 2 | 1 | 3 |
| 9 | PC-15 | CSE-306-P | Android Lab. | - | - | 2 | 2 | 1 | 3 |
| 10 | PC-18 | CSE-312-P | Python Lab | - | - | 2 | 2 | 1 | 3 |
| | | **TOTAL** | | 19 | **5** | **8** | **32** | 25.5 Credits | |

# B.Tech. (CSE) --- IV Year

| | | Semester- 7 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Sl. No.** | **Subject Code** | **Course Code** | **Subject** | **Teaching Schedule** | | | **Total Hours** | **Credits** | **Duration of Exam (Hours)** |
| | | | | **L** | **T** | **P** | | | |
| 1 | PC-19 | CSE-401-L | Compiler Design | 3 | 1 | - | 4 | 3.5 | 3 |
| 2 | PC-20 | CSE-403-L | Wireless & Mobile Communication | 3 | 1 | - | 4 | 3.5 | 3 |
| 3 | PC-21 | CSE-405-L | Software Project Management | 3 | 1 | - | 4 | 3.5 | 3 |
| 4 | PC-22 | CSE-407-L | Data Warehousing and data Mining | 3 | 1 | | 4 | 3.5 | 3 |
| 5 | PC-23 | CSE-409-L | Advanced Computer Architecture | 3 | 1 | - | 4 | 3.5 | 3 |
| 6 | OE-3 | OE-3 | Open Elective-3 | 4 | - | | 4 | 4 | 3 |
| 7 | PC-19 | CSE-401-P | Compiler Design Lab. | - | - | 2 | 2 | 1 | 3 |
| 8 | PW-1 | CSE-411-P | Major PROJECT – Part I | - | - | 8 | 8 | 4 | 3 |
| 9 | PW-2 | CSE-413-P | Assessment of Mini-Project based upon .NET / Android / Java done during 3rd Year Summer Vacations | - | - | - | - | 1 | - |
| | | **TOTAL** | | 19 | 5 | 10 | 34 | 27.5 Credits | |

## OPEN ELECTIVE 1, 2 & 3

To be offered by other departments

# B.Tech. (CSE) --- IV Year

| | | Semester- 8 | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **S. No** | **Subject Code** | **Course Code** | **Subject** | **Teaching Schedule L** | **T** | **P** | **Total hours** | **Credits** | **Duration of Exam (Hours)** |
| 1 | PC-24 | CSE-402-L | Distributed Operating System | 3 | 1 | - | 4 | 3.5 | 3 |
| 2 | PC-25 | CSE-404-L | Information & Cyber Security | 3 | 1 | - | 4 | 3.5 | 3 |
| 3 | PC-26 | CSE-406-L | Cloud Computing | 3 | 1 | | 4 | 3.5 | 3 |
| 4 | PE-1 | PE-1 | Professional Elective I | 3 | 1 | | 4 | 3.5 | 3 |
| 5 | PE-2 | PE-2 | Professional Elective II | 3 | 1 | | 4 | 3.5 | 3 |
| 6 | PC-26 | CSE-406-P | Cloud Computing lab | - | - | 2 | 2 | 1 | 3 |
| 7 | PW-3 | CSE-408-P | Major Project - Part II | - | - | 8 | 8 | 4 | 3 |
| 8 | PW-4 | CSE-410-P | General Fitness for the Profession | - | - | - | - | 1 | 3 |
| 9 | PW-5 | CSE-412-P | Colloquium | - | - | 2 | 2 | 1 | 3 |
| | | **TOTAL** | | 15 | **5** | **12** | 32 | 24.5 Credits | |

**OR**

| | | | **Credits** |
|---|---|---|---|
| Industrial Training* | CSE-442-P | Full Semester Industrial Training* | 24.5 |

**PROFESSIONAL ELECTIVES-I**

CSE-414-L Object Oriented Systems Development
CSE-416-L Embedded system Design
CSE-418-L Digital Image Processing
CSE-420-L Network Programming
CSE-422-L Software Testing
CSE-424-L Ubiquitous Computing
CSE-426-L Machine Learning
Any one MOOC Course not studied earlier

**PROFESSIONAL ELECTIVES-II**

CSE-428-L Bioinformatics
CSE-430-L Big data analytics
CSE-432-L Operations Research
CSE-434-L Multimedia technologies
CSE-436-L Natural Language Processing
CSE-438-L Management Information system
Any one MOOC Course not studied earlier

**OR**

***Full Semester Industrial Training**

The student will be required to submit to the department, the offer letter for the full semester industrial training, at-least 15 days before the commencement of 8th semester. The options shall be according to the following conditions:

A student may opt for one semester industrial training in lieu of attending the courses of 8th semester. The credit/marks for industrial training will be equals to the total credits/marks of courses offered in 8th semester study. A student will be allowed to join the industrial training under following conditions:

a. If the student gets selected for the job through campus placements and the employer is willing to take the student for the training for a period of full semester.
b. If the student gets offer of pursuing training from reputed Research organization/Govt. sponsored project/ Govt. research institution/ Multinational corporations (MNCs)/ Public sectors. For pursuing this training, the student shall require prior approval from Dean of Faculty of Engineering & Technology through the Chairperson of the respective department. To ensure the fruitfulness of this training, a list of companies, beside the Govt. organizations/ Public sectors, will be provided. The student will be allowed to go for training only to the companies/organizations mentioned in the list. The list can be modified (addition/deletion) from time to time subject to approval from Dean of Faculty of Engineering and Technology.

# Mathematics-III

| | |
|---|---|
| Course Code: MAT-201-L<br><br>Course Credits: 3.5<br><br>Mode: Lecture(L) and Tutorial(T)<br><br>Type: Compulsory<br><br>Contact Hours: 3 hours (L) + 01 hour (T) per week.<br><br>Examination Duration: 03 hours. | **Course Assessment Methods (Internal: 30; External: 70)** Two minor test each of 20marks, class performance measured through percentage of lecture attended (4 marks), assignments, quiz etc. (6 marks) and end semester examination of 70 marks. \<br><br>For the end semester examination, nine question are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus, it will contain seven short answer type question. Rest of the eight questions is to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the four units. All questions carry equal marks. |

**Prerequisite:** Basic knowledge of calculus, complex analysis and statistics.

**Course outcomes:**

CO1: Problems of Fourier series and Fourier transforms used in engineering applications

CO2: Calculation of improper/ singular integrals with the help of complex analysis

CO3: Statistical tests for system goodness.

CO4: Problems of LPP and their interpretation.

**Unit-I**

Fourier Series and Fourier Transforms: Euler's formulae, conditions for a Fourier expansion, change of interval, Fourier expansion of odd and even functions, Fourier expansion of square wave, rectangular wave, saw-toothed wave, half and full rectified wave, half range sine and cosine series. Fourier integrals, Fourier transforms, Shifting theorem (both on time and frequency axes), Fourier transforms of derivatives, Fourier transforms of integrals, Convolution theorem, Fourier transform of Dirac delta function.

**Unit-II**

Functions of Complex Variable: Definition, Exponential function, Trigonometric and Hyperbolic functions, Logarithmic functions. Limit and Continuity of a function, Differentiability and Analyticity. Cauchy-Riemann equations, necessary and sufficient conditions for a function to be analytic, polar form of the Cauchy-Riemann equations. Harmonic functions. Integration of complex functions. Cauchy Theorem, Cauchy- Integral formula.

**Unit-III**

Power series, radius and circle of convergence, Taylor's Maclaurin's and Laurent's series. Zeroes and singularities of complex functions, Residues. Evaluation of real integrals using residues (around unit and semi circle only).

**Unit-IV**

Probability Distributions and Hypothesis Testing: Expected value of a random variable. Properties and application of Binomial, Poisson and Normal distributions. Testing of a hypothesis, tests of significance for large samples, Student's t-distribution (applications only), Chi-square test of goodness of fit. **Linear Programming:** Linear programming problems formulation, Solving linear programming problems using (i) Simplex method.

**Text books:**

1. Advanced Engg. Mathematics , F Kreyszig.

2. Higher Engg. Mathematics, B.S. Grewal.

**Reference books:**

1. Advance Engg. Mathematics, R.K. Jain, S.R.K. Iyenger.
2. Advanced Engg. Mathematics, Michael D. Greenberg.
3. Operation Research , H.A. Taha.
4. Probability and statistics for Engineers, Johnson. PHI.

## Data Structures & Algorithms

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-201-L/IT-201-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**

Students are expected to be proficient in programming in a standard programming language like C.

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are to:**

1. Familiarize students with basic data structures
2. Learn theoretical analysis, implementation and applications of data structures
3. Make them learn the use of these data structures in fundamental algorithms

**By the end of the course a student is expected to:**

CO1. Study algorithms for various computing problems
CO2. Analyze the time and space complexity of algorithms
CO3. Efficiently implement their solution using programming language C
CO4. Apply algorithms for solving problems based on linear and non- linear data structures.

**Course Contents**

**Unit I**

Data Structures: Definition and its types, Abstract Data Types, Static and dynamic memory storage.

Arrays, matrices, sparse matrices, multi-dimensional arrays, operations on arrays, Linear search, Binary search, Insertion sort, selection sort, Bubble sort, Merge sort.

Linked Lists: List Types (singly, doubly, singly circular, header, doubly circular), Operations on Lists – create, insert, delete, search, display; Applications of linked list

**Unit II**

Stacks: Definition, Array implementation of stacks, Linked implementation of stacks, Applications of Stacks: Infix, Postfix and prefix expression, conversions and evaluation of expressions, Recursion, Quick Sort.
Queues: Definition, Array implementation of queues, Linked implementation of queues, Circular queues, Priority queues, Double-ended queues

**Unit III**

Trees: Binary Trees and their properties, Linked and static representation of binary trees, Complete Binary Tree, Threaded Binary tree, Different tree traversal algorithms(non-recursive), Binary Search Tree (create, delete, search, insert, display), Heap Sort and its complexity analysis, AVL Trees, Balanced multi-way search trees.

**Unit IV**

Graphs: Definition, Array and linked representation of graphs, Graph Traversal (BFS and DFS), Adjacency matrix and adjacency lists, path matrix, Finding Shortest Path - Warshall's Algorithm, Hashing, Hash table, Hash functions.
Running time: Time Complexity, Big – Oh - notation, Best Case, Worst Case, Average Case, Factors depends on running time, Evaluating time Complexity

T**ext and Reference Books:**


1. Theory & Problems of Data Structures, Jr. Symour Lipschetz, Schaum's outline, TMH, 1998
2. Data Structures using 'C', Tenenbaum, Langsam, Augenstein, Pearson Education, 2008.
3. Data Structures Using 'C', Bala Guruswamy, TMH, 2007.
4. Data Structures Using 'C , Weiss, Pearson Education,1997.
5. Data Structures and Algorithms, A.V. Aho, J.E. Hopcroft and T.D. Ullman, Original edition, Addison Wesley, Low Priced Edition, 1999.
6. Data Structures and Program Design in C, D.Robert Kruse,PHI, 1994.

## Discrete Structures

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-203-L/IT-203-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4 Hours/Week<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 Hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**

Basic knowledge of Pre-calculus, Algebra and Trigonometry.

**About the Course and its Objectives & Outcomes:**

The purpose of this course is to understand and use discrete structures that are backbones of computer science. Introduction to Discrete Mathematics is a course designed for students interested in information technology and programming that includes topics in set theory, algebraic structures, Boolean algebra, and graph theory.

**The objectives of this course are to:**

1. Familiarize students with basic logic and set theory.
2. To make students learn core ideas in combinatorial mathematics.
3. To make them learn core ideas in graph theory

By the end of the course a student is expected to:

CO1. be able to use logical notation to define and reason about fundamental mathematical concepts such as sets, relations, functions, and integers.
CO2. be able to Draw and apply Venn diagrams.
CO3. be able to classify types of graphs, find paths, circuits.
CO4. be able to apply graph theory model.

### Course Contents

**Unit I**

**Set Theory:** Introduction to set theory, Set operations, Algebra of sets, Duality, Finite and Infinite sets, Classes of sets, Power Sets, Multi sets, Cartesian Product, Representation of relations, Types of relation, Equivalence relations

and partitions, Partial ordering relations and lattices, Function and its types, Composition of function and relations, Cardinality and inverse relations.

**Unit II**

**Propositional Calculus:** Basic operations: AND(^), OR(v), NOT(~), Truth value of a compound statement, propositions, tautologies, contradictions.
**Algebric Structures:** Definition and examples of a monoid, Semigroup, Groups and rings, Homomorphism, Isomorphism and Automorphism, Subgroups and Normal subgroups, Cyclic groups, Integral domain and fields, Cosets, Lagrange‟s theorem.

**Unit III**

**Recursion and Recurrence Relation:** Polynomials and their evaluation, Sequences, Introduction to AP, GP and AG series, partial fractions, linear recurrence relation with constant coefficients, Homogeneous solutions, Particular solutions, Total solution of a recurrence relation using generating functions.

**Unit IV**

**Graphs and Trees:** Introduction to graphs, Directed and Undirected graphs, Homomorphic and Isomorphic graphs, Subgraphs, Cut points and Bridges, Multigraph and Weighted graph, Paths and circuits, Shortest path in weighted graphs, Eurelian path and circuits, Hamilton paths and circuits, Planar graphs, Euler‟s formula, Trees, Spanning trees, Binary trees and its traversals

T**ext and Reference Books:**

1. Elements of Discrete Mathematics, C.L Liu, McGraw Hill, 1985.
2. Discrete Mathematics, Johnson Bough R., 5th Edition, PEA, 2001.
3. Concrete Mathematics: A Foundation for Computer Science, Ronald Graham, Donald Knuth and Oren Patashik, Addison-Wesley, 1989.
4. Mathematical Structures for Computer Science, Judith L. Gersting, Computer Sc Press, 1993.
5. App Discrete Structures for Computer Science, Doerr and Levasseur, (Chicago:SRA), 1985.
6. Discrete Mathematics ,A. Chtewynd and P. Diggle (Modular Mathematics series), Edward Arnold, London, 1995.
7. Schaums Outline series: Theory and problems of Probability by S. Lipshutz, McGraw-Hill Singapore, 1982.
8. Discrete Mathematical Structures, B. Kolman and R.C. Busby, PHI, 1996.
9. Discrete Mathematical Structures with Applications to Computers by Tembley & Manohar, Mc Graw Hill, 1995.

## Object Oriented Programming Using C++

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-205-L/IT-205-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours:<br><br>Mode: Lecture(L)<br><br>Examination Duration: | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**

Basic knowledge of programing language like C and Data structure is requiring for good understanding of C++

**About the Course and its Objectives & Outcomes:**

C++ is a generic and step-stone for all languages. Students can switch to any other language without any problem.

**The objectives of this course are to:**

1. To get a clear understanding of object oriented programming through C++.
2. To make understanding of functional, logic, and object-oriented programming paradigms
3. Understand design/implementation issues involved with variable allocation and binding, control flow, types, subroutines, generic parameter passing

**By the end of the course a student is expected to:**

CO1. Be able to develop, design and implement simple computer programs.
CO2. Describe the concept of function overloading, operator overloading, virtual functions and polymorphism.
CO3. Understand object-oriented design and programming.
CO4. Understand dynamic memory management techniques using keywords, pointers, constructors, destructors, etc.
CO5. Be able to design, implement, and test relatively large C++ programs.
CO6. Classify inheritance with the concept of early and late binding, exception handling, and generic programming for STL.

**Course Contents**

**Unit I**

**Introduction:** Object Oriented Programming, C++ Standard Library, Basics of a Typical C++ Environment, Illustrative Simple C++ Programs, Pre-processors Directives, Macro Programs, Header Files and Namespaces, library files, new features of ANSI C++ standard, Introduction to VC++, Dev C++ etc.

**OOPs concepts:** Encapsulation (Information Hiding), Access Modifiers: Controlling access to a class, method, or variable (public, protected, private, block level and scope), Other Modifiers.

**Class:** Block structure of Class and Struct in memory, Accessing Members of Structures, Class Scope and Accessing Class Members, Separating Interface from Implementation.

**Initializing Class Objects:** Default Constructors, Chained Constructor, Default Arguments with Constructors, Constant Object and Const Member Functions, Object as Member of Class, Using Destructors.

**Function:** Controlling Access Function and Utility Functions, Function overloading.

**Unit II**

**Classes and Data Abstraction:** Introduction, Inline Function, Friend Function and Friend Classes, Using This Pointer, Dynamic Memory Allocation with New and Delete, Static Class Members, Container And Iterators, algorithm and functional Classes, Proxy Class.

**Polymorphism:** Overloading, Inheritance, Overriding Methods, Abstract Classes, Reusability, Class's Behaviors.

**Inheritance:** Base Classes And Derived Classes, Protected Members, Casting Base- Class Pointers to Derived-Class Pointers, Using Member Functions, Overriding Base –Class Members in a Derived Class, Public, Protected and Private Inheritance, Using Constructors and Destructors in derived Classes, Implicit Derived –Class Object to Base- Class Object Conversion, Composition Vs. Inheritance.

**Unit III**

**Virtual Functions and Polymorphism:** Introduction to Virtual Functions, Abstract Base Classes and Concrete Classes, Polymorphism, New Classes and Dynamic Binding, Virtual Destructors, Polymorphism, Dynamic Binding.

**Files and I/O Streams:** Files and Streams, Creating a Sequential Access File, Reading Data From a Sequential Access File, Updating Sequential Access Files, Random Access Files, Creating a Random Access File, Writing Data Randomly to a Random Access File, Reading Data Sequentially from a Random Access File.

**Unit IV**

**Managing Console I/O:** Stream Input/output Classes and Objects, Stream Output, Stream Input, Unformatted I/O (with read and write), Stream Manipulators, Stream Format States, Stream Error States.

**Exception Handling:** Introduction, Basics of C++ Exception Handling: Try Throw, Catch, Throwing an Exception, Catching an Exception, Rethrowing an Exception, Exception specifications, Processing Unexpected Exceptions, Stack Unwinding, Constructors, Destructors and Exception Handling, Exceptions and Inheritance.

**Generic Classes (Templates):** Introduction, Function Templates, Overloading Template Functions, Class Template, Class Templates and Non-Type Parameters, Templates and Inheritance, Templates and Friends, Templates and Static Members.

T**ext and Reference Books:**


1. C++ How To Program 6th Ed by H M Deitel and P J Deitel, Prentice Hall, 2008.
2. Object-Oriented Programming in C++ 3rd Ed by Robert Lafore, 2001.
3. Programming with C++ 3rd Ed by D Ravichandran, T.M.H, 2011.
4. Object oriented Programming with C++ 6th Ed by E Balagurusamy, Tata McGraw-Hill, 2013.
5. Computing Concepts with C++ Essentials 3rd Ed by Horstmann, John Wiley,2003.
6. The Complete Reference in C++ 5th Ed by Herbert Schildt , TMH, 2012.

## Digital Electronics

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-207-L/IT-207-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 hrs | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:** Basic knowledge of computers and Boolean algebra**.**

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are to:**

1. acquiant the students with Digital techniques of circuit design
2. design counters, multiplexers and other sequential circuits and combinational circuits

**By the end of the course a student is expected to:**

CO1. Design, simulate, built and debug complex combinational and sequential circuits based on an abstract functional specification
CO2. Combinational systems using standard gates and minimization methods such as karnaugh maps.
CO3. Have an understanding of registers and counters.
CO4. Have an understanding of logic families.

**Course Contents**

**Unit I**

**C**ombinational circuits: Boolean algebra – Boolean theorems, minimization of Boolean function, K-Map, minimization using tabular method, Basic logic gates. Boolean functions realization using logic gates, half & full adder, subtractors, coders, decoders, multiplexer, & de-multiplexers & their applications, Comparators, Digital techniques related to PLAs, PALs, ROMs

**Unit II**

Sequential circuits : introduction to sequential logic, concept of history sensitive circuits & feed back , introduction to flip-flop , RS, D, T, JK flip-flops, race around condition, Master slaves , flip-flop clocked sequential circuits.Asynchronous Sequential Logic: Introduction, Analysis Procedure, Circuits with Latches, Design Procedure, Reduciton of state and Flow Tables, Race-Free state Assignment Hazards

**Unit III**

Counter and shift register :Asynchronous & Synchronous counters , design of synchronous circuits, state transition diagram , excitation table for flip – flop , design using minimization techniques ,shift register , basic principal , serial and parallel data transfer , shift left/right register.

**Unit IV**

Logic families, diode switching, transistor as a switching element , circuit concept andcomparison of logic families – TTL, ECL, NMOS & CMOS. Tristate logic open collector outputs, interfacing between logic families digital Techniques related to PLDs, FPGA, GaAs defined circuits, Introduction to BiCMOS circuits

**Text and Reference Books**:


1. Digital Design, Morris Mano, PHI, 2008
2. Digital Electronics, Bignill & Donovan, Delmar publishers, 1989.
3. Digital Integrated Circuit, A.K.Gautam- Katson Publication
4. Digital Integrated Electronics, Taub and Schilling ,TMH, 1977
5. Fundamentals of Digital Computers, Bartee , Thomas C , Tata McGraw-Hill, 2004
6. Introduction To Digital Microelectronic Circuits, Gopalan, K. Gopal,Tata McGraw-Hill, 2002
7. Digital Principles & Applications , Malvino, A.P. & Leach, Donald P. , Tata McGraw-Hill , 2001
8. Digital Electronics Principles & Application, Tokheim, H. Roger L. ,Tata McGraw-Hill ,8th Ed., 2014

## Environmental Studies

**General Course Information:**

| | |
|---|---|
| Course Code: EVS-201-L<br><br>Course Credits: 3.0<br><br>Mode: Lecture (L) and Tutorial (T)<br><br>Type: Compulsory<br><br>Contact Hours: 3 hours (L) + 0 hour (T) per week.<br><br>Examination Duration: 03 hours. | **Course Assessment Methods (Internal: 30; External: 70)** Two minor test each of 20marks, class performance measured through percentage of lecture attended (4 marks), assignments, quiz etc. (6 marks) and end semester examination of 70 marks. \<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus; it will contain seven short answer type questions. Rest of the eight questions is to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the four units. All questions carry equal marks. |

**Prerequisite:** Student should have prior knowledge of basic environment science.

**Objectives:**

- To enhance knowledge skills and attitude to environment.
- To understand natural environment and its relationship with human activities.

**Course outcomes:**

CO-1 Students will be able to enhance and analyze human impacts on the environment.
CO-2 Integrate concepts & methods from multiple discipline and apply to environmental problems.
CO-3 Design and evaluate strategic terminologies and methods for sustainable management of environmental systems.
CO-4 Field studies would provide students first-hand knowledge on various local environment aspects which forms an irreplaceable tool in the entire learning process.

**Unit-I**

Multidisciplinary nature of Environmental studies: Definition, scope and importance, need for public awareness; Concept, Structure and function of an ecosystem: Producers, consumers and decomposers, Energy flow in the ecosystem ,Ecological succession ,Food chains, Food webs and ecological pyramids; Introduction, types, characteristics features, structure and function of Forest ecosystem, Grassland ecosystem ,Desert ecosystem, Aquatic ecosystem (Ponds, Stream, lakes, rivers, oceans, estuaries); Biodiversity: Introduction, Definition: genetic, species and ecosystem diversity, Bio-geographical classification of India, Value of biodiversity: consumptive use, productive use, social ethical, aesthetic and option values; Biodiversity at global, national and local level, India as a mega-diversity nation, Hot-spot of biodiversity, Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts, Endangered and endemic species of India, Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity

**Unit-II**

Renewable and non-renewable resources, Natural resources and associated problems ,Forest resources: Use and over-exploitation, deforestation, case studies, Timber extraction, mining, dams and their effects on forests and tribal people; Water resources: Use and over utilization of surface and ground water, floods, droughts conflicts over water, dams benefits and problems; Mineral resources: Use and exploitation, environmental effects of extracting and mineral resources; Food resources: World food problem, changes caused by agriculture and overgrazing, effects of

modern agriculture, fertilizer-pesticide problems, water logging, salinity; Energy resources: Growing energy needs, renewable and non-renewable energy sources, use of alternate energy sources, case studies; Land resources: Land as a resource, land degradation, main induced landslides, soil erosion and desertification, Role of an individual in conservation of natural resources, Equitable use of resources for suitable lifestyle.

**Unit-III**

Definition of Environment Pollution; Causes, effects and control measures of: Air Pollution, Water Pollution, Soil pollution, Marine pollution, Noise pollution, Thermal pollution, Nuclear hazards; Solid waste Management: Causes effects and control measures of urban and industrial wastes; Role of and individual in prevention of pollution, Pollution case studies; Disaster management: floods, earthquake, cyclone and landslides; Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust, Case studies; different laws related to environment: Environment Protection Act, Air (Prevention and Control of Pollution) Act, Water (Prevention and Control of Pollution) Act, Wildlife Protection Act, Forest Conservation Act.; Issues involved in enforcement of environmental legislation, Public awareness

**Unit-IV**

Social issues and the Environment: From unsustainable to Sustainable development, Urban problems related to energy; Water conservation, rain water harvesting, watershed management; Resettlement and rehabilitation of people; its problem and concern, case studies; Environment ethics: Issues and possible solutions;Wasteland reclamation; Consumerism and waste products; Human Population growth, variation among nation, Population explosion- Family Welfare Programme, Environment and human health , Human Rights, Value Education, HIV/AIDS, Women and Child Welfare, Role of Information Technology in Environment and human health, Case Studies.

**Field Work**: Visit to a local area to document environmental assets- river/forest/grassland/hill/mountain; Study of simple ecosystems – ponds, river, hill slopes etc; Study of common plants, insects, birds; Visit to a local polluted site- Urban/Rural/Industrial/Agricultural.

**TEXT BOOK:**

1. Erach Bharucha , "Environmental Studies for Undergraduate Courses", University Grants Commission and

   Bharati Vidyapeeth Institute of Environment Education and Research, Pune, University press pvt. Ltd. (India)

2. Fundamental concepts in Environmental studies, Dr. D.D. Mishra. S. Chand publications

**REFERENCE BOOKS:**


**1.** Essentials of Ecology and Environmental Science,Dr. S .V .S. Rana, PHI Learning Pvt. Ltd, Delhi

**2.** Environmental Chemistry, Anil Kumar De, Wiley Eastern Limited.

**3.** Environmental Science, T.G. Miller, Wadsworth Publishing Co, 13th edition.

**4.** Ecology and Environment, P. D. Sharma, Rastogi publications

## Data Structures & Algorithms Lab.

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-201-P/IT-201-P<br><br>*Course Credits: 1<br><br>Mode: Practical<br><br>Contact Hours: 2<br><br>Examination Duration: 03 Hrs:<br><br>*In lab. work one credit is equivalent to two hours | **Course Assessment Methods (internal: 30; external: 70)**<br><br>An internal evaluation is done by the course coordinator.<br><br>The end semester practical examination is conducted jointly by external and internal examiners. External examiner is appointed by the COE of the university from the panel of examiners approved by BOSR of the Department of Computer Science and Engineering, Hisar and the internal examiner is appointed by the Chairperson of the Department. |

**Pre-requisites:**

Students are expected to have the strong theoretical concepts and computer fundamentals as well as are expected to be proficient in programming language like 'C'.

**The objectives of this laboratory course are to:**

1. Learn how to efficiently implement basic and advanced data structures and various operations on these data structures in a programming language.
2. Learn how to deal with memory management.

**By the end of the course a student is expected to:**

CO1. Able to choose appropriate data structure as applied to specified problem definition.
CO2. Able to handle operations like searching, insertion, deletion, traversing mechanism etc. on various data structures.
CO3. Able to design various linear and non-linear data structures.
CO4. Able to apply concepts learned in various domains.

Students are required to do eight to ten assignments. The lab. assignments are evenly spread over the semester. Every student is required to prepare a file of lab. experiments done.

List of Programs:-

1. Write C functions to perform following operations on tables using functions only a) Addition b) Subtraction c) Multiplication d) Transpose e) Search an element in a table using linear search.

2. Write C functions (Iterative and recursive) for finding the element in an array using Binary Search Method.
3. Write C functions to implement the following insertion operations on Singly linked list: a) create a singly linked list b)insert a node at the beginning c) insert a node at end c) insert a node after a given location d) c) insert a node before a given location and e) display content of the list.
4. Write C functions to implement the following deletion operations on Singly linked list: a) create a singly linked list b)delete a node at the beginning c) delete a node at end c) delete a node at given location and d) display content of the list.
5. Write C functions for the following to implement an integer stack using arrays: a) Initialize b) Push an element c) pop an element d) Check empty stack e) Check full stack f) Display stack size g) Display stack contents.
6. Write C functions for the following to implement an integer stack using singly linked list: a) Initialize b) Push an element c) pop an element d) Display stack size e) Display stack contents.
7. Write C functions for the following to implement an integer queue using arrays: a) Initialize b) Insert an element c) remove an element d) Check empty queue e) Check full queue f) Display queue size g) Display queue contents.
8. Write C functions for the following to implement an integer circular queue using singly linked list: a) Initialize b) Insert an element c) remove an element d) Display queue size e) Display queue contents.
9. Write the following C functions to implement Binary tree and binary search tree: a) create a Binary tree or binary search tree, b) traverse the BT and BST in pre-order, in-order and post-order.
10. Write C functions to implement the following sorting techniques a) Bubble sort b) Quick sort c) Selection sort d) Merge sort.

**Note:** The list is indicative. The Teacher can alter/add more number of experiments as per the requirement.

## OOPS using C ++ Lab.

**General Course Information**

| | |
|---|---|
| Course Code: CSE-205-P/IT-205-P<br><br>Course Credits: 1<br><br>Mode: Practical<br><br>Contact Hours: 2<br><br>Examination Duration: 03 Hrs:<br><br>*In lab. work one credit is equivalent to two hours | **Course Assessment Methods (internal: 30; external: 70)**<br><br>An internal evaluation is done by the course coordinator.<br><br>The end semester practical examination is conducted jointly by external and internal examiners. External examiner is appointed by the COE of the university from the panel of examiners approved by BOSR of the Department of Computer Science and Engineering, Hisar and the internal examiner is appointed by the Chairperson of the Department. |

**Pre-requisites:**

Students are expected to have the strong theoretical concepts and computer fundamentals.

**The objectives of this laboratory course are to:**

1. Learn how to efficiently implement concepts of classes, methods.
2. Learn how to deal with encapsulation, inheritance.

**By the end of the course a student is expected to:**

CO1. Able to create classes.
CO2. Able to handle operations like encapsulation, inheritance etc.
CO3. Able to write file handling programs.
CO4. Able to apply concepts learned to various domains.

Students are required to do eight to ten assignments. The lab. assignments are evenly spread over the semester. Every student is required to prepare a file of lab. experiments done.

Q 1. C++ program to print ASCII value of a character and convert lowercase to uppercase and vice versa.

Q 2. Create class to get and print details of a student.

Q 3. Raising a number n to a power p is the same as multiplying n by itself p times. Write a function called power ( ) that takes a double value for n and an int value for p, and returns the result as double value. Use a default argument of 2 for p, so that if this argument is omitted, the number will be squared. Write a main ( ) function that gets values from the user to test this function.

Q 4. Create two classes DM and DB which store the value of distances. DM stores distances in metres and centimeters and DB in feet and inches. Write a program that can read values for the class objects and add one object of DM with another object of DB. Use a friend function to carry out the addition operation. The object that stores the results maybe a DM object or DB object, depending on the units in which the results are required. The display should be in the format of feet and inches or meters and centimeters depending on the object on display.

Q 5. Create a class rational which represents a numerical value by two double values- NUMERATOR & DENOMINATOR. Include the following public member Functions:

• constructor with no arguments (default).

• constructor with two arguments.

• void reduce( ) that reduces the rational number by eliminating the highest common factor between the numerator and denominator.

• Overload + operator to add two rational number.

• Overload >> operator to enable input through cin.

• Overload << operator to enable output through cout.

Write a main ( ) to test all the functions in the class.

Q 6. A hospital wants to create a database regarding its indoor patients. The information to store include

a) Name of the patient b) Date of admission c) Disease d) Date of discharge

Create a structure to store the date (year, month and date as its members). Create a base class to store the above information. The member function should include functions to enter information and display a list of all the patients in the database. Create a derived class to store the age of the patients. List the information about all the to store the age of the patients. List the information about all the pediatric patients (less than twelve years in age).

Q 7. C++ program to use function as a LVALUE using reference variable

Q 8. Write a program to access a function with the help of pointer.

Q 9. Make a class **Employee** with a name and salary. Make a class **Manager** inherit from **Employee**. Add an instance variable, named department, of type string. Supply a method to **toString** that prints the manager's name, department and salary. Make a class **Executive** inherit from **Manager**. Supply a method **to String** that prints the string **"Executive"** followed by the information stored in the **Manager** superclass object. Supply a test program that tests these classes and methods.

Q 10. Imagine a tollbooth with a class called toll Booth. The two data items are a type unsigned int to hold the total number of cars, and a type double to hold the total amount of money collected. A constructor initializes both these to 0. A member function called payingCar ( ) increments the car total and adds 0.50 to the cash total. Another function, called nopayCar ( ), increments the car total but adds nothing to the cash total. Finally, a member function called displays the two totals. Include a program to test this class. This program should allow the user to push one key to count a paying car, and another to count a nonpaying car. Pushing the ESC kay should cause the program to print out the total cars and total cash and then exit.

Q 11. Write a function called reversit ( ) that reverses a string (an array of char). Use a for loop that swaps the first and last characters, then the second and next to last characters and so on. The string should be passed to reversit ( ) as an argument. Write a program to exercise reversit ( ). The program should get a string from the user, call reversit ( ), and print out the result. Use an input method that allows embedded blanks. Test the program with Napoleon's famous phrase, "Able was I ere I saw Elba)".

Q 12. Program to write and read an object in, from binary file using write() and read() in C++

Q 13. C++ program to write and read time in/from binary file using fstream

Q 14. C++ Program to implement Stack in STL, Vector in STL

Q 15. C++ Program to Implement String in ST

**Note:** The list is indicative. The Teacher can alter/add more number of experiments as per the requirement.

## Digital Electronics Lab

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-207-P/IT-207-P<br><br>*Course Credits: 1<br><br>Mode: Practical<br><br>Contact Hours: 2<br><br>Examination Duration: 03 Hrs:<br><br>*In lab. work one credit is equivalent to two hours | **Course Assessment Methods (internal: 30; external: 70)**<br><br>An internal evaluation is done by the course coordinator.<br><br>The end semester practical examination is conducted jointly by external and internal examiners. External examiner is appointed by the COE of the university from the panel of examiners approved by BOSR of the Department of Computer Science and Engineering, Hisar and the internal examiner is appointed by the Chairperson of the Department. |

**Pre-requisites: Knowledge of Electronics Components, Boolean Algebra.**

**The objectives of this laboratory course are to:**

1. Realize combinational circuits using transistors such as realize half and full adders and verify truth tables using NAND gates.
2. Realize decoders , multiplexers and various types of flip-flops
3. Verify the operation of decade counter and display the count on seven segment display.
4. Verify the operation of 4 bit shift register using IC 7474.

**By the end of the course a student is expected to:**

CO1. Realize and verify the operation of combinational circuits
CO2. Realize counters and display count on seven-segment dispalys.
CO3. Verify the truth table of flip-flops, multiplexers and demultiplexers
CO4. Realize gates using transistors.

Students are required to do eight to ten assignments. The lab. assignments are evenly spread over the semester. Every students is required to prepare a file of lab. experiments done.

**Note:** The list is indicative. The Teacher can alter/add more number of experiments as per the requirement.

**Skills And Innovation Lab**

| | |
|---|---|
| Course Code: BME-211-P<br><br>Course Credits: 0.0<br><br>Mode: Practical<br><br>Contact Hours: 03 hours per week<br><br>Examination Duration: 03 hours | **Course Assessment Methods (internal: 30; external: 70):** This is a non-credit course of qualifying nature.<br><br>Internal practical evaluation is to be done by the course coordinator. The end semester practical examination will be conducted jointly by external and internal examiners. |

**Prerequisite:** Basic knowledge of computers.

**Objectives:**

1. Understand and identify research topics related to computer science through brain storming sessions.
2. Propose a novel idea/modified technique/new interpretation after identifying the existing research work.
3. Devise specific identified issue/problem in the form of research objectives.
4. Work in a group and communicate effectively the research topic though presentation and/or brain storming.

**Course outcomes:**

CO1. Understand the research analysis of issues/problems on topics related to computer science.

CO2. Understand the techniques and tools used for research analysis.

CO3. Understand literature related to a research topic.

CO4. Communicate effectively the research topic though presentation and/or brainstorming.

**Lab Contents**

A group of students are required to carry out a study related to current development and emerging trends in the field of computer science. Each group of students will also try to improve their basic skills in their respective field. The students may use the equipment's/machines/instruments available in the labs/workshops with the due permission of Chairperson/Director on recommendation of the Course Coordinator.

The students in consultation with the course coordinator will decide the topic of the study. The study report will be submitted by group at the end of semester and will be evaluated by Course Coordinator

## Database Management System

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-202-L/IT-202-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4 hours/week<br><br>Mode: Lectures<br><br>Exam Duration: 3 hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question number one is compulsory and contains seven short answer questions covering entire syllabus. Rest eight questions are set by giving two questions from each of the unit of the syllabus. A candidate is required to attempt any of four questions selecting at least one from each of the four units. All questions carry equal marks. |

**Prerequisite**

Elementary knowledge about computers including some experience of using Unix or Windows. It is beneficial have the knowledge about programming in some common programming language and knowledge about data structures and algorithms, corresponding to the basic course on Data Structures and Algorithms.

**About the Course and its Objectives & Outcomes:**

The objectives of this course are

1. To provide comprehensive coverage of the problems involved in database design, in-depth coverage of data models and database languages, and a survey of implementation techniques applied in modern DBMS.
2. To provide practical skills of conceptual/logical database design and general familiarity with the problems and issues of database management.
3. To develop skills that is appropriate for Database Administrators, Database Application Developers, Database Specialists, and DBMS developers.

By the end of the course a student is expected to be familiar with:

CO1. the basic concepts and appreciate the applications of database systems.
CO2. the distributed databases and concurrency control.
CO3. the basics of SQL and construct queries using SQL.
CO4. a relational database system theory and be able to write relational algebra expressions for queries by writing SQL using the system.

**Course Contents**

**Unit I**

Overview: Database, File Systems vs. DBMS, Characteristics of the Data Base Approach, Database users, Advantages and Disadvantages of a DBMS, Responsibility of Database Administrator.
Data Base Systems Concepts and Architecture: Data Models, Schemas and Instances, DBMS architecture and various views of Data, Data Independence, Database languages.

**Unit II**

E-R Model: Entity Types, Attributes & Keys, Relationships, Roles and Structural Constraints, E-R Diagrams, Reduction of an E-R Diagram to Tables.
Relational Data Model: Relational Algebra & various operations.

**Unit III**

SQL: Data Definition, Constraints, Insert, Delete & Update statements in SQL, Queries in SQL.
Relational Database Design: Functional Dependencies, Integrity Constraints, Decomposition, Nominalization (Up to 4NF).

**Unit IV**

DDBMS Design: Replication and Fragmentation Techniques
Concurrency Control Techniques: ACID properties of a Transaction, Locking Techniques, Time-stamp ordering, Multi-version Techniques, Deadlock, Recovery Techniques in centralized DBMS.

**Text and Reference Books:**


1. Fundamentals of Database systems, Elmasri & Navathe, Addison Wesley, 3rd Edition, New Delhi, 2010.
2. Database Management Systems, R.Pannerselvam, PHI Learning Pvt Ltd, New Delhi , Second Edition, 2011.
3. An Introduction to Database System, Bipin C.Desai, Galgotia Publication, New Delhi, 1990.
4. Essentials of Data Base Management System , Alexis Leon and Mathews Leon, Vikas Publishing Limited, Chennai First Edition, 2009.
5. Database Management Systems, Ramon a.Mato-Toledo, Pauline K.Cushman, Schaums'Outline series, TMH, New Delhi Special Indian Edition 2007.
6. Database Management Systems, Sharad Maheswari, Ruchin Jain, Firewall Media, New Delhi, Second Edition Reprint 2010.

## Principles of Software Engineering

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-204-L/IT-204-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4Hours<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 Hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:** Students are expected to have knowledge of algorithms, flow charts and at least one programming language

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are to:**

1. Introduce students about software development life cycle and models of developing effective and efficient software.
2. Identify software requirements for manual and automated real world systems.
3. Provide the students with the opportunity to practice software development skills.
4. Provide the students with the opportunities to develop basic computing skills with respect to preparation of documents.

**By the end of the course a student is expected to:**

CO1. Understand the basic concepts and issues of Software Development
CO2. Illustrate the software requirement specification and system design.
CO3. Understand about software design methodologies and software testing techniques.
CO4. Understand the purpose and functionality of computer aided software engineering tools.
CO5. Understand about the software quality standards.

### Course Contents

**Unit I**

Introduction to Software and Software Engineering, the process, software products, Phases of software development, software engineering paradigms, software characteristics, role of software engineer and software project manager,

software project management plan (SPMP), Metrics for project size estimation, Software cost estimation, Project scheduling, personnel planning, organization and team structure.

**Unit II**

Requirement Engineering process,Software Requirements, Guidelines for software requirements, Software Requirement Specification, characteristics of SRS, structure of SRS, Structured Analysis, Tools of structured analysis-Data Flow Diagrams, Decision tables, Decision trees, Data dictionary, Structured charts, Object Oriented Analysis, Data modelling, Behavioural Modelling,

**Unit III**

Software Configuration Management, Software Risks, Risk Management, Software Design fundamentals, Design principles, Module level Concepts, Design methodology (Structured design and Object Oriented Design), Design Documentation, User Interface Design.

Coding standards and Guidelines, Code verification techniques, Code documentation, Computer Aided Software Engineering (CASE) tools, characteristics and advantages of CASE tools.

**Unit IV**

Testing fundamentals, Test Plan, Test Case design, Levels of Software Testing-Unit Testing, Integration Testing-Top down Integration and Bottom up Integration Testing, Regression Testing, Smoke Testing, System Testing-Recovery Testing, Security Testing, Stress Testing, Performance Testing, Acceptance Testing-Alpha Testing, Beta Testing. Testing Techniques-White box Testing and Black Box testing.

Software Quality concepts, ISO9126, McCall's Quality factors, SQA, SQA activities, Software Reviews-Review process, Walkthroughs, Formal Technical Review (FTR), Defect amplification Model, ISO 9000 Quality standards, Capability Maturity Model (CMM). Software Reliability, Software Maintenance, Software Re-engineering.

T**ext and Reference Books:**


1. Fundamentals of Software Engineering, Rajib Mall, Prentice Hall India, 2004.
2. An integrated approach to Software Engineering, Pankaj Jalote, Narosa, 3rd ed., 2014.
3. Software Engineering: A practitioner's approach, Roger S. Pressman, McGraw Hill, 7th ed., 2014.
4. Software Engineering, Ian Sommerville,Addison-Wesley, 10th ed. 2015.

## Computer Networks

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-206-L/IT-303-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4 hours/week<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:** Basic knowledge of Digital and Analog Communication

**About the Course and its Objectives & Outcomes:**

This course has been designed with an aim to provide students with an overview of the concepts and fundamentals of data communication and computer networks.

**The objectives of this course are to:**

1. Independently understand basic computer network technology.
2. Recognize the different types of network topologies, internetworking devices and their functions.
3. Familiarize the student to advanced networking concepts; make ready the student for entry advanced courses in computer networking
4. Awareness with the basic protocols of computer networks, and how they can be used to support in network design and implementation.

**By the end of the course a student is expected to:**

CO1. To have depth knowledge of the fundamental concepts of computer networking.
CO2. Explain the role of each layer of the OSI model and TCP/IP and their functions.
CO3. To have knowledge of congestion control and routing algorithms.
CO4. Be able to gain expertise in some specific areas of networking such as the design and maintenance of individual networks.

## Syllabus

**Unit I**
**Introduction:** Uses, Topologies, Reference Models.

**Networking Devices:** Hubs, Repeaters, Bridges, Modems, Switches, Routers, Gateways.

**Unit II**
**Flow Control at Data Link Layer:** Need of flow control, Stop-and-wait, Go-back-N and selective repeat flow control protocols.

**Framing & Error Handling:** Framing Protocols, Error detection and correction mechanisms.

**Multiple Access Communication:** Pure and Slotted, Carrier sense, splitting and controlled access multiple access algorithms.

**LAN Standards & Technologies**: Ethernet (IEEE 802.3, IEEE 802.3u, IEEE 802.3z, IEEE 802.3 ae), Wireless LAN (IEEE 802.11), Bluetooth (IEEE 802.15).

**Unit III**
**Network Layer in Internet:** IPV4 and IPV6, IP addressing (IP Addressing (Classful Addressing, Private IP Addresses, Classless Addressing: Sub-netting and Super-netting, NAT: Network Address Translation), ARP.

**Routing**: Introduction, Administrative Distance, Types of Routing: Default Routing, Static Routing, IGP & EGP. Mobile IP, Internet Control Protocols, Congestion Control Algorithm, Introduction to Voice over IP.

**Unit IV**
**Transport Layer:** Connection Management, Flow control and multiplexing. Basics of Internet Transport Protocols.

**Network Management:** Network Management Basics, Remote Monitoring Techniques, SNMP, Applications.
**The Application Layer**
Introduction to DNS, FTP, TELNET, HTTP, SMTP, Electronic Mail, WWW and Multimedia.

**Text and Reference Books:**


1. Computer Networks, Andrew S Tanenbaum, 5th Edition, Pearson, 2010.
2. Data Communication and networking, Forouzan, 5th Edition, TMH, 2012.
3. Data & Comp. Communication, William Stalling, 6th edition, LPE Pearson Education, 2013.
4. CCNA Study Guide, Todd Lammle, 6th Edition, 2013.
5. RFCs and Internet Drafts, available from Internet Engineering Task Force.

## Java Programming

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-208-L/IT-208-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4<br><br>Mode: Lecture (L)<br><br>Examination Duration: 3 Hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:** Knowledge of any programming language.

**About the Course and its Objectives & Outcomes:**

This course gives you a platform to design and develop market based software applications.

**The objectives of this course are to:**

1. Understand the Object Oriented Technology using Java.
2. Know how to write Java Applications using OO concepts.
3. Have the knowledge of information hiding, reusability and Graphical User Interface.
4. Get an idea how to handle the exceptions occurred at run time environment.
5. Be familiar with parallel processing and synchronization using multithreading environment.

**By the end of the course a student is expected to:**

CO1. Express fundamental concepts of Object Oriented Technology and Java Programming.
CO2. Solve different kind of mathematical problems.
CO3. Design a small Java Based Application.
CO4. Run a problem into multiple parts for better efficiency.
CO5. Design and develop an efficient software application using the concepts of multithreading and exception handling.

**Course Contents**

**Unit I**

**Programming Introduction and Evolution -** Programming Languages: Machine Language, Assembly Language and High Level Languages, Object Technology: Object, Methods, Classes, Instantiation, Reuse, Messages and Methods Calls, Data-types and Instance Variables, Abstraction, Encapsulation, Inheritance, Polymorphism, Java

Development Environment: Creating and Running a Program (understanding all phases), Main Method, Comments, Identifiers and Their Rules, Common Escape Sequences, Packages, Classes and Methods, Anatomy of a Java Program.

**Unit II**

**Decision Making and Overloading –** Java Tokens (Keywords, Identifiers, Literals, Operators (Arithmetic Operators, Relational Operators, Logical Operators, Assignment Operators, Increment and Decrement Operators etc.), Separators, JVM (Java Virtual Machine), Control Statements (Decision Making and Branching), Looping Statements (Decision Making and Looping), Jumping Statements, Objects Declaration, Classes Declaration and Use, User Defined Methods, Visibility Controls, Constructors and Its Types, Constructors Overloading, Methods Overloading, Method Overriding, Static Members, Abstract Methods and Classes.

**Unit III**

**Inheritance, Interfaces, Packages and GUI –** Inheritance: Single Inheritance, Multilevel Inheritance, Hierarchical Inheritance and Hybrid Inheritance, Interfaces: Defining, Extending and Implementing, Packages: Putting Classes Together, Java API Packages, Using System Packages, Creating and Accessing a Package, Adding a class to a package, Hiding Classes, Introduction to GUI Programming: Displaying a Message, Graphics Class, Lines and Rectangles, Circle and Ellipses, Java Applets.

**Unit IV**

**Multithreading, Exception and File Handling –** Creating a Thread, Extending the Thread Class, run Method, Stopping and Blocking a Thread, Life Cycle of a Thread, Thread Methods, Thread Exceptions, Thread Priority, Synchronization, Exception: Types of Errors, Run-Time Error, Try, Catch, Multiple Catch, Finally Statement, Throwing User Defined Exception, Stream Classes Concept, Byte Stream Classes (Input and Output Stream Classes), Character Stream Classes (Reader and Writer Stream Classes), Creation of Files.

**Text and Reference Books:**


1. Java™: How to Program, Paul Deital, Harvey Deital, 9th Edition, Pearson Education (Prentice Hall), 2012.
2. Java™: The Complete Reference, Herbert Schildt, 7th Edition, McGraw-Hill, 2007.
3. Head First Java, Kathy Sierra, Bert Bates, 2nd Edition, O Reilly, 2005.
4. Java Programing From the Group Up, Ralph Bravaco , Shai Simoson, Tata McGraw-Hill.
5. Programming in Java, Sachin Malhotra, Saurabh Chaudhary, Oxford University Press, 2011.
6. Programming with Java: A Premier, E. Balagurusamy, 3rd Edition, Tata McGraw-Hill, New Delhi, 2007.

## Computer Architecture & Organization

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-210-L/IT-210-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 Hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**

Students are expected have the elementary knowledge about digital Electronic and computers system.

**About the Course and its Objectives & Outcomes:**

Computer Architecture and organization described the role of processors, main memory, and input/output devices. Illustrate the simple data path and control designs for processors. It helps to understand the different operations and concept of instructions. It would help the student to learn the basic function and architecture of modern computer system.

**The objectives of this course are to:**

1. This course will teach the fundamentals of Computer Organization and Architecture.
2. Basic understanding of Digital logic and computer design, understanding the concepts and design of instruction code, instruction cycle and operation in instruction sets.
3. Provide basic knowledge of Parallel Processing, Memory Hierarchy and Input /output Techniques.

**By the end of the course a student is expected to:**

CO1. To know the basic knowledge of digital system design and instruction in computer system.
CO2. To be able to perform different operation on instruction sets.
CO3. Understand the addressing sequence and implementation of control memory.
CO4. Have knowledge different type of memory their architecture and access methods.
CO5. Able to understand the parallel processing and different input/ output techniques.

**Course Contents**

**Unit I**

Basic Principles: Boolean algebra and Logic gates, Combinational logic blocks (Adders, Multiplexers, Encoders, de-coder), Sequential logic blocks (Flip-Flops, Registers, Counters); Flynn's classification of computers (SISD, MISD,

MIMD); Multi-level viewpoint of a machine: digital logic, micro architecture, operating systems, high level language; Performance metrics: MIPS, MFLOPS; CPU Architecture types: computer register, (accumulator, register, stack, memory/ register) detailed data path of a typical register based CPU.

**Unit II**

Computer Organization: Store program control concept, Instruction codes, timing and control, instruction cycle; type of instructions: memory reference, register reference, I/O reference; Basics of Logic Design, accumulator logic, Control memory; Micro Programmed Control: address sequencing, micro-instruction formats, micro-program sequencer, Implementation of control unit.

**Unit III**

Instruction Set Architecture & Parallelism: Instruction set based classification of processors (RISC, CISC, and their comparison); Stack Organization, Instruction Formats; addressing modes: register, immediate, direct, indirect, indexed; Operations in the instruction set: Arithmetic and Logical, Data Transfer, Control Flow; Types of interrupts; Introduction to Parallelism: Goals of parallelism (Exploitation of concurrency, throughput enhancement); Amdahl's law; Instruction level parallelism (pipelining, super scaling –basic features); Processor level parallelism (Multiprocessor systems overview).

**Unit IV**

Memory Hierarchy & I/O Techniques: The need for a memory hierarchy (Locality of reference principle, Memory hierarchy in practice: Cache, main memory and secondary memory, Memory parameters: access/ cycle time, cost per bit); Main memory (Semiconductor RAM & ROM organization, memory expansion, Static & dynamic memory types); Cache memory (Associative & direct mapped cache organizations; input-output interface, mode of transfer, DMA (Direct memory transfer).

T**ext and Reference Books:**


1. Digital Logic and Computer Design, Mano, M. Morris, Prentice Hall of India Pvt. Ltd., 1981.
2. Computer System Architecture, M. Morris Mano, Prentice Hall of India Pvt. Ltd., 1993.
3. Computer Architecture and Organization, An Integrated Approach, Milles J. Murdocca, Vincent P. Heuring, John Wiley & Sons Inc., 2007
4. Computer Organization & Architecture, William Stallings, 10th edition, Prentice Hall, 2016.
5. Computer Systems Design and Architecture, Heuring, V.P., Jordan, H.F., Addison Wesley,1997.

# Fundamentals of Management

**General Course Information:**

| | |
|---|---|
| Course Code: HUM-201-L<br><br>Course Credits: 3.0<br><br>Mode: Lecture (L) and Tutorial (T)<br><br>Type: Compulsory<br><br>Contact Hours: 3 hours (L) + 0 hour (T) per week.<br><br>Examination Duration: 03 hours. | **Course Assessment Methods (Internal: 30; External: 70)** Two minor test each of 20marks, class performance measured through percentage of lecture attended (4 marks), assignments, quiz etc. (6 marks) and end semester examination of 70 marks. \<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus; it will contain seven short answer type questions. Rest of the eight questions is to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the four units. All questions carry equal marks. |

**Prerequisite:** The students should have basic understanding of the concept of management and business organizations.

**Objectives:**

- To enhance knowledge skills and attitude to Management.
- To understand management and its relationship with organisation.

**Course outcomes:**
CO-1 To develop the basic understanding of the concept of management and functions of management.
CO-2 the students will come to know about Human Resource management and Marketing management functions of management.
CO-3 Students will come to know about the production activities of any manufacturing organisations.
CO-4 To know that how finances are arranged and disbursed for all the activities of business organisations.

**Unit-I**

Concept of Management: Definitions, Characteristics, Significance, Practical Implications; Management Vs. Administration; Management- Art, Science and Profession; Development of Management Thoughts; Managerial Functions.

**Unit-II**

Concept of Human Resource Management: Human resource planning; Recruitment, Selection, Training and Development, Compensation; Concept of Marketing Management: Objectives and functions of Marketing, Marketing Research, Advertising, Consumer Behaviour.

**Unit-III**

Concept of Production Management, Production Planning and Control, Material management, Inventory Control, Factory location and Production Layout.

**Unit-IV**

Concept of Financial Management, Capital Structure and various Sources of Finance, Working Capital, Short term and long term finances, Capital Budgeting.

**TEXT BOOK:**

1. Principles and Practices of Management: R. S. Gupta, B. D. Sharma, N. S. Bhalla; Kalyani Publishers.
2. Organisation and Management: R. D. Aggarwal; Tata McGraw Hill.

**REFERENCE BOOKS:**

1. Marketing Management: S. A. Sherlikar; Himalaya Publishing House.
2. Financial Management: I.M. Pandey; Vikas Publishing House.
3. Production Management: B. S. Goel; Himalaya Publishing House.

**Database Management System Lab**

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-202-P/IT-202-P<br><br>*Course Credits: 1<br><br>Type: Compulsory<br><br>Contact Hours: 2 hours/week<br><br>Mode: Experimental Lab.<br><br>*In lab work one credit is equivalent to two hours | **Course Assessment Methods (internal: 30; external: 70)**<br><br>An internal practical examination is conducted by the course coordinator.<br><br>The end semester practical examination is conducted jointly by external and internal examiners. External examiner is appointed by the COE of the university from the panel of examiners approved by BOSR of the Department of Computer Science and Engineering, Hisar and the internal examiner is appointed by the Chairperson of the Department. |

**Pre-requisites:**

Student should have sound knowledge of database systems and their entities like tuple, relation, join operation, select operation, project operation etc. Knowledge of MS- Access will be added advantage.

**About the Course and its Objectives & Outcomes:**

**The objectives of this lab course are to:**

1. develop proficiency in execution of commands of the different types of database languages, and
2. teach the database design, query and PL/SQL.

**By the end of the course a student is expected to be able:**

CO1. Create database objects
CO2. Modify database objects
CO3. Manipulate the data
CO4. Retrieving the data from the database server
CO5. Performing database operations in a procedural manner using PL/SQL.
CO6. Design and Develop applications like banking, reservation system, etc.,

**Syllabus**

Create a database and write the programs to carry out the following operation:

1. Create a database
2. Alter the structure of an existing database
3. Add a record in the database
4. Delete a record in the database
5. Modify the record in the database
6. Generate queries
7. Generate the report

8. List all the records of database in ascending / descending order
9. Execute various set operations such as Union, Subtraction and Intersection
10. Execute of Aggregate functions as Sum, Count, Avg, Max, Min etc.
11. Implement various Outer Join operations.

**Reference Books:**


1. Database System Concepts by A. Silberschatz, H. F. Korth and S. Sudarshan, 3rd edition, 1997, McGraw-Hill, International Edition.
2. Teach Your Self SQL/PLSQL using oracle 8i and 9i with SQLJ, Bayross, BPB, 2002.
3. An Introduction to database Systems C. J. Date, 7th edition, Low Priced edition 2000.
4. Oracle 8i-A beginners guide, Abbey, TMH-2000.
5. A Guide to SQL Standard, Date, C. and Darwen, H. 3rd edition, Reading, MA:, Addison-Wesley,1996.

## PC and Computer Networks Lab.

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-206-P/IT-303-P<br><br>*Course Credits: 1<br><br>Mode: Practical<br><br>Contact Hours: 2<br><br>Examination Duration: 03 Hrs:<br><br>*In lab. work one credit is equivalent to two hours | **Course Assessment Methods (internal: 30; external: 70)**<br><br>An internal evaluation is done by the course coordinator.<br><br>The end semester practical examination is conducted jointly by external and internal examiners. External examiner is appointed by the COE of the university from the panel of examiners approved by BOSR of the Department of Computer Science and Engineering, Hisar and the internal examiner is appointed by the Chairperson of the Department. |

**Pre-requisites:** Basic Knowledge of Programming in C

**The objectives of this laboratory course are to:**

1. develop skills to design and analyze the basic scenarios
2. develop skills to configuring network devices (Switches, Routers etc), establishing Local area networks (LAN), implement different routing and wide area network (WAN) protocols.

**By the end of the course a student is expected to:**

CO1. Able to understand different models used for study of computer networks and ability to identify different designs.

CO2. Able to understand, how information transforms while moving through network and understand different technologies used to improve efficiency of communication.

CO3. Able to design and engineer routes to create interconnect of nodes.

CO4. Able to build some simple networking models using the Network Simulator.

Students are required to do eight to ten assignments. The lab. assignments are evenly spread over the semester. Every student is required to prepare a file of lab. experiments done.

## Java Programming Lab

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-208-P/IT-208-P<br><br>*Course Credits: 1<br><br>Mode: Practical<br><br>Contact Hours: 2<br><br>Examination Duration: 03 Hrs:<br><br>*In lab. work one credit is equivalent to two hours | **Course Assessment Methods (internal: 30; external: 70)**<br><br>An internal evaluation is done by the course coordinator.<br><br>The end semester practical examination is conducted jointly by external and internal examiners. External examiner is appointed by the COE of the university from the panel of examiners approved by BOSR of the Department of Computer Science and Engineering, Hisar and the internal examiner is appointed by the Chairperson of the Department. |

**Pre-requisites:** Knowledge of any programming language.

**About the Course and its Objectives & Outcomes:**

This course gives you a platform to design and develop market based software applications.

**The objectives of this course are to:**

1. Understand the Object Oriented Technology using Java.
2. Make students capable of writing Java programs using OO concepts.
3. Make students aware of information hiding, reusability and Graphical User Interface.
4. Make them able to write programs for handling exceptions.

**By the end of the course a student is expected to:**

CO1. Express fundamental concepts of Object Oriented Technology and Java Programming.
CO2. Solve different kind of mathematical problems.
CO3. Design a small Java Based Application.
CO4. Run a problem into multiple parts for better efficiency.
CO5. Design and develop an efficient software application using the concepts of multithreading and exception handling.

Students are required to do eight to ten assignments. The lab. assignments are evenly spread over the semester. Every student is required to prepare a file of lab. experiments done.

# Personality Development

| | |
|---|---|
| **Course Code:** PSY-201-L<br><br>**Course Credit:** 0.0<br><br>**Contact Hours:** 03hrs/week<br><br>**Mode:** Lectures **(L**-2;**T**-01)<br><br>**Examination Duration:** 3 Hours | **Course Assessment Methods (Internal: 30; External: 70)** Two minor test each of 20marks, class performance measured through percentage of lecture attended (4 marks), assignments, quiz etc. (6 marks) and end semester examination of 70 marks. \<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus; it will contain seven short answer type questions. Rest of the eight questions is to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the four units. All questions carry equal marks. |

**Objectives:**

1. Holistic development of the students.
2. Make the students to understand self and personality through the interactive task based sessions.
3. To develop the life skills required to lead an effective personal and professional life.

**Expected outcomes:**

CO1: Understand the concept of self and personality.

CO2: Develop the life skills required to lead an effective personal and professional life.

**Course Contents**

**Unit-I**

Understanding the concept of self, Self-Esteem, Characteristics of individuals with high and low self-esteem. Self-Confidence, Strategies of building self-confidence. Case Study.

**Unit-II**

Understanding Personality, Factors affecting Personality: Biological, Psychological
Social, Theories of Personality: Freud, Allport.
Personality Assessment- Neo-Big Five Personality Test; T.A.T

**Unit-III**

Stress: Causes of Stress and its impact, Strategies of stress management.
Case study.

**Unit-IV**

Emotional Intelligence: Concept, emotional quotient why Emotional Intelligence matters, Measuring EQ, Developing healthy emotions.
Management of anger and interpersonal relations. Case study.

**TEXT BOOK:**

1. Burger, J.M. (1990), Personality, Wardsworth: California.
2. Hall C.S., Lindzey, G. (1978), Theories of Personality, New York: Wiley Eastern Limited.
3. Morgan, C.T.King R.A. Weisz, J.R., and Schopler, J. (1987), Introduction to Psychology, Singapore: Mc Graw Hill.
4. Byronb. D., and Kalley, N. (1961). Introduction to Personality: Prentice Hall.
5. Taylor,S.E., (2009). Health Psychology (9th Ed). New Delhi: Tata McGraw-Hill Publishing Company Ltd.

## Principles of Operating System

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-301-L/IT-301-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: - 4 hours/week<br><br>Mode: Lectures- Tutorial<br><br>Examination Duration: 3 hrs. | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**

Students are expected to be proficient in programming in a standard programming language like C.

**About the Course and its Objectives & Outcomes:**

The objective of this course is to help students become familiar with the fundamental concepts of operating systems and provide students with sufficient understanding of operating system design.

**The objectives of this course are to:**

1. Ability to handle the issues associated with operating systems and comparison of different operating systems
2. Ability to manage different memory management schemes including virtual memory
3. Ability to handle issues related to file system interface and implementation, disk scheduling & management
4. Able to gain knowledge about various process management concepts including scheduling, synchronization and I/0 related issues
5. Ability to resolve deadlocks and to handle the problems associated with deadlock detection and recovery

**By the end of the course a student is expected to:**

CO1. Exhibit familiarity with the fundamental concepts of operating systems.
CO2. Understand deadlock and memory management techniques.
CO3. Exhibit competence in recognizing operating systems features and issues.
CO4. Apply a mature understanding of operating system design and its impact on application systems design and performance.

## Course Contents

**Unit I**

Introductory Concepts: Operating systems functions and characteristics, operating system services and systems calls, system programs, operating system structure. operating systems generation, operating system services and systems calls. Types of Operating systems: Batch operating system, Time-sharing OS, Distributed operating system, Real time systems.
File Systems: Types of Files and their access methods, File allocation methods, Directory Systems: Structured Organizations, directory and file protection mechanisms, disk scheduling and its associated algorithms.

**Unit II**

Processes: Process concept, Process Control Block, Operations on processes, Cooperating processes.
CPU scheduling: Levels of Scheduling, scheduling criteria, Comparative study of scheduling algorithms, Algorithm evaluation, Multiple processor scheduling. Critical-section problem, Semaphores.

**Unit III**

Storage Management: Storage allocation methods: Single contiguous allocation, non-contiguous memory allocation, Paging and Segmentation techniques, segmentation with paging, Virtual memory concepts, Demand Paging, Page-replacement Algorithms, Thrashing.

**Unit IV**

Deadlock: System model, Deadlock characterization, Methods for handling deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from deadlock
Case Studies: Comparative study of WINDOW, UNIX & LINUX system.

T**ext and Reference Books:**


1. Operating System Concepts (8th Edition) by Silberschatz, Peter B. Galvin and Greg Gagne, WileyIndian Edition , 2010.
2. Modern Operating Systems (Third Edition) by Andrew S Tanenbaum, Prentice Hall India, 2008.
3. Principles of Operating Systems by Naresh chauhan, Oxford Press, 2014.
4. Operating Systems by D.M. Dhamdhere, Tata McGraw Hill 2nd edition.
5. Operating Systems (5th Ed) – Internals and Design Principles by William Stallings, Prentice Hall India, 2000

## Microprocessors And Interfacing

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-303-L/IT-206-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4<br><br>Mode: Lecture (L)<br><br>Examination Duration: 3 Hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites**:

The students are expected to have a strong background in the Computer Organization Digital System Design.

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are:**

1. To impart knowledge on the basics of Microprocessor Architecture
2. To acquire knowledge on the concepts of Peripheral Interfacing
3. To develop assembly language Programming skills

**By the end of the course a student is expected to:**

CO1. Understand the architecture of 8085 and 8086 Microprocessor.
CO2. Interface Memory, Input/output with Microprocessor.
CO3. Summarize the functionality of various peripheral chips.
CO4. Ability to develop programs in assembly language

### Course Contents

**Unit-I**

Introduction to microprocessor, 8085 microprocessor architecture, instruction set, interrupt structure.

Architecture of 8086, block diagram of 8086, details of sub-blocks such as EU, BIU; memory segmentation and physical address computations, program relocation.

**Unit II**

Addressing modes, instruction formats, pin diagram and description of various signals.

Instruction execution timing, assembler instruction format, data transfer instructions, arithmetic instructions, branch instructions, looping instructions, NOP and HLT instructions, flag manipulation instructions, logical instructions, shift and rotate instructions, directives and operators, programming examples.

**Unit III**

Assembler directives, Programming with an Assembler, Programming examples, coding style, The art of assembly language programming.

Software Development with Interrupts, Introduction to Stack, Stack Structure of 8086, Introduction to Subroutines, Recursion, MACROS. BIOS (Basic Input/Output System), DOS (Disk Operating System.

**Unit IV**

The 8255 PPI chip: Architecture, control words, modes and examples.

Introduction to DMA process, 8237 DMA controller.

8259 Programmable interrupt controller, Programmable interval timer chips.

T**ext and Reference Books:**


1. Microprocessor Architecture, Programming & Applications with 8085 : Ramesh S Gaonkar; Wiley Eastern Ltd. ,5th edition, 2002.
2. The Intel Microprocessors 8086- Pentium processor : Brey; PHI, 8th edition, 2009
3. Microprocessors and Interfacing Douglas V Hall TMH -2005
4. The 8088 & 8086 Microprocessors-Programming, interfacing,Hardware & Applications :Triebel & Singh; PHI , 4th edition , 2003
5. Microcomputer systems: the 8086/8088 Family: architecture, Programming & Design : Yu-Chang Liu & Glenn A Gibson; PHI, 2001.
6. Advanced Microprocessors and Interfacing : Badri Ram; TMH, 2001.
7. The Intel Microprocessors, Barry B. Brey, 8th Edition, PHI,2013

# High Speed Network Technologies

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-305-L/IT-302-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 HOURS | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**

Basic knowledge of computer networks, layers of OSI reference model, protocols at different layers of OSI reference model.

**About the Course and its Objectives:**

1. Today we have seen sufficient growth in terms of computing power but a lot of work is required to be done to improve communication speed of computers.

2. This course has been designed with an aim that student should learn about different high speed communication technologies like 10 G Ethernet, WiFi, WiMAX, Fiber Channel, GSM, CDMA, ATM, ISDN and Frame Relay.

**Course Outcomes :**

CO1. In-depth knowledge about use and applications of Gigabit and 10 Gigabit Ethernet technology.

CO2. Design and implementation of Fiber Chanel technologies.

CO3. Knowledge about high speed WAN such as ISDN and Frame Relay

CO4. Design and implementation of Wireless technologies such as WiFi, Bluetooth and WiMAX.

CO5. To know the relevance and importance of Internet Suite of Protocols for Fast Network design.

**Syllabus**

**UNIT I**

**HIGH SPEED LAN :Gigabit Ethernet**: Overview of fast Ethernet, Gigabit Ethernet – overview, specifications, layered protocol architecture, network design using Gigabit Ethernet, applications, 10GB Ethernet – overview, layered protocol architecture, applications.

**Fibre Channel** : Fibre channel physical characteristics – topologies & ports, layered protocol architecture, class of service.

**UNIT II**

**HIGH SPEED WAN: Frame Relay** : Protocol architecture and frame format.

**ISDN & B-ISDN** : Channels, interfaces, addressing, protocol architecture, services.

**ATM** : Virtual circuits, cell switching, reference model, traffic management.

**UNIT III**

**WIRELESS LAN: Wireless Networks:** Existing and emerging standards, Wireless LAN(802.11), Broadband Wireless(802.16), Bluetooth(802.15) their layered protocol architecture and security. Mobile Networks – GSM, CDMA and GPRS

**UNIT IV**

**INTERNET SUITE OF PROTOCOLS: Internet Layer** : IPV4 and IPV6, IP addressing, IP classes, CIDR.

**Transport Layer** : UDP/TCP protocols & architecture, TCP connection management.

**Application Layer** : DNS, E-Mail, Voice over IP, audio & video compression.

T**ext and Reference Books:**


1. Computer Networks, Andrew S Tanenbaum, 5th Edition, Pearson 2013
2. Mobile Communication, Jochen Schiller, 2nd Edition, Pearson,2009.
3. Mobile Cellular Telecommunications, Lee, McGRAW- WILL, 2nd Edition, 2006.

## .NET Technologies

### General Course Information:

| | |
|---|---|
| Course Code: CSE-307-L/IT-307-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours:4<br><br>Mode: Lecture(L)-3 Tutorials (T)-1<br><br>Examination Duration: 03 Hrs | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**

Concepts of Object oriented programming approach.

**About the Course and its Objectives & Outcomes:**

This course will cover all the tools of .net framework and popular languages support by the framework. Basic knowledge of database connectivity also included.

**The objectives of this course are to:**

1. To utilize the .NET framework to build applications.
2. To develop ASP.NET Web Services, secure web services, and .NET remoting applications.
3. To understand the protocols behind web services including: SOAP, DISCO, and UDDI.
4. To develop web applications using a combination of client-side (JavaScript, HTML, XML, WML) and server-side technologies (ASP.NET, ADO.NET).

**By the end of the course a student is expected to:**

CO1. Use .NET framework architecture, various tools, and validation techniques,

CO2. Use different templates available in Visual Studio

CO3. Be able to implement and test strategies in real time applications.

CO4. Use advanced concepts related to assembly (DLL), Web Services, WCF, and WPF in project development.

**Course Contents**

**Unit I**

**Introduction to .NET Framework:** NET Architecture, MSIL, CLR, CLS, CTS, JIT, Namespaces, Common Language Implementation, Assemblies, COM, ILDASM, GAC, Strong name, Garbage Collection,.

**C# - Basics and Console Applications in C#:** Namespaces - Constructor and Destructors, Function Overloading & Inheritance, Operator Overloading, Boxing and Unboxing, ‘ref’ and ‘out’ parameters, Modifiers - Property and

Indexers , Attributes & Reflection API, When to use Console Applications - Generating Console Output, Processing Console Input.

**ADO.NET:** Benefits of ADO.NET, ADO.NET compared to classic ADO -, Datasets, Managed Providers and Data Binding: Introducing Data Source Controls, Reading and Write Data Using the SqlDataSource Control

**Unit II**

**C#.NET:** Language Features and Creating .NET Projects, Namespaces Classes and Inheritance, Namespaces Classes and Inheritance, Exploring the Base Class Library, Debugging and Error Handling, Data Types, Exploring Assemblies and Namespaces, String Manipulation ,Files and I/O, Collections.

**Windows Forms and Controls in details:** The Windows Forms Model, Creating Windows Forms Windows Forms Properties and Events, Delegates, Windows Form Controls, Menus, Dialogs, ToolTips.

**Visual Inheritance in C#.NET:** Apply Inheritance techniques to Forms, Creating Base Forms, and Programming Derived Forms.

**Mastering Windows Forms:** Handling Multiple Events, GDI+, Creating Windows Forms Controls

**Unit III**

**ASP.NET:** Introduction to ASP.NET, Working with Web and HTML Controls, Using Rich Server Controls, Login controls, Overview of ASP.NET Validation Controls, Using the Simple Validations, Using the Complex Validators Accessing Data using ADO.NET, Using the Complex Validators Accessing Data using ADO.NET, Configuration Overview, UNICODE,

**Managing State:** Preserving State in Web Applications and Page-Level State, Using Cookies to Preserve State, ASP.NET Session State, Storing Objects in Session State, Configuring Session State, Setting Up an Out-of-Process State Server, Storing Session State in SQL Server, Using Cookie less Session IDs, Application State Using the Data List and Repeater Controls, Overview of List-Bound Controls, Creating a Repeater Control and Data List Control.

**Unit IV**

**Themes and Master Pages:** Creating a Consistent Web Site, ASP.NET 5 Themes, Master Pages, Displaying Data with the Grid View Control Introducing the Grid View Control, Filter Data in the Grid View Control, Allow Users to Select from a Dropdown List in the Grid, Add a Hyperlink to the Grid, Deleting a Row and Handling Errors

**Creating and Consuming Web Services:** The Motivation for XML Web Services, Creating an XML Web Service with Visual Studio, SAO architecture, Designing XML Web Services, Creating Web Service Consumers, Discovering Web Services Using SOAP, DISCO, and UDDI.

**Advanced in .NET:** Introduction to Windows Presentation Foundation (WPF), Window Communication Foundation and its Application.

T**ext and Reference Books:**

1. Mastering C# and .NET Framework by Marino Posadas, 2016.
2. Beginning ASP.NET 4.5 in C# and VB, Wrox, 2012.
3. Beginning ASP.NET 4.5 in C#, Apress, 2012.
4. .NET 4.5 Programming 6-in-1, Black Book by Kogent Learning Solutions Inc. 2013.
5. C # PROGRAMMING with .Net Framework, by Dr. Ashutosh Kumar Bhatt, Kamlesh Padaliya, 2016.
6. Pro C# with .NET 3.0, by Andrew Troelsen, Apress, 2007

## Web Development

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-309-L/IT-304-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 HOURS | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**

Basic knowledge of HTML, XML, ASP, JSP and Web Designing.

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are to:**

1. Learn HTML, XML and design various web pages.
2. Study about Client Side Programming and Server Side Programming.
3. Learn ASP, JSP and its uses in web designing process with HTML.

**By the end of the course a student is expected to:**

CO1. Learn Web Designing Complete Process.
CO2. Made Web Pages using HTML and XML.
CO3. To use ASP and JSP with HTML in web design..
CO4. Get proficient in using HTML and XML.

### Syllabus

**Unit I**

Information Architecture The Role of Information Architect, Collaboration and Communication, Organizing information, Organizational challenges, Organizing Web Sites and Intranets, Creating Cohesive Organization Systems, Designing Navigation Systems, Types of navigation Systems, Integrated Navigation Elements, Remote Navigation Elements, Designing Elegant Navigation Systems, Designing the Search Interface, Indexing the Right Stuff, Grouping Content, Conceptual Design; High-Level Architecture Blueprints, Architectural Page Mockups, Design Sketches.

**Unit II**

Dynamic HTML and Web Designing HTML Basic Concepts, Good Web Design, Process of Web Publishing, Phases of Web Site development, Structure of HTML documents, HTML Elements - Core attributes, absolute and relative links, ordered and unordered lists, Linking Basics, Linking in HTML, Images and Anchors, Anchor Attributes, Image Maps, Semantic Linking Meta Information, Image Preliminaries, , Images as Buttons, Introduction to Layout: Backgrounds, Colors and Text, Fonts, Layout with Tables, Advanced Layout : Frames and layers, HTML and other media types, FORMS, Forms Control, New and emerging Form Elements.

**Separating style from structure with style sheets**: Internal style specifications within HTML, External linked style specification using CSS, page and site design considerations, Positioning with Style sheets.

**Unit III**

**Client side programming**: Introduction to the JavaScript syntax, the JavaScript object model, Event handling, Output in JavaScript, Forms handling, miscellaneous topics such as cookies, hidden fields, and images; Applications.

**Server side programming**: Introduction to Server Side Technologies CGI/ASP/JSP., Programming languages for server Side Scripting, Configuring the server to support CGI, applications; Input/ output operations on the WWW, Forms processing, (using PERL/VBSCRIPT/JavaSCIPT)

**Unit IV**

**Java Server Pages and Active Server Pages**: Basics, Integrating Script, JSP/ASP Objects and Components, configuring and troubleshooting,: Request and response objects, Retrieving the contents of a an HTML form, Retrieving a Query String, Cookies, Creating and Reading Cookies. Using application Objects and Events.

Overview of advance features of XML, XML Relationship between HTML, SGML, and XML, The future of XML.

T**ext and Reference Books:**


1. HTML-The Complete Reference, Thomas A Powell, Tata McGraw Hill, 3rd edition, 2001.
2. CGI Programming with Perl, Scott Guelich, Shishir Gundavaram, Gunther Birzniek, 2nd edition,.O'Reilly, 2000.
3. Programming Web Services with SOAP, Doug Tidwell, James Snell, Pavel Kulchenko, O'Reilly, 2009.
4. XML in Action, Web Technology, Pardi, PHI,1999.
5. XML Step by Step, Yong, PHI, 2nd edition, 2002
6. Web Authoring Desk Reference, Aaron Weiss, Rebecca Taply, Kim Daniels, Stuven Mulder, Jeff Kaneshki, Techmedia Publications.

## OS lab

**General Course Information**

| | |
|---|---|
| Course Code: CSE-301-P/ IT-301-P<br><br>*Course Credits: 1<br><br>Mode: Practical<br><br>Contact Hours: 2 hours/week<br><br>Examination Duration: 03 Hrs:<br><br>*In lab. work one credit is equivalent to two hours | **Course Assessment Methods (internal: 30; external: 70)**<br><br>An internal evaluation is done by the course coordinator.<br><br>The end semester practical examination is conducted jointly by external and internal examiners. External examiner is appointed by the COE of the university from the panel of examiners approved by BOSR of the Department of Computer Science and Engineering, Hisar and the internal examiner is appointed by the Chairperson of the Department. |

**Pre-requisites:**

Students are expected to have basic concepts (theoretical) of computer graphics as well as programming capability in C/C++.

**The objectives of this laboratory course are to:**

1. Teach students various Linux utilities
2. To make them aware of shell scripting, sed scripts, awk programming.

**By the end of the course a student is expected to:**

CO1. To use and execute vi editor, Emacs editor.
CO2. Run general commands and utilities.
CO3. Use file system related commands
CO4. To write basic shell scripts, use sed commands, write awk programs, use perl command line.

Students are required to do eight to ten assignments. The lab. assignments are evenly spread over the semester. Every students is required to prepare a file of lab. experiments done.

List of Experiments/Practicals

- Study of WINDOWS operating system.
- Study of LINUX Operating System (Linux kernel, shell, basic commands pipe & filter commands).
- Administration of LINUX Operating System.
- Writing of Shell Scripts (Shell programming).
- AWK programming.

**Note:** The list is indicative. At least 5 to 10 more exercises to be given by the teacher concerned. The Teacher can alter/add more number of experiments as per the requirement.

**Recommended Books:**


1. Operating System Concepts, (6th Edition), by Abraham Silberschatz, Peter Baer Galvin, Greg Gagne.
2. A Practical Guide to Linux Commands, Editors, and Shell Programming, by Mark G. Sobell
3. Linux: A Practical Approach, by B. Mohamed Ibrahim, Laxmi Publications; First edition (2016)

## Microprocessors and interfacing Lab

**General Course Information**

| | |
|---|---|
| Course Code: CSE-303-P/IT-206-P<br><br>*Course Credits: 1<br><br>Mode: Practical<br><br>Contact Hours: 2 hours/week<br><br>Examination Duration: 03 Hrs:<br><br>*In lab. work one credit is equivalent to two hours | **Course Assessment Methods (internal: 30; external: 70)**<br><br>An internal evaluation is done by the course coordinator.<br><br>The end semester practical examination is conducted jointly by external and internal examiners. External examiner is appointed by the COE of the university from the panel of examiners approved by BOSR of the Department of Computer Science and Engineering, Hisar and the internal examiner is appointed by the Chairperson of the Department. |

**Pre-requisites:**

Students are expected to have basic concepts of Digital Electronics and Logic Design, Computer Organization

**The objectives of this laboratory course are to:**

1. To become familiar with the instruction set of Intel 8085 and 8086 microprocessor.
2. To provide practical hands on experience with Assembly Language Programming.
3. To familiarize the students with interfacing of various peripheral devices.

**By the end of the course a student is expected to:**

CO1. Describe and comprehend the instruction set of 8085.
CO2. Describe and comprehend the instruction set of 8086.
CO3. Understand the principles of Assembly Language Programming.
CO4. Apply assembly language programming in developing microprocessor based applications.

Students are required to do eight to ten assignments. The lab. Assignments are evenly spread over the semester. Every students is required to prepare a file of lab. experiments done.

## .NET Lab

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-307-P/IT-307-P<br><br>*Course Credits:<br><br>Mode: Practical<br><br>Contact Hours:<br><br>Examination Duration: 03 Hrs:<br><br>*In lab. work one credit is equivalent to two hours | **Course Assessment Methods (internal: 30; external: 70)**<br><br>An internal evaluation is done by the course coordinator.<br><br>The end semester practical examination is conducted jointly by external and internal examiners. External examiner is appointed by the COE of the university from the panel of examiners approved by BOSR of the Department of Computer Science and Engineering, Hisar and the internal examiner is appointed by the Chairperson of the Department. |

**Pre-requisites:**

Student should be familiar with OOPs concept.

**The objectives of this laboratory course are to:**

1. To develop programs practically too good understanding of technology.
2. To make better understanding of dependent applications.

**By the end of the course a student is expected to:**

CO1. To play in DOS with .NET framework architecture .
CO2. Write programs in C#.
CO3. Write programs in ASP.Net.
CO4. To build and call library files.

Students are required to do eight to ten assignments. The lab. assignments are evenly spread over the semester. Every students is required to prepare a file of lab. experiments done.
List of Experiments/Practical's (if any)

1. Write a program to check whether empty query string is entered in Asp .net
2. Write a program to change color of Label text control programmatically in Asp .Net
3. Write a program to Enable-Disable Textbox and change width of TextBox programmatically in Asp .Net

4. Write a program to increase and decrease font size programmatically.
5. Write C# code to display the asterisk pattern as shown below:
   *****
   *****
   *****
   *****
   *****
6. Write C# code to prompt a user to input his/her name and country name and then the output will be shown as an example below: Hello Ram from country India!
7. Write C# code to do the following
   - Convert binary to decimal
   - Convert decimal to hexadecimal
   - Convert decimal to binary
   - Convert decimal to octal
8. Write C# code to convert infix notation to postfix notation.
9. Write a C# code to convert digits to words
10. Write a C# code to convert following currency conversion. Rupees to dollar, frank, euro.
11. Write a C# code to Perform Celsius to Fahrenheit Conversion and Fahrenheit to Celsius conversion.
12. Write ASP.Net program to Store Objects in Session State and Storing Session State in SQL Server.

**Note:** The list is indicative. The Teacher can alter/add more number of experiments as per the requirement.

## Web Development Lab

**General Course Information**

| | |
|---|---|
| Course Code: CSP-309-P/It-304-P<br><br>*Course Credits: 1<br><br>Mode: Practical<br><br>Contact Hours: 2<br><br>Examination Duration: 03 Hrs:<br><br>*In lab. work one credit is equivalent to two hours | **Course Assessment Methods (internal: 30; external: 70)**<br><br>An internal evaluation is done by the course coordinator.<br><br>The end semester practical examination is conducted jointly by external and internal examiners. External examiner is appointed by the COE of the university from the panel of examiners approved by BOSR of the Department of Computer Science and Engineering, Hisar and the internal examiner is appointed by the Chairperson of the Department. |

**Pre-requisites:**

Knowledge of HTML language, XML Language.

**The objectives of this laboratory course are to:**

1. Make students expertise in HTML Language.
2. Learn PERL/XML/JSP along with HTML Language.
3. Understand various uses of HTML/CGI.

**By the end of the course a student is expected to:**

CO1. Be able to develop the HTML programs.

CO2. Be able to develop XML Programs.

CO3. Describe the use of HTML/CGI and Get expertise in use of XML/JSP/PERL along with HTML/CGI.

CO4. Be able to write code HTML/CGI using XML/JSP/PERL.

**Software and Tools to be learnt:** Simple Notepad and PERL Langauage.

## Computer Graphics

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-302-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4hrs/Week<br><br>Mode: Lecture(L)<br><br>Examination Duration:3hrs | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**
Programming skills in C/C++ and Data Structures.

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are to:**

1. Study the graphics techniques and algorithms.
2. Study the multimedia concepts and various I/O technologies
3. Enable the students to represent 3-D objects on 2-D display devices.

**By the end of the course a student is expected to have:**

CO1. Familiarity of the principles of graphical user interfaces, system support for graphical, window-based systems, and introductory concepts in computer graphics and multimedia processing.
CO2. Knowledge of 2-D and 3-D transformations.
CO3. Knowledge of different algorithms of clipping, hidden surface rempval etc.
CO4. Ease to use techniques which will allow them to create user-friendly interfaces for computer applications.

### Course Contents

**Unit I**

**Introduction to Computer Graphics:** What is Computer Graphics, Computer Graphics Applications, Computer Graphics Hardware and software, Two dimensional Graphics Primitives: Points and Lines, Line drawing algorithms: DDA, Bresenham's; Circle drawing algorithms: Using polar coordinates, Bresenham's circle drawing, mid point circle drawing algorithm; Filled area algorithms: Scanline: Polygon filling algorithm, boundary filled algorithm.

**Unit II**

**Two/Three Dimensional Viewing:** The 2-D viewing pipeline, windows, viewports, window to view port mapping; Clipping: point, clipping line (algorithms):- 4 bit code algorithm, Sutherland-cohen algorithm, parametric line clipping algorithm (Cyrus Beck).

Polygon clipping algorithm: Sutherland-Hodgeman polygon clipping algorithm. Two dimensional transformations: transformations, translation, scaling, rotation, reflection, composite transformation.

Three dimensional transformations: Three dimensional graphics concept, Matrix representation of 3-D Transformations, Composition of 3-D transformation.

**Unit III**

**Viewing in 3D:** Projections, types of projections, the mathematics of planner geometric projections, coordinate systems.

**Hidden surface removal:** Introduction to hidden surface removal .Z- buffer algorithm , scanline algorithm, area sub-division algorithm.

**Unit IV**

**Representing Curves and Surfaces:** Parametric representation of curves: Bezier curves, B-Spline curves. Parametric representation of surfaces; Interpolation method.

**Illumination, shading, image manipulation:** Illumination models, shading models for polygons, shadows, transparency. What is an image? Filtering, image processing, geometric transformation of images.

**Text and reference books:**


1. Computer Graphics Principles and Practices second edition by James D. Foley, Andeies van Dam, Stevan K. Feiner and Johb F. Hughes, 2000, Addision Wesley.
2. Computer Graphics by Donald Hearn and M.Pauline Baker, 2nd Edition, 1999, PHI
3. Procedural Elements for Computer Graphics – David F. Rogers, 2001, T.M.H Second Edition
4. Fundamentals of 3Dimensional Computer Graphics by Alan Watt, 1999, Addision Wesley.
5. Computer Graphics: Secrets and Solutions by Corrign John, BPB
6. Graphics, GUI, Games & Multimedia Projects in C by Pilania & Mahendra, Standard Publ.
7. Computer Graphics By Pradeep K Bhatia, IK International Pub, New Delhi, 3ed, 2013
8. Introduction to Computer Graphics By N. Krishanmurthy T.M.H 2002

## Intelligent Systems

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-304-L/IT-309-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 Hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**

Knowledge of Data structure, Probability and Statistics

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are to:**

1. Study the concepts of artificial intelligence.
2. Understand and learn methods for solving problem using artificial intelligence.
3. Understand various ways to represent knowledge in AI.
4. Study the concept of expert system and machine learning.

**By the end of the course a student is expected to:**

CO1. Apply underlying searching techniques in basic problems.
CO2. Select and apply knowledge representation to solve AI problems.
CO3. Study expert systems.
CO4. Apply machine learning techniques in various domains.

**Course Contents**

**Unit-I**

**Introduction to AI:** Introduction, Turing Test, AI problems and techniques, production system & its characteristics, problem characteristics.

**Problem Solving Using Search:** Blind search techniques - Breadth first search, Depth first search. Heuristic search techniques - Generate and test, Hill Climbing, Best first search, A* Algorithm, AO* Algorithm.

**Unit-II**

**Knowledge Representation:** Introduction, Knowledge Representation- Representation and Mappings, Issues in Knowledge Representation, Symbolic Logic - Propositional logic, Predicate logic- Representing simple facts in logic, Representing Instances and ISA Relationship, Computable functions and Predicates, Unification, Resolution, Natural Deduction.

**Representing Knowledge Using Rules:** Procedural versus Declarative Knowledge, Logic Programming, Forward versus Backward Reasoning, Matching, Control Knowledge.

**Unit-III**

**Reasoning Under Uncertainty:** Introduction to Nonmonotonic Reasoning, Logics for Nonmonotonic Reasoning, Implementation Issues, Probability and Baye's Theorem, Certainty Factors and Rule-based Systems, Bayesian Networks, Dempster-Shafer Theory.

**Fuzzy logic system:** Introduction, Crisp Set, Fuzzy Sets, Fuzzy Membership Functions, Operations on Fuzzy Sets, Fuzzy Relations.

**Unit-IV**

**Game Playing:** Overview, The Minimax Search Procedure, Adding Alpha-Beta Cut-offs, Iterative Deepening.

**Learning:** Introduction, Rote Learning, Learning by Taking Advice, Learning From Examples, Explanation Based Learning, Discovery, Analogy.

**Expert system:** Introduction, Architecture, Representing and Using Domain Knowledge, Expert System Shells, Explanation, Knowledge Acquisition.

T**ext and Reference Books:**


1. Artificial intelligence, Elaine Rich, Kevin Knight and Shivashankar B Nair, McGraw Hill Education. 3rd edition, 2009.
2. Artificial intelligence: A modern Approach, Stuart Russel and Peter Norvig, Pearson Education, 3rd edition, 2015
3. Introduction to Artificial Intelligence and Expert System, Dan W. Patterson, Pearson Education.1st edition, 2007.
4. A first course in Artificial Intelligence, Deepak Khemani, McGraw Hill Education. 3rd edition, 1st edition, 2013.
5. Artificial Intelligence: Structures and Strategies for Complex Problem Solving, George F. Luger, Pearson Education, 5th edition, 2009.

## Android Programming

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-306-L/IT-306-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4<br><br>Mode: Lecture (L)<br><br>Examination Duration: 3 Hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**

Java programming and Object-oriented programming, Knowledge of RDBMS and OLTP

**About the Course and its Objectives & Outcomes:**

This course will teach students how to develop Android apps.

**The objectives of this course are to:**

1. The Android environment
2. Tools for creating Android applications
3. The Android approach to structuring applications
4. Basic user interfaces
5. Application life cycles

**By the end of the course a student is expected to:**

CO1. To have knowledge of android evolution and its architechture.
CO2. To be able to understand the process of developing software for the android mobile.
CO3. To be able to create mobile applications on the Android Platform.
CO4. To be able to create mobile applications involving data storage in SQLite database

**Course Contents**

**Unit I**

**Basics of Android:** OOPs concepts, What is Android, History and Version, Android architectural overview, Challenges of the mobile platform, Installing software, Setup Eclipse, Hello Android example, Internal Details, Dalvik VM, Software Stack, Android Core Building Blocks, Android Emulator, AndroidManifest.xml, R.java file, Hide Title Bar, Screen Orientation.

**Unit II**

**UI Widgets:** Working with Button, Toast, Custom Toast, Button, Toggle Button, Switch Button, Image Button, CheckBox, Alert Dialog, Spinner, AutoCompleteTextView, RatingBar, DatePicker, TimePicker, ProgressBar, Quick Contact Budge, Analog Clock and Digital Clock, Working with hardware Button, File Download.

**Unit III**

**Activity, Intent & Fragment:** Activity Lifecycle, Activity Example, Implicit Intent, Explicit Intent, Fragment Lifecycle, Fragment Example, Dynamic Fragment.
**Android Menu:** Option Menu, Context Menu, Popup Menu
**Layout Manager:** Relative Layout, Linear Layout, Table Layout, Grid Layout.

**Unit IV**

**Adaptor:** Array Adaptor, ArrayList Adaptor, Base Adaptor.
**View:** GridView, WebView, ScrollView, SearchView, TabHost, DynamicListView, Expanded ListView.
**SQLite:** SQLite API, SQLite Spinner, SQLite ListView
**XML & JSON:** XML Parsing SAX, XML Parsing DOM, XML Pull Parser, JSON basics, JSON Parsing.

T**ext and Reference Books:**


1. Android Programming by Redazione Io Programmo, 2011
2. Android Programming for Beginners by John Horton, 2015
3. Android Database Programming by Jason Wei, 2012
4. Android Programming Tutorials, 3rd Edition by Mark L Murphy, 2010
5. Android Programming - The "Big Nerd Ranch" Guide by Bill Phillips et al., 2017
6. Android Application Development: Programming with the Google SDK by Rick Rogers et al. 2009

## Analysis and Design of Algorithms

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-308-L/IT-305-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours:4<br><br>Mode: Lecture(L)-3 Tutorials (T)-1<br><br>Examination Duration: 03 Hrs | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:** Knowledge of Data Structure and a Programming Language

**About the Course and its Objectives & Outcomes:**

This Course focus on effective and efficient design of algorithms. In this course various algorithm design techniques and their analysis is to be studied. After studying this subject a student will be expected to find one of the better techniques for solving any given programming problem and prove it analytically.

**The objectives of this course are to:**

1. To understand the basics of algorithm design and its analysis
2. To study and analyse various data structures and sorting & searching techniques.
3. To understand the contextual applicability of problem specific algorithm design techniques
4. To study and understand benefits and limitations of above mentioned techniques.

**By the end of the course a student is expected to:**

CO1. Select one of the better techniques for solving any given programming problem and prove it analytically.
CO2. Recognize the use of several design techniques and use these methods to solve simple problems.
CO3. Write and solve recurrence relations for recursive algorithms.
CO4. Determine asymptotic growth rates for algorithms.

**Course Contents**

**Unit I**

Algorithms , Algorithms as a technology, Insertion sort, Analyzing algorithms, Asymptotic notations, Divide and Conquer: General method, binary search, merge sort, qick sort, Strassen‟s matrix multiplication algorithms and analysis of algorithms for these problems.

**Unit II**
Sorting and Data Structure: Heapsort, Hash Tables, Red Black Trees
Greedy Method: General method, knapsack problem, minimum spanning trees, single souce paths and analysis of these problems.

**Unit III**
Dynamic Programming: General method, matrix chain multiplication, longest common subsequence, optimal binary search trees,
Back Tracking: General method, 8 queen's problem, graph colouring, Hamiltonian cycles, analysis of these problems.

**Unit IV**
Branch and Bound: Method, O/I knapsack and traveling salesperson problem
NP Completeness: Polynomial time, NP-completeness and reducibility, NP-complete problems

Text and Reference Books:


1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronalde L. Rivest, Clifford Stein, MIT press, 3rd Edition, 2009.
2. Fundamental of Computer Algorithms, Ellis Horowitz, Satraj Sahni, Sanguthevar Rajasekaran, Galgotia publivcation pvt. Ltd., 1999.
3. Algorithms, S. Dasgupta, C. Papadimitriou, and U. Vazirani, McGraw-Hill Higher Education. 2006

## Theory of Automata and Computation

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-310-L/IT-310-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4<br><br>Mode: Lecture (L)<br><br>Examination Duration: 3 Hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**

The students are expected to have a strong background in the fundamentals of discrete mathematics like in the areas of symbolic logic, set, induction, number theory, summation, series, combinatorics, graph, recursion, basic proof techniques.

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are:**

1. Give an account of important concepts and definitions for automata and formal languages;
2. Classify machines by their power to recognize languages.
3. Employ finite state machines to solve problems in computing.
4. Explain deterministic and non-deterministic machines.
5. Comprehend the hierarchy of problems arising in the computer sciences.

**By the end of the course a student is expected to:**

CO1. Understand the basic concepts and application of Theory of Computation.
CO2. Understand the concept of abstract machines and their power to recognize the languages
CO3. Attains the knowledge of language classes & grammars relationship among them with the help of Chomsky hierarchy.
CO4. Apply this basic knowledge of Theory of Computation in the computer field to solve computational problems and in the field of compiler also.

**Course Contents**

**Unit I**

Finite Automata and Regular Expressions: Finite State Systems, Basic Definitions Non-Deterministic finite automata (NDFA), Deterministic finite automata (DFA), Equivalence of DFA and NDFA Finite automata with E-

moves, Regular Expressions, Equivalence of finite automata and Regular Expressions, Regular expression conversion and vice versa, Conversion of NFA to DFA by Arden‟s Method.

**Unit II**

Introduction to Machines: Concept of basic Machine, Properties and limitations of FSM. Moore and mealy Machines, Equivalence of Moore and Mealy machines.

Properties of Regular Sets: The Pumping Lemma for Regular Sets, Applications of the pumping lemma, Closure properties of regular sets, Myhill-Nerode Theorem and minimization of finite Automata, Minimization Algorithm.

**Unit III**

Grammars: Definition, Context free and Context sensitive grammar, Ambiguity regular grammar, Reduced forms, Removal of useless Symbols and unit production, Chomsky Normal Form (CNF), Griebach Normal Form (GNF).

Pushdown Automata: Introduction to Pushdown Machines, Application of Pushdown Machines

**Unit IV**

Turing Machines: Deterministic and Non-Deterministic Turing Machines, Design of T.M, Halting problem of T.M., PCP Problem.

Chomsky Hierarchies: Chomsky hierarchies of grammars, Unrestricted grammars, Context sensitive languages, Relation between languages of classes.

Computability: Basic concepts, Primitive Recursive Functions.

**Text and Reference Books and Links:**


1. Introduction to automata theory, language & computations- Hopcroaft & O.D.Ullman, R Mothwani, 2001, AW
2. Theory of Computer Sc.(Automata, Languages and computation):K.L.P.Mishra & N.Chandrasekaran, 2000, PHI.
3. Introduction to formal Languages & Automata-Peter Linz, 2001, Narosa Publ..
4. Fundamentals of the Theory of Computation- Principles and Practice by RamondGreenlaw and H. James Hoover, 1998, Harcourt India Pvt. Ltd..
5. Elements of theory of Computation by H.R. Lewis & C.H. Papaditriou, 1998, PHI. • Introduction to languages and the Theory of Computation by John C. Martin 2003, T.M.H.

## Computer Graphics Lab

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-302-P<br><br>*Course Credits: 1<br><br>Mode: Practical<br><br>Contact Hours: 2<br><br>Examination Duration: 03 Hrs:<br><br>*In lab. work one credit is equivalent to two hours | **Course Assessment Methods (internal: 30; external: 70)**<br><br>An internal evaluation is done by the course coordinator.<br><br>The end semester practical examination is conducted jointly by external and internal examiners. External examiner is appointed by the COE of the university from the panel of examiners approved by BOSR of the Department of Computer Science and Engineering, Hisar and the internal examiner is appointed by the Chairperson of the Department. |

**Pre-requisites:**

Knowledge of Computer and C.

**The objectives of this laboratory course are to:**

1. Implement the line drawing techniques.
2. Enable the students to represent 3-D objects.

**By the end of the course a student is expected to:**

CO1. Understand and learn the concepts of graphics in C.
CO2. Implement various line drawing algorithm .
CO3. Implement various clipping and curve drawing algorithms.
CO4. Apply graphical techniques to make graphical applications.

Students are required to do eight to ten assignments. Lab assignments are evenly spread over the semester. Every student is required to prepare a file of lab. experiments done.

## Intelligent Systems Lab

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-304-P/It-309-P<br><br>*Course Credits: 1<br><br>Mode: Practical<br><br>Contact Hours: 2<br><br>Examination Duration: 03 Hrs:<br><br>*In lab. work one credit is equivalent to two hours | **Course Assessment Methods (internal: 30; external: 70)**<br><br>An internal evaluation is done by the course coordinator.<br><br>The end semester practical examination is conducted jointly by external and internal examiners. External examiner is appointed by the COE of the university from the panel of examiners approved by BOSR of the Department of Computer Science and Engineering, Hisar and the internal examiner is appointed by the Chairperson of the Department. |

**Pre-requisites:**

Knowledge of Computer and Information System, Programming Languages

**The objectives of this laboratory course are to:**

1. Implement the concepts of AI using prolog.
2. Use and modify heuristic state-space search algorithms.
3. Implement various AI techniques/algorithms in Lab.

**By the end of the course a student is expected to:**

CO1. Understand and learn the concepts of prolog.
CO2. Write simple programs in prolog.
CO3. Understand and implement various AI techniques.
CO4. Apply AI techniques to solve problems.

Students are required to do eight to ten assignments. Lab assignments are evenly spread over the semester. Every student is required to prepare a file of lab. experiments done.

List of Experiments/Practical's (if any)

1. Turbo Prolog features and format.
2. Write a program for usage of rules in prolog.
3. Write a program for using Input, Output and fail predicates in prolog.
4. Write a program for studying usage if arithmetic operators in prolog.
5. Write a program to study usage of Cut, Not, Fail predicates in prolog.
6. Write a program to study usage of recursion in prolog.
7. Write a program to implement DFS/ BFS.
8. Write a program to implement A* algorithm.

9. Write a program to solve 8 queens problem.
10. Write a program to solve travelling salesman problem.

**Note:** The list is indicative. The Teacher can alter/add more number of experiments as per the requirement.

**Recommended Books:** Lab Manual

## Android Lab

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-306-P/IT-306-P<br><br>*Course Credits: 1<br><br>Mode: Practical<br><br>Contact Hours: 2<br><br>Examination Duration: 03 Hrs:<br><br>*In lab. work one credit is equivalent to two hours | **Course Assessment Methods (internal: 30; external: 70)**<br><br>An internal evaluation is done by the course coordinator.<br><br>The end semester practical examination is conducted jointly by external and internal examiners. External examiner is appointed by the COE of the university from the panel of examiners approved by BOSR of the Department of Computer Science and Engineering, Hisar and the internal examiner is appointed by the Chairperson of the Department. |

**Pre-requisites:**

The students are expected to have a knowledge of Java programming, object-oriented programming, RDBMS and OLTP.

**The objectives of this laboratory course are to:**

1. To develop programs practically too good understanding of technology.
2. To understand Android Studio framework.

**By the end of the course a student is expected to:**

CO1. To be able to understand the process of developing software for the android mobile.
CO2. To be able to create mobile applications on the Android Platform.
CO3. Student will be able to design android application.
CO4. To be able to design database connectivity application.

Students are required to do eight to ten assignments. The lab assignments are evenly spread over the semester. Every student is required to prepare a file of lab experiments done.

**List of Experiments/Practical's (if any)**

1. Create "Hello World" application to "Hello World" in the middle of the screen in the red color with white background.
2. Create sample application with login module.(Check username and password), validate it for login screen or alert the user with a Toast.
3. Create and validate a login application using username as Email ID else login button must remain disabled.
4. Create and Login application and open a browser with any one search engine.

5. Create an application to display “Hello World” string the number of times user inputs a numeric value. (Example. If user enters 5, the next screen should print “Hello World” five times.)
6. Create spinner with strings from the resource folder (res >> value folder). On changing spinner value, change image.
7. Create an application to change screen color as per the user choice from a menu.
8. Create a background application that will open activity on specific time.
9. Create an application that will have spinner with list of animation names. On selecting animation name, that animation should effect on the images displayed below.
10. Create an UI listing the diploma engineering branches. If user selects a branch name, display the number of semesters and subjects in each semester.
11. Use content providers and permissions by implementing read phonebook contacts with content providers and display in the list.
12. Create an application to call a phone number entered by the user the Edit Text.
13. Create an application that will create database to store username and password.
14. Create an application to insert, update and delete a record from the database.

**Note:** The list is indicative. The Teacher can alter/add more number of experiments as per the requirement.

## Python Lab

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-207-P/IT-312-P<br>*Course Credits: 1<br>Mode: Practical<br>Contact Hours: 2<br>Examination Duration: 03 Hrs:<br><br>*In lab. work one credit is equivalent to two hours | **Course Assessment Methods (internal: 30; external: 70)**<br>An internal evaluation is done by the course coordinator.<br>The end semester practical examination is conducted jointly by external and internal examiners. External examiner is appointed by the COE of the university from the panel of examiners approved by BOSR of the Department of Computer Science and Engineering, Hisar and the internal examiner is appointed by the Chairperson of the Department. |

**Pre-requisites:**

Basic operating skills of Linux Systems, Logic development.

**The objectives of this laboratory course are to:**

1. Apply and test algorithmic plans using simple Python.
2. Identify and track the values of variables representing both scalar and complex types within Python programs
3. Employ repetition, selection, and user defined methods in the design of basic Python Programs
4. Write and use programs that manage a variety of data types and media (numbers, text, pictures, sound etc.)
5. Write object-oriented programs using Python.
6. Exception handling using Python

**By the end of the course a student is expected to:**

CO1. Be fluent in the use of procedural statements — assignments, conditional statements, loops, method calls and arrays.

CO2. Be able to design, code, and test small Python programs that meet requirements expressed in English. This includes a basic understanding of top-down design.

CO3. Understand the concepts of object-oriented programming as used in Python: classes, subclasses, properties, inheritance, and overriding.

CO4. Have knowledge of basic searching and sorting algorithms. Have knowledge of the basics of vector computation.

Students are required to do eight to ten assignments. The lab. assignments are evenly spread over the semester. Every students is required to prepare a file of lab. experiments done.

**Note:** The list is indicative. The Teacher can alter/add more number of experiments as per the requirement.

## Compiler Design

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-401-L/IT-401-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours:4<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:** Discrete Mathematics, Theory of Computation, Data Structure or Equivalent

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are to:**

1. Understand the basic concepts and application of Compiler Design
2. Apply their basic knowledge Data Structure to design Symbol Table, Lexical Analyser, and Intermediate Code Generation, Parser (Top Down and Bottom Up Design) and will able to understand strength of grammar and Programming Language.
3. Understand and Implement a Parser.
4. Understand various Code optimization Techniques and Error Recovery mechanisms

**By the end of the course a student is expected to:**

CO1. Learn the design principles of a Compiler.
CO2. Learn the various parsing techniques.
CO3. Learn different levels of translation
CO4. Learn how to optimize and effectively generate machine codes

**Course Contents**

**Unit I**

**Introduction To Compilers**: Compilers and translators, need of translators, structure of compiler :its different phases, Compiler construction tools.
**Lexical Analysis**: Role of lexical analyzer, design of lexical analyzer, regular expressions, Specification and recognition of tokens, input buffering, A language specifying lexical analyzer. Finite automata, conversion from regular expression to finite automata, and vice versa, minimizing number of states of DFA, Implementation of lexical analyzer.

**Unit II**

Syntax Analysis: Role of parsers, context free grammars, definition of parsing. Parsing Technique: Shift- reduce parsing, operator precedence parsing, top down parsing, predictive parsing.

**Unit III**

LR parsers, SLR, LALR and Canonical LR parser. Syntax Directed Translations: Syntax directed definition, construction of syntax trees, syntax directed translation scheme, implementation of syntax directed translation, three address code, quadruples and triples.

**Unit IV**

Symbol Table & Error Detection and Recovery: Symbol tables, its contents and data structure for symbol tables; trees, arrays, linked lists, hash tables. Errors, lexical phase error, syntactic phase error, semantic error.
Code Optimization & Code Generation: Code generation, forms of objects code, machine dependent code, optimization, register allocation for temporary and user defined variables

T**ext and Reference Books:**


1. Compilers Principle, Techniques & Tools - Alfread V. AHO, Ravi Sethi & J.D. Ullman, Addison Wesley, 2007**.**
2. Theory and practice of compiler writing, Tremblay & Sorenson, Mc. Graw Hill, 1985.
3. System software , Dhamdare, MGH, 1986.
4. Principles of compiler Design, Alfred V. Aho, Jeffrey D. Ullman Narosa Publication, 2002

# Wireless & Mobile Communication

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-403-L/IT-404-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours:4<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**
Basic knowledge of computer networks, Network Architecture and reference model, High Speed Network technologies, Ethernet, TCP/IP architecture.

**About the Course and its Objectives:**

1. Today we have seen sufficient growth in terms of network speed in fixed network but a lot of work is required to be done to improve mobile/wireless communication speed and quality.

2. This course has been designed with an aim that student should learn basis about mobile/wireless communication system, mobile protocols and mobile/wireless communication technologies like WiFi, WiMAX, Bluetooth, LTE, GSM, CDMA.

**Course Outcomes :**

CO1. In-depth knowledge about use and applications of Mobile Communication Technologies
CO2. Design and implementation of Mobile Communication System.
CO3. Study of Wireless LAN such as WiFi, WiMAX, Bluetooth and LTE.
CO4. Knowledge of Network layer protocols for Mobile Communication.
**CO5.** Knowledge of Transport layer protocols for Mobile Communication.

## Course Contents

**Unit I**

Mobile Communication**:** Wireless Transmission--- Frequencies, signals, antennas, signal propagation, multiplexing, modulation, spread spectrum, cellular system. Specialized MAC, SDMA, FDMA, TDMA- fixed TDM, classical ALOHA, slotted ALOHA, CSMA, DAMA, PRMA, reservation TDMA. Collision avoidance, polling inhibit sense multiple access. CDMA, GSM- mobile services, architecture, radio interface, protocol, localization, calling, handover, security, new data services, Introduction to WLL.

**Unit II**

Wireless LAN IEEE 802.11-System and protocol architecture, physical layer. Frame format.

Bluetooth--- Protocol architecture, Frame format.
WiMAX – Layered Protocol architecture, frame types, format, Applications
Introduction to LTE, LTE advanced, VoLTE

**Unit III**

Mobile network Layer: Mobile IP- goals, assumption, requirement, entities, terminology, IP packet delivery, Agent advertisement and discovery, registration, tunneling, encapsulation, optimization , reverse tunneling, IPV6. DHCP. Adhoc Networks—routing , Destination Sequence Distance Vector, dynamic source routing, hierarchical algorithm, alternative metric.

**Unit IV**

Mobile Transport Layer: Traditional TCP, Indirect TCP, Snooping TCP, Mobile TCP fast retransmission/ recovery, transmission/time out freezing, selective retransmission, Transaction oriented TCP.

**Text and Reference Books:**

1 Computer Networks, Andrew S Tanenbaum,, 5th Edition, Pearson, 2013

2 Mobile Communication, Jochen Schiller , 2nd Edition, Pearson, 2009.

3 Mobile Cellular Telecommunications, Lee, McGrawHILL, 2nd Edition, 1995

# Software Project Management

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-405-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4Hours<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 Hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:** Preliminary knowledge of Software Engineering

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are to:**

1. To study how to plan and manage projects at each stage of software development life cycle.
2. To understand successful software projects that support organization's strategic goals.
3. To train the software project managers and other stakeholders in software project planning and project tracking

**By the end of the course a student is expected to:**

1. Understand the basic concepts and issues of Software Project Management.
2. Effectively plan the software projects.
3. Implement the project plans by managing people and communication.
4. Create the project plans and address real world management challenges.
5. Match the organizational needs by the most effective software development model.
6. Conduct the activities necessary to successfully complete the software projects.
7. Develop the skills for tracking and controlling the software projects.

**Course Contents**

**Unit I**

**Introduction to Software Project Management(SPM):** Definition of Software Project, Software Project Vs Other types of projects, activities covered by SPM, categorizing Software Projects, Project as system, management control, requirement specification, information and control in organization.
**Stepwise Project Planning:** Introduction, selecting a project, identifying project scope and objectives, identifying project infrastructure, analyzing project characteristics, identifying the project products and activities, estimate

efforts for each activity, identifying activity risk, allocate resources, review/publicize plan.

**Unit II**

**Project Evaluation and Estimation:** Cost-Benefit analysis, cash flow forecasting, cost benefit evaluation techniques, Selection of an appropriate project, choosing technologies, choice of process models, rapid application development, waterfall model, V process model, spiral model, Albrecht function point analysis.
**Activity Planning:** Objectives of activity planning, project schedule, projects and activities, sequencing and scheduling activities, network planning model, representation of lagged activities adding the time dimension, backward pass and forward pass, identifying the critical path, shortening project, precedence networks.

**Unit III**

**Risk Management:** Introduction, the nature of risk, managing risk, risk identification, risk analysis, reducing the risks, evaluating risks to schedule, calculating z-values**.**
**Resource Allocation:** Introduction, the nature of resources, identifying resource requirements, scheduling resources, creating critical paths, counting the cost, publishing the resource schedule, cost schedule, the scheduling sequence.

**Unit IV**

**Managing Contracts and People:** Introduction, types of contract, stages in contract placement, terms of contract, contract management, acceptance, managing people and organizing teams: Introduction,understanding organization behaviour: a back ground, selecting the right person for job, instruction in best methods, motivation, working in groups, becoming a team, decision making, leadership, organization structures.
**Software Quality:** Introduction, the place of software quality in project planning, the importance of software quality, defining software quality, ISO 9126, Practical software quality measures, product versus process quality management, external standards, techniques to enhance software quality.

T**ext and Reference Books:**


1. Software Project Management, 5th Edition) by Bob Hughes and Mike Cotterell,TMH, 2011
2. Software Project Management, Walker Royce, Addison Wesley, 1998
3. Software Project Management in Practice, Pankaj Jalote, Pearson, 2002
4. Managing Global Software Projects, Ramesh,TMH, 2005.

## Data Warehousing and Data Mining

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-407-L/IT-407-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4 hours /week<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:** Knowledge of database systems, elementary knowledge of statistics and probability.

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are to:**

1. Understand fundamental knowledge of data warehouse and data mining.
2. Understand various data mining techniques/ algorithms.
3. Understand the strength and limitations of underlying data mining techniques.

**By the end of the course a student is expected to:**

CO1. Have knowledge and understanding of data mining process and tasks
CO2. Have understanding of different kinds of data and its handling
CO3. Select and apply a data mining techniques for any particular problem.
CO4. Mine patterns and apply association rules, classify data.

**Course Contents**

**Unit I**

Data Mining: Introduction, Kind of data to be mined, Data Mining Functionalities, Technologies used in Data Mining, Applications of data Mining, Major Issues in Data Mining.

Data Pre-Processing: Introduction, Need of preprocessing, Data Objects and Attribute type, Statistical description of data, Data Visualization, Measuring similarity and dissimilarity of data, data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization

**Unit II**

Data Warehouse: Introduction, Data Warehouse and Database Systems, Data Warehouse Architecture, Data Warehouse Models, Data Cube and OLAP, Multidimensional data Model, Concept Hierarchies, OLAP operations, Data Warehouse Implementation

**Unit III**

Mining Frequent Patterns, Associations and Correlations: Introduction, Frequent Itemset Mining using Apriori Algorithm ,Generating Association Rule from Frequent Itemsets. Improving efficiency of Apriori, Pattern Growth Approach for Mining Frequent Itemsets, Pattern evaluation Methods.

Advanced Pattern Mining: Pattern Mining in Multilevel and Multidimensional Space, Constraint-Based Frequent Pattern Mining.

**Unit IV**

Classification: Introduction, Classification using Decision Tree Induction, Bayesian Classification Methods, Rule Based Classification, Model Evaluation and Selection, Techniques to Improve Classification Accuracy. Classification by Backpropagation, Support Vector Machines, Lazy Learners.

Cluster Analysis: Introduction, Basic Clustering Methods, Partitioning Methods, Hierarchical Methods, Evaluation of Clustering.

T**ext and Reference Books:**


1. Data Mining Concepts and Techniques, Jiawei Han, Micheline Kamber and Jian Pei, Third Edition, Morgan Kaufmann Publishers, July 2011
2. Data Warehousing, Data Mining &Olap, AlexBerson And Stephen J. Smith, Tata Mcgraw – Hill Edition, 2004.
3. Introduction To Data Mining, Pang-Ning Tan, Michael Steinbach and Vipin Kumar, Pearson Education, 2014.
4. Insight Into Data Mining Theory And Practice, K.P. Soman, Shyam Diwakar and V. Ajay, Easter Economy Edition, Prentice Hall Of India, 2009.
5. Introduction To Data Mining With Case Studies, G. K. Gupta, Easter Economy Edition, Prentice Hall Of India, 2006.
6. Data Mining Methods And Models, Daniel T. Larose, Wiley, 2006.
7. Building The Data Warehouse, W.H. Inmon , 4th , Wiley India,2005.

## Advanced Computer Architecture

### General Course Information:

| | |
|---|---|
| Course Code: CSE-409-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4 hours /week<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**

Knowledge of Computer Hardware, Basic Digital Logic and Simple Microprocessors is required.

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are to:**

1. Understand the micro-architectural design of processors
2. Learn about the various techniques used to obtain performance improvement, design architecture of current processors and implementation details

**By the end of the course a student is expected to:**

CO1. Evaluate performance of different architectures with respect to various parameters
CO2. Analyze performance of different ILP techniques
CO3. Identify cache and memory related issues in multi-processors.
CO4. Have knowledge of vector processors.

**Course Contents**

**Unit I**

**Basics:** Review of Fundamentals of CPU, Memory and IO – Trends in technology, power and cost, Dependability – Performance Evaluation.
**Parallel Computer Models:** The state of Computing, Multiprocessors and Multicomputer, Multivector and SIMD Computers, PRAM and VLSI Models

**Unit II**

**Bus, Cache, and Shared Memory:** Backplane Bus Systems, Cache Memory Organizations, Shared-Memory Organizations, Sequential and Weak Consistency Models

**Symmetric and Distributed Shared Memory Architectures**: Performance Issues, Synchronization and Models of Memory Consistency, Case studies: Intel i7 Processor, ARM, SMT and CMP Processors

**Unit III**

**Program and Network Properties:** Conditions of Parallelism, Program Partitioning and Scheduling, Program Flow Mechanism, System Interconnect Architecture,
**Multiprocessor and Multicomputer:**
Multiprocessor System Interconnects, Cache Coherence and Synchronization Mechanisms, Message-Passing Mechanisms.

**Unit IV**

**Processors and Memory Hierarchy:** Advanced Processor Technology, Superscalar and vector Processors, Memory Hierarchy Technology, Virtual Memory Technology.
**Multivector, Scalable, Multithreaded, Data Flow Architecture:** Vector Processing principles, Multivector Multiprocessors, Compound Vector Processing, Principles of Multithreading, Dataflow and Hybrid Architectures.

T**ext and Reference Books:**


1. Advance computer architecture by Kai Hwang MGH, 2010.
2. Pipelined and Parallel processor design by Michael J. Fiynn , Narosa. 1995
3. Computer Architecture and Parallel Processing by Faye A. Briggs, Kai Hwang, McGraw Hill Education, 2012
4. Computer Architecture A Quantitative Approach", Morgan Kaufmann/ Elsevier, 5th Edition, John L Hennessey and David A Patterson, 2012.
5. Advanced Computer Architectures: A Design Space, Sima D, Fountain T and Kacsuk P, 1997

## Compiler Design Lab

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-401-P/IT-401-P<br><br>*Course Credits:<br><br>Mode: Practical<br><br>Contact Hours: 2<br><br>Examination Duration: 03 Hrs:<br><br>*In lab. work one credit is equivalent to two hours | **Course Assessment Methods (internal: 30; external: 70)**<br><br>An internal evaluation is done by the course coordinator.<br><br>The end semester practical examination is conducted jointly by external and internal examiners. External examiner is appointed by the COE of the university from the panel of examiners approved by BOSR of the Department of Computer Science and Engineering, Hisar and the internal examiner is appointed by the Chairperson of the Department. |

**Pre-requisites: C Language**

**The objectives of this laboratory course are to:**

1. Develop the understanding of compiler design.
2. Develop problem solving ability using programming.
3. Develop ability to design and analyse a compiler.

**By the end of the course a student is expected to:**

CO1. Demonstrate a working understanding of the process of lexical analysis, parsing and other compiler design aspects
CO2. Check the Grammars for operator precedence and recursion
CO3. Find terminals and non-terminals
CO4. To show all operations of stack and files

# List of experiments

1. Practice of LEX/YACC of compiler writing.

2. Write a program to check whether a string belong to the grammar or not.

3. Write a program to generate a parse tree.

4. Write a program to find leading terminals.

5. Write a program to find trailing terminals.

6. Write a program to compute FIRST of non-terminal.

7. Write a program to compute FOLLOW of non-terminal.

8. Write a program to check whether a grammar is left Recursion and remove left Recursion.

9. Write a program to remove left factoring.

10. Write a program to check whether a grammar is operator precedent.

11. To show all the operations of a stack.

12. To show various operations i.e. red, write and modify in a text file.

Students are required to do eight to ten assignments. The lab. assignments are evenly spread over the semester. Every students is required to prepare a file of lab. experiments done.

List of Experiments/Practical's (if any)

**Note:** The list is indicative. The Teacher can alter/add more number of experiments as per the requirement.

**Recommended Books:**


1. Introduction to C Programming, Reema Thareja, Oxford University press, 1st Edition 2012.
2. Byron S Gottfriend, "Programming with C", Second Edition, Schaum Out Lines, TATA Mc Graw Hill,2007
3. Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, Monica S. Lam, Compilers: Principles, Techniques, and Tools. Addison-Wesley, 2006.
4. Thomas W. Parsons, Introduction to Compiler Construction. Computer Science Press, 1992.

**Distributed Operating System**

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-402-L/IT-402-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours:4<br><br>Mode: Lecture(L)-3 Tutorials (T)-1<br><br>Examination Duration: 03 Hrs | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:** Knowledge of operating system , computer networks and a programming language

**About the Course and its Objectives & Outcomes:**

This course focus on study of distributed system concepts and its applicability. In this course various advantages of distributed computing system are studied. After studying this subject a student will be expected to understand the design issues of distributed systems and propose problem specific solution.

**The objectives of this course are to:**

1. To understand the basics of distributed operating system
2. To understand the concepts of synchronization and related issues in distributed systems
3. To study process management in distributed systems
4. To understand the issues related to memory management and file handling and associated solutions in distributed systems

**By the end of the course a student is expected to:**

CO1. A student will be expected to understand the benefits of distributed systems over simple client server based computer networks and find solutions to the problems inherent in distributed systems

CO2. Study the synchronization in distributed environment and deadlock avoidance

CO3. Scheduling in distributed environment of processes and processors

CO4. Distributed File systems and distributed shared memory

**Course Contents**

**Unit I**

Introduction: Introduction to Distributed System, Goals of Distributed system, Hardware and Software concepts, Design issues. Communication in distributed system: Layered protocols, ATM networks, Client – Server model ,Remote Procedure Calls and Group Communication. Middleware and Distributed Operating Systems.

**Unit II**

Synchronization in Distributed System: Clock synchronization, Mutual Exclusion, Election algorithm, the Bully algorithm, a Ring algorithm, Atomic Transactions, Deadlock in Distributed Systems, Distributed Deadlock Prevention, Distributed Deadlock Detection .

**Unit III**

Processes and Processors in distributed systems: Threads, System models, Processors Allocation, Scheduling in Distributed System, Real Time Distributed Systems.

**Unit IV**

Distributed file systems: Distributed file system Design, Distributed file system Implementation, Trends in Distributed file systems.

Distributed Shared Memory: What is shared memory, Consistency models, Page based distributed shared memory, shared variables distributed shared memory.

T**ext and Reference Books:**


1. Distributed Operating Systems: Concepts and Design, Pradeep K Sinha, Prentice Hall of India, 2007.
2. Distributed Systems: Principles and Paradigms, Tanenbaum A.S., Van Steen M., , Pearson Education, 2007.
3. Distributed Computing, Principles and Applications, Liu M.L., , Pearson Education, 2004.
4. Distributed Algorithms, Nancy A Lynch, Morgan Kaufman Publishers, USA, 2003.

## Information And Cyber Security

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-404-L/IT-403-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3hrs | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:** Basic knowledge of Number theory, Complexity Theory, Basic programming skills for security.

**About the Course and its Objectives & Outcomes:**

The increase in techniques to penetrate into systems has led to variety of information and cyber attacks, To mitigate the exploitation of the vulnerabilities leading to these attacks we need to adopt robust security architecture into our premises. We have to choose between various security technologies such as cryptography, Digital Signatures, Key Management, Program Security, Database security, cyber laws, Wifi security. In the current scenario we require to secure end-to-end devices, Networks, Networking devices, clouds.

**The objectives of this course are to:**

1. To understand several security issues.
2. To understand several cryptographic algorithms.
3. To understand the social legal and ethical implications of modern security systems.
4. To get aquainted with the latest trends in cyber security.

**By the end of the course a student is expected to:**

CO1. Have a basic Conceptual Knowledge of security aspects involved in information systems.
CO2. Get known to various Cryptographic Algorithms and other security technologies
CO3. Learn cyber Attacks and cyber Laws to protect against miscreants in the public Networks.
CO4. Apply tools to analyze contents of packet on local Area Network

**Course Contents**

**Unit I**

**C**ryptography: Overview of Information Security,Basic Concepts, Cryptosystems, Cryptoanalysis, Ciphers &Cipher modes,Symmetric Key CryptographyDES, AES.Asymmetric Key Cryptography, RSA algorithm, Key management protocols,Diffie Hellman Algorithm. Digital Signature-Digital Signatures, Public Key Infrastructure.

**Unit II**

System Security: Program Security, Security problems in Coding, Malicious Logic, Protection. Database Security-Access Controls, Security & Integrity Threats, Defence Mechanisms. OS Security-Protection of System Resources, Models for OS security, .Net Security-User based security, Code access security, form authentication.

**Unit III**

Ethics in Cyber Security:Privacy, Intellectual Property in the cyberspace, Professional Ethics, Freedom of Speech, Fair User and Ethical Hacking, Trademarks, Internet Fraud, Electronic Evidence, forensic Technologies, Digital Evidence collections.
Tools and Methods Used in Cybercrime: Introduction, Proxy Servers and Anonymizers, Phishing,Password Cracking, Keyloggers and Spywares, Virus and Worms, Phishing and Identity Theft, Trojan Horses and Backdoors, Steganography, DoS and DDoS Attacks, SQL Injection, Buffer overflows

**Unit IV**

Cybercrimes and Cybersecurity: Cybercrime and Legal Landscape around the world, Cyberlaws, The Indian IT Act, Challenges, Digital Signatures and Indian IT Act, Amendments to the Indian IT Act, Cybercrime and punishment, Cost of Cybercrimes and IPR Issues, Web threats for Organizations, Social Computing and associated Challenges for Organizations.

T**ext and Reference Books:**

1.Cryptography and Network security-Principles and Practices, Pearson Education, Ninth Indian Reprint 2005

2.Charlie Kaufman , Network Security : Private communication in Public World, Prentice-Hall International, Inc. April 2008

3. Cyber Security by Nina Godhole, Sunit Belapure, Wiley India, 2011.

4.Cyber Security Essentials by James Graham, Ryan Olson, Rick Howard CRC Press, Taylor & Francis, 2011.

## Cloud Computing

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-406-L/IT-406-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4<br><br>Mode: Lecture (L)<br><br>Examination Duration: 3 Hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:** Basics of Computing.

**About the Course and its Objectives & Outcomes:**

This course gives an introduction to cloud computing and its techniques, issues, services and securities that will lead to design and development of a small cloud service.

**The objectives of this course are:**

1. To analyse cloud computing components and its business perception.
2. To know the latest Trends in Cloud.
3. To collaborate with real time cloud services.
4. To analyse the case studies to derive the best practice model to apply while developing and deploying cloud applications.

**By the end of the course a student is expected to:**

CO1. Understand the cloud system and its business model.
CO2. Have the knowledge of various cloud development tools.
CO3. Know how to work together with real time cloud services.
CO4. Design and develop small cloud applications.

**Course Contents**

**Unit I**

**Overview of Cloud Computing-** Cloud at a Glance: The Vision of Cloud Computing, Defining a Cloud, Cloud Computing Reference Model, Characteristics and Benefits, Historical Developments: Distributed Systems, Virtualization, Web 2.0, Service-Oriented Computing, Building Cloud Computing Environment: Application

Development, Infrastructure and System Development, Computing Platforms and Technologies: Amazon Web Services, Google AppEngine, Microsoft Azure, Hadoop, Force.com and Salesforce.com

**Unit II**

**Virtualization & Cloud Computing Architecture –** Introduction, Characteristics of Virtualized Environments, Taxonomy of Virtualization Techniques: Execution Virtualization, Other Types of Virtualization, Virtualization and Cloud Computing: Pros and Cons of Virtualization, Technology Examples: Xen: Paravirtualization, VMware: Full Virtualization, Microsoft Hyper – V, Cloud Architecture: Introduction, Cloud Reference Model Architecture, Infrastructure as a Service, Platform as a Service, Software as a Service, Types of Clouds: Public, Private, Hybrid, Community.

**Unit III**

**Cloud in Industry and Its Applications –** Amazon Web Services: Compute Services, Storage Services, Communication Services, Additional Services, Google AppEngine: Architecture and Core Concepts, Application Life-Cycle, Microsoft Azure: Core Concepts, SQL Azure, Windows Azure Platform Appliance, Cloud Applications: Scientific Applications: Healthcare: ECG, Biology: Protein Structure Preduction, Gene Expression Data Analysis for Cancer Diagnosis, Geo-Science Satellite Image Processing, Business and Consumer Applications: CRM and ERP, Social Networking, Media Applications, Multiplayer Online Gaming.

**Unit IV**

**Security in Cloud –** Cloud Information Security Fundamentals, Cloud Security Services, Design Principles, Secure Cloud Software Requirements, Policy Implementation, Cloud Computing Security Challenges, Virtualization Security Management, Cloud Computing Security Architecture.

**Text and Reference Books and Links:**


1. Rajkumar Buyya, Christian Vecchiola and S. Thamarai Selvi, "Mastering Cloud Computing," McGraw Hill Publication (India) Private Limited, 2013 (ISBN 978-1-25-902995-0).
2. Krutz, Vines, "Cloud Security," Wiley Publication.
3. Bloor R., Kanfman M., Halper F. Judith Hurwitz "Cloud Computing for Dummies," (Wiley India Edition),2010.
4. John Rittinghouse & James Ransome, "Cloud Computing Implementation Management and Strategy," CRC Press, 2010.
5. Antohy T Velte ,Cloud Computing : "A Practical Approach," McGraw Hill,2009.
6. Rajkumar Buyya, James Broberg and Andrez Gosscinski, "Cloud Computing: Principles and Paradigm," John Wiley and Sons, Inc. 2011

## Cloud Computing Lab

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-406-P/IT-406-P<br><br>Course Credits: 1<br><br>Contact Hours: 2<br><br>Mode: Practical<br><br>Examination Duration: 3 Hours | **Course Assessment Methods (internal: 30; external: 70)**<br><br>An internal evaluation is done by the course coordinator.<br><br>The end semester practical examination is conducted jointly by external and internal examiners. External examiner is appointed by the COE of the university from the panel of examiners approved by BOSR of the Department of Computer Science and Engineering, Hisar and the internal examiner is appointed by the Chairperson of the Department. |

**Pre-requisites:** Basic knowledge of java, C#,

**The objectives of this laboratory course are to:**

1. Objective of this module is to provide students an overview of the Cloud Computing and architecture and different types of cloud computing.
2. Learn the Virtualization Basics, Objectives of Virtualization, and Benefits of Virtualization in cloud

**By the end of the course a student is expected to:**

CO1. Be exposed to tool kits for cloud environment.
CO2. Be familiar with developing web services/Applications in cloud framework.
CO3. Learn to run virtual machines of different configuration.
CO4. Learn to use Aneka.

**Software and Tools to be learnt:-** Aneka, cloudsim.

**List of Experiments**

1. Working of Goggle Drive to make spreadsheet and notes.
2. Installation and Configuration of Hadoop/Eucalyptus
3. Working and installation of Google App Engine
4. Working and installation of Microsoft Azure
5. Working with Mangrasoft Aneka Software

**Object Oriented Systems Development**

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-414-L/IT-414-L<br><br>Course Credits: 3.5<br><br>Type: Prog Elective II<br><br>Contact Hours: 4<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**

Software Engineering or Equivalent.

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are to:**

1. This graduate course is intended to provide an in depth understanding of object oriented approaches to software development
2. Learn the UML design diagrams.
3. Learn to map design to code.
4. The analysis and design phases of the software life cycle

.

**By the end of the course a student is expected to:**

CO1. Design and implement projects using OO concepts.
CO2. Use the UML analysis and design diagrams.
CO3. Apply appropriate design patterns.
CO4. Create code from design.

**Course Contents**

**Unit I**

**Introduction:** Introduce the concepts of object-orientation, object-oriented analysis and design, Unified Modeling Language (UML). In addition, the concepts of software development process and activities, and Unified Development Process are also introduced. A case study is used to illustrate the overview of object-oriented analysis and design with UML.

Review of the Traditional Methodologies, Advantages of Object Oriented, Methodologies over Traditional Methodologies, Classes, Objects, Encapsulation, Association, Aggregation, Inheritance, Polymorphism, States and Transitions.

**Unit II**

**Static Design Model** : Design system static model, including design class diagram, identification of classes, attributes and methods, identification of generalization, aggregation, composition, and dependency relations, Naming Relationships, Role Names ,Defining associations with multiplicities and constraints by illustrating with case studies**.**

**Checking the Model**: Making the Model Homogeneous, Combining Classes, Splitting Classes, Eliminating Classes, Consistency Checking, Scenario Walk-through, Event Tracing, Documentation Review, Designing the System Architecture: The need for Architecture, The"4+1" view of Architecture, The Logical view, The Component View, The Process View, The Deployment View, The Use Case view.

**Unit III**

**Use Case Model**: Analyze and specify the requirements model, including use case diagram, use case definition, system operation sequence diagram, activity diagram, and conceptual class diagram by illustrating with case studies**.** Design system dynamic model, including design sequence diagram, activity diagram and state diagram, Package, component and Deployment diagrams mapping design to codes by illustrating with case studies

**Unit IV**

**Software Development Life Cycle:** Inception -Use case Modeling - Relating Use cases – include, extend and generalization - Elaboration - Domain Models - Finding conceptual classes and description classes – Associations – Attributes – Domain model refinement – Finding conceptual class Hierarchies - Aggregation and Composition

**The Iteration Planning Process**: Benefits, Goals, Design the User Interface, Adding Design Classes, The Emergence of Patterns, Designing Relationships, Designing Attributes and Operations, Designing for Inheritance, Coding, Testing, and Documenting the Iteration.

T**ext and Reference Books:**


1. UML User Guide, Grady Booch, James Rumbaugh, Ivar Jacobson, Addison Wesley ,2005.
2. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development, Craig Larman, Third Edition, Pearson Education, 2005.
3. Object Oriented Systems Analysis and Design Using UML, ,Simon Bennett, Steve Mc Robb and Rayarmer Fourth Edition, Mc-Graw Hill Education, 2010.
4. UML Distiled, Maxtin Fowler with Kendall Scott, Second Edition, 2000
5. Sams Teach Yourself UML In 24 Hours , Joseph Schmuller ,2000

**Embedded System Design**

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-416-L/IT-416-L<br><br>Course Credits: 3.5<br><br>Type:<br><br>Contact Hours: 4<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:** students should have knowledge of microprocessors and operating systems.

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are to:**

1. Design embedded computer system hardware.
2. Design, implement, and debug multi-threaded application software that operates under real-time constraints on embedded computer systems
3. Describe the implementation of a real -time operating system on an embedded computer system

**By the end of the course a student is expected to:**

CO1. Ability to understanding of general system theory and how this applies to embedded system.

CO2. Ability to build a prototype circuit on breadboard using 8051 microcontroller.

CO3. Study communication with 8051 using RS232C Connections

CO4. Interfacing 8051 to other devices like stepper motor, ADC etc.

**Unit I**

Introduction to an embedded systems design & RTOS: Introduction to Embedded system, Processor in the System, Microcontroller, Memory Devices, Embedded System Project Management, ESD and Co-design issues in System development Process, Design cycle in the development phase for an embedded system, Use of target system or its emulator and In-circuit emulator, Use of software tools for development of an ES.

Inter-process Communication and Synchronization of Processes, Tasks and Threads, Problem of Sharing Data by Multiple Tasks, Real Time Operating Systems: OS Services, I/O Subsystems, Interrupt Routines in RTOS Environment, RTOS Task Scheduling model, Interrupt Latency and Response times of the tasks.

**Unit II**

Overview of Microcontroller: Microcontroller and Embedded Processors, Overview of 8051 Microcontroller family: Architecture, basic assembly language programming concepts, The program Counter and ROM Spaces in the 8051, Data types, 8051 Flag Bits ad PSW Register, 8051 Register Banks and Stack Instruction set, Loop and Jump Instructions, Call Instructions, Time delay generations and calculations, I/O port programming Addressing Modes, accessing memory using various addressing modes, Arithmetic instructions and programs, Logical instructions, BCD and ASCII application programs, Single-bit instruction programming, Reading input pins vs. port Latch, Programming of 8051 Timers, Counter Programming

**Unit III**

Communication with 8051**:** Basics of Communication, Overview of RS-232, $I^2C$ Bus, UART, USB, 8051 connections to RS-232, 8051 serial communication programming, 8051 interrupts, Programming of timer interrupts, Programming of External hardware interrupts, Programming of the serial communication interrupts, Interrupt priority in the 8051

**Unit IV**

Interfacing with 8051**:** Interfacing an LCD to the 8051, 8051 interfacing to ADC, Sensors, Interfacing a Stepper Motor, 8051 interfacing to the keyboard, Interfacing a DAC to the 8051, 8255 Interfacing with 8031/51, 8051/31 interfacing to external memory.

Text and reference books:


1. Embedded Systems, Raj Kamal, TMH, 2004.
2. The 8051 Microcontroller and Embedded Systems, M.A. Mazidi and J. G. Mazidi, PHI, 2004.
3. An Embedded Software Primer, David E. Simon, Pearson Education, 1999.
4. The 8051 Microcontroller, K.J. Ayala, , Penram International, 1991.
5. 8051 Microcontroller & Embedded Systems, Dr. Rajiv Kapadia, Jaico Press
6. Embedded Real Time System, Dr. Prasad, Wiley Dreamtech, 2004.
7. Design with PIC Microcontrollers, John B. Peatman, Pearson Education Asia, 2002
8. Computers as components: Principles of Embedded Computing System Design Wayne Wolf, Morgan Kaufman Publication, 2000
9. The Design of Small-Scale embedded systems, Tim Wilmshurst, Palgrave, 2003
10. Embedded System Design, Marwedel, Peter, Kluwer Publishers, 2004

# Digital Image Processing

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-418-L/IT-418-L<br><br>Course Credits: 3.5<br><br>Type: Prog Elective II<br><br>Contact Hours: 4<br><br>Mode: Lecture and Tutorial<br><br>Examination Duration: 3 | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**
Students are expected to have knowledge in basic linear algebra, basic probability theory and basic programming techniques, Fourier Transform.

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are:**:

1. To make them understand a theoretical foundation of digital image processing concepts.
2. To provide mathematical foundations for digital manipulation of images, image acquisition, preprocessing, enhancement, segmentation, and compression.
3. To make them learn algorithms that perform basic image processing operations (e.g., histogram processing, noise removal and image enhancement and restoration);
4. To make them learn algorithms for image analysis (e.g., image compression, image segmentation and image representation.

**By the end of the course a student is expected to**:

CO1. Describe two-dimensional signal acquisition, sampling, and quantization.
CO2. Apply mathematical functions for digital manipulation of images such as image acquisition, preprocessing, segmentation, compression and representation.
CO3. Explain the image enhancement in the spatial domain and frequency domain.
CO4. Design and implement algorithms for digital image processing operations such as histogram equalization, filtering, enhancement, restoration and denoising, segmentation, compression.

## Course Contents

**Unit I**

Introduction and fundamental to digital image processing: What is digital image processing, Origin of digital image processing, Examples that use digital image processing, Fundamental steps in digital image processing, Components of digital image processing system, Image sensing and acquisition, Image sampling, Quantization and representation, Basic relationship between pixels.

Image enhancement in spatial domain and frequency domain: Background, Basic gray level transformation, Histogram processing, Basics of spatial filtering, Smoothing and sharpening spatial filters, Introduction to Fourier transform and the frequency domain, discrete fourier transform, Smoothing and sharpening frequency domain filters.

**Unit II**

Image Restoration: Image degradation/restoration Process, Noise models, Restoration in presence of noise, Inverse filtering, Minimum mean square filtering, Geometric mean filter, Geometric transformations.
Color Image Processing: Color fundamentals, Color models, Basics of full color image processing, Color transformations.

**Unit III**

Image Compression: Fundamentals, Image compression models, Error free compression, Lossy compression.
Image Segmentation: Detection of discontinuities, Edge linking and boundary detection, Thresholding, Region based segmentation.

**Unit IV**

Representation and Description and Recognition: Representation-chain codes, polygonal approximation and skeletons, Boundary descriptors-simple descriptors, shape numbers, Regional descriptors- simple, topological descriptors.
Recognition: Pattern and Pattern classes, Decision theoretic models.

**Text and Reference Books:**


1. Digital Image Processing, Rafael C. Gonzalez and Richard E. Woods,, Pearson Education,Ed, 2001.
2. Fundamentals of Digital Image Processing, Anil K. Jain, Pearson Education, PHI, 2001.
3. Image Processing-Principles and Applications, Tinku Acharya and Ajoy K. Ray, John Wiley & Sons, Inc., 2005.
4. Digital Image Processing and Analysis, Chanda and D. Dutta Majumdar, PHI, 2003.
5. Image Processing, Analysis, and Machine Vision, Milan Sonka, Vaclav Hlavac, Roger Boyle, ,Brookes/Cole, PWS Publishing Company, Thomson Learning, 2nd edition, 1999.

# Network Programming

**General Course Information:**

| | |
|---|---|
| Course Code: IT-420-L/IT-420-L<br>Course Credits: 3.5<br>Type: Compulsory<br>Contact Hours: 4<br>Mode: Lecture(L)<br>Examination Duration: 3 Hours | **Course Assessment Methods (internal: 30; external: 70)**<br>Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**

Java programming, Programming Techniques.

**About the Course and its Objectives & Outcomes:**

The course is an introduction to programming applications that use computer networks. The focus is on problem solving with emphasis on network programming. The operation and characteristics of major computer networks are studied because of their strong influence on programming interfaces (APIs) and application design.

**The objectives of this course are to:**

On successful completion of this course you should be able to :-

1. Analyse the requirements of a networked programming environment and identify the issues to be solved.
2. Create conceptual solutions to those issues and implement a programming solution.
3. Understand the key protocols that support the Internet.
4. Apply several common programming interfaces to network communication.
5. Understand the use of TCP/UDP Sockets.
6. Apply advanced programming techniques such as Broadcasting, Multicasting etc.

**By the end of the course a student is expected to:**

CO1. Understand the addressing schemes of hosts on various Networks.
CO2. Create sockets for TCP and UDP on Client- Server paradigm.
CO3. Apply algorithms for iterative connection oriented servers and iterative connectionless servers for TCP and UDP.
CO4. To get versatile in RPC models for client and server.

**Course Contents**

**Unit I**

Introduction to UNIX O.S and networking, Inter process communication of TC/IP Protocol and its architecture, Classful internet addresses, CIDR(Classless Inter Domain routing) ,VLSM (Variable Length subnet masking),super netting & aggregation , address resolution Protocol (ARP) and RARP, IP datagram format, UDP and TCP , ICMP its purpose , NET STAT details & IP config.

**Unit II**

Socket introduction, elementary TCP sockets, TCP client sever, I/O functions, select & poll functions, socket options elementary UDP sockets, elementary node and address conversions,echo service (TCP and UDP),Routing sockets,broadcasting to mobile network, data link access, debugging techniques .

**Unit III**

Algorithm and issues in server software design :iterative connectionless servers, (UDP), Iterative,connection oriented servers (TCP), single process, concurrent servers multi protocol servers (TCP,UDP),
multi service servers (TCP,UDP).

**Unit IV**

Remote procedure call concept (RCP) : RPC models, analogy between RPC of client and server,remote programs and procedures, their multiple versions and mutual exclusion communication semantics, RPC retransmits, dynamic port mapping ,authentication.

**Text and reference books:**


1. Unix Network programming, W.Richard Stevens Vol -2 2nd edition, Addison-Wesley, 2003
2. Internetworking with TCP/IP Vol-1, Doubles E-commer. 5Th edition, Prentice Hall, 2005
3. Unix Network Programming – The Sockets Networking API, W. Richard Stevens, B. Fenner , A.M. Rudoff , 3rd edition, Pearson ,2004.
4. Internetworking with TCP/IP Vol. II, Comer, Douglas E.; Stevens, David L., 3rd Edition, Prentice Hall, 2004
5. Internetworking with TCP/IP Vol III, Doubles E comer, David L.Stevens, Prentice Hall, 1993

# Software Testing

**General Course Information**

| | |
|---|---|
| Course Code: CSE-422-L/IT-422-L<br><br>Course Credits: 3.5<br><br>Type: Prog Electiv<br><br>Contact Hours: 04<br><br>Mode: Lecture(L)<br><br>Examination Duration: 03 hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:** Basic knowledge of Software Process, SRS, Software Development Life Cycle and Software Models.

**About the Course and its Objectives & Outcomes:**
This course covers software testing in the context of testing concepts and methods that can be implemented in practice. The course will introduce basics of software testing, types of software testing, software verification, Selection, Minimization and Prioritization of Test Cases for Regression Testing, Software Testing Activities, Object Oriented Testing and finally Metrics and Models in Software Testing.

**The objectives of this course are to:**

1. To help students in finding defects which may get created by the programmer while developing the software.
2. Gaining confidence in and providing information about the level of quality and to prevent defects, thereby increasing overall quality of software.

**By the end of the course a student is expected to:**

CO1. understand the importance of software testing and design test cases without considering internal structure of the program.
CO2. be proficient in control flow testing, dataflow testing, slice based testing, mutation testing and verification testing.
CO3. be able to select, minimize and prioritize test cases for regression testing and familiar with other software testing activities, debugging approaches and testing tools along with their commercial applications.
CO4. understand the importance of Object Oriented Testing and Software Metrics used in software testing with their practical application in predicting models using commercial data sets.

**Course Contents**

**Unit I**

**Introduction:** Some Testing .Failures, Testing Process, Some Terminologies, Limitations of Testing and V-Shaped Software Life-Cycle Model.

**Functional Testing:** Boundary Value Analysis, Equivalence Class Testing, Decision Table Based Testing, and Cause Effect Graphing Technique.

**Unit II**

**Structural Testing**: Control Flow Testing, Data Flow Testing, Slice Based Testing and Mutation Testing.
**Software Verification:** Verification Methods, Software Requirement Specification Document Verification, Software Design Description Document Verification, Source Code Reviews, User Documentation Reviews and Software Project Audit.

**Unit III**

**Selection, Minimization and Prioritization of Test Cases for Regression Testing:** Regression Testing, regression Test Case Selection, Reducing the Number of Test Cases, Risk Analysis and Code Coverage Prioritization Techniques.
**Software Testing Activities:** Levels of Testing, Debugging, Software Testing Tools, Software Test Plan.

**Unit IV**

**Object Oriented Testing:** Object Orientation, Object Oriented Testing, Path Testing, State Based Testing and class testing.
**Metrics and Models in Software Testing:** Software Metrics, Categories of Metrics, Object Oriented Metrics in Testing, What Should We Measure During Testing? and Software Quality Attributes Prediction Models.

T**ext and Reference Books:**
1. Software Testing, . Yogesh Singh, Cambridge University Press, 2012.
2. Effective Methods for Software Testing, William E. Perry, John Wiley and Sons, 2002.
3. Software Testing: Principle, Techniques and Tools, M. G. Limaye, Tata McGraw Hill, 2009.
4. Software Engineering, K. K. Aggarwal and Yogesh Singh, New Age International Publishers, Third Edition, 2008.
5. The Art of Software Testing, Glenford J.Myers,Tom Badgett and Corey Sandler, Wiley & Sons, Third Edition, 2012.

## Ubiquitous Computing

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-424-L/IT-424-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours:4<br><br>Mode: Lecture(L)-3 Tutorials (T)-1<br><br>Examination Duration: 03 Hrs | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**

Knowledge of operating system , computer networks and a programming language

**About the Course and its Objectives & Outcomes:**

This Course focus on study of ubiquitous computing system concepts and its applicability. In this course the ideas of ubiquitous computing techniques based on human experience is introduced. After studying this subject a student will be expected to design, analyze on real life problems and perform experiments using various smart devices, smart interaction and smart environment.

**The objectives of this course are to:**

1. To introduce the ideas of ubiquitous computing techniques based on human experience.
2. To generate an ability to design, analyze and perform experiments on real life problems using various smart devices, smart interaction and smart environment.
3. To integrate computation into the environment, rather than having computers as distinct objects.
4. To enable people to move around and interact with computers more naturally than they currently do.

**By the end of the course a student is expected to have knowledge and understanding regarding:**

CO1. The objectives and the historical development of the field of ubiquitous computing
CO2. Fundamentals of sensor technology and sensor networks
CO3. Apply middleware techniques to implement ubiquitous computing systems
CO4. Design of new (often embedded) interactive artifacts
CO5. Context aware and adaptive systems
CO6. Compare the usability of alternative design of interactions for specific ubiquitous computing systems

**Course Contents**

**Unit I**

Introduction to Ubiquitous Computing: Definition, Advantage, Application and Scope. Properties of Ubiquitous Computing, Ubiquitous System Environment Interaction, Architectural Design for UbiCom Systems: Smart DEI model.

Smart Devices and Services: Introduction to Smart Devices : Users, Mobiles, Cards and Device Networks. Service Architecture Models, Service Provision Life-Cycle. Virtual Machines and Operating Systems Mobile Computers and Communicator Devices.

**Unit II**

Sensing and Controlling : Tagging the Physical World. Sensors and Sensor Networks. Micro Actuation and Sensing: Micro-Electro-Mechanical Systems (MEMS). Embedded Systems and Real-Time Systems. Control Systems for Physical World Tasks. Robots

Context-Aware Systems : Introduction to Context-Aware Computing, Context- Aware Systems, Context-Aware Applications, Designing and Implementing Context-Aware Applications, Issues for building Context-Aware Applications.

**Unit III**

Human–Computer Interaction: User Interfaces and Interaction for Four Widely Used Devices. Hidden UI Via Basic Smart Devices. Hidden UI Via Wearable and Implanted Devices. Human- Centered Design (HCD). User Models: Acquisition and Representation. iHCI Desi

**Unit IV**

Ubiquitous Communication: Data Networks. Audio Networks. Wireless Data Networks. Universal and Transparent Audio, Video and Alphanumeric Data. Ubiquitous Networks. Network Design Issues. Human Intelligence Versus Machine Intelligence. Challenges in Ubiquitous System, Social Issues: Promise Versus Peril.

T**ext and Reference Books:**


1. Ubiquitous Computing: Smart Devices, Environments and Interactions, Stefan Poslad. Wiley Publication, 2009
2. Ubiquitous Computing Fundamentals , John Krumm.. CRC Press, 2009
3. Ubiquitou.s Computing: Design, Implementation, and Usability,Yin-Leng Theng and Henry B. L. Duh. IGI Global, 2008
4. Everyware the Dawning age of Ubiquitous Computing,Adam Greenfield. Published in Association with AIGA, 2006
5. Mobile and Ubiquitous Computing", Georgia Tech, 2003.

## Machine Learning

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-426-L/IT-426-L<br><br>Course Credits: 4<br><br>Type: Elective<br><br>Contact Hours: 4<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:** Basics of Linear Algebra and Statistics, Basics of Probability Theory, Data Structures and Computer Algorithms.

**About the Course and its Objectives & Outcomes:**

Machine learning is the study of computer algorithms that improve their performance through experience. Machine learning draws its conceptual foundation from the fields like artificial intelligence, probability and statistics, computational complexity, cognitive science, biology and information theory etc. The course introduces some of the key machine learning algorithms and the theory that form the backbone of these algorithms. The examples of such algorithms are classification algorithms for learning patterns from data, clustering algorithms for grouping objects on the basis of similarity, neural network algorithms for pattern recognition, genetic algorithms for searching large and complex search spaces etc.

**The objectives of this course are to:**

1. introduce students to the basic concepts in the domain of machine learning.
2. make students understand a range of machine learning algorithms.
3. develop skills of applying machine learning algorithms to real world problems.

**By the end of the course a student is expected to:**

CO1. appreciate the use of machine learning algorithms.
CO2. apply decision tree, instance based learning and clustering algorithms on toy problems or problems with moderate complexity.
CO3. design Genetic Algorithms for optimization and search problems.
CO4. apply Neural Networks for pattern recognition problems.
CO5. apply Bayesian framework for predicting class probabilities.

CO6. examine issues related to sampling and error estimation.

**Course Contents**

**Unit I**

**Introduction**: Well posed learning problems, Designing a learning system, Issues in machine learning, The concept learning task, Concept learning as search, Finding a maximally specific hypothesis, Version spaces and candidate elimination algorithm, Remarks on version spaces and candidate-eliminations, Inductive bias.

**Unit II**

**Decision Tree Learning:** Decision tree representation, Appropriate problems for decision tree learning, The basic decision tree learning algorithm, Hypothesis space search in decision tree learning algorithm, Inductive bias in decision tree learning, Issues in decision tree learning.
**Instance based learning**: k-nearest neighbour learning, distance-weighted nearest neighbour algorithm.
**Clustering**: Clustering analysis, type of data in clustering analysis, k-means and k-medoids, CLARAN clustering algorithms, Hierarchical agglomerative and divisive clustering methods, DBSCAN density based clustering method.

**Unit III**

**Artificial Neural networks**: Neural Network representations, Appropriate problems for neural network learning, Perceptron. The perceptron training rule, Gradient descent and delta rule, Multilayer Networks and back propagation algorithm.
**Genetic Algorithms**:
Basic terminology related to Genetic Algorithms, Working and block diagram of a genetic algorithm, Representing individual solutions or hypotheses, Genetic operators, Fitness functions and selection, Illustrative examples of genetic algorithms in function optimization and knapsack problem. Population and schema theorem.

**Unit IV**

**Bayesian Learning**: Bayes theorem, Bayes theorem and concept learning, Maximum likelihood and least-squared error hypotheses, Naïve Bayes Classifier.
**Evaluating Hypotheses**: Estimating hypothesis Accuracy, Basics of sampling theory, Error estimation and estimating Binomial proportions, The binomial distribution, Mean and variance, Bias and variance, Confidence intervals, Two sided or one sided bounds, Central limit theorem, Hypothesis testing, Comparing learning algorithms

T**ext and Reference Books:**


1. Machine Learning , Tom M. Mitchell, McGraw-Hill, 1997.
2. Pattern Recognition and Machine Learning, Bishop Christopher, Springer Verlag, 2006.
3. The Elements of Statistical Learning: Data Mining, Inference and Prediction, Trevor Hastie, Robert Tibshirani, Jerome Friedman, Springer, 2nd edition, 2009..
4. Data Mining Concepts and Techniques, J. Han and M. Kamber, 3rd Edition, Elsevier, 2012.
5. Genetic Algorithms +Data Structures = Evolutionary Program, Z. Michalewicz, 3rd Edition, Springer-Verlag, 1996.
6. Neural Networks, Fuzzy Logic and Genetic Algorithm: Synthesis and Applications, S. Rajeshkaran, G. A. Vijayalakshmi Pai, PHI, 2003.

## Bioinformatics

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-428-L/IT-428-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4<br><br>Mode: Lecture (L)<br><br>Examination Duration: 3 Hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Prerequisites**

Basic knowledge of databases, biology

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are:**

1. To give a basic introduction to molecular biology.
2. To introduce students to the field of bioinformatics.
3. To explain proteins, DNA, and RNA.
4. Explain the role of DNA and proteins in human disease and therapies.
5. To explain analysis of genomic and proteomic data.

**By the end of the course a student is expected to:**

CO1. Be aware of basic terminologies used in the field of Bioinformatics.
CO2. Be aware of databases related to Bioinformatics and able to comprehend data in these databases.
CO3. Be able to perform sequence alignment and analysis using software tools.
CO4. Be able to apply computational techniques and prediction algorithms to solve problems related to the domain of Bioinformatics.

**Unit I**

Introduction to Basic Cell Chemistry: Cell chemistry and macromolecules. Biochemical pathways e.g. Glycolysis. Protein structure and functions.

Cell Structure and Function: Cell components. Different types of cell. Chromosome structure and organisation. Cell division.

The Hereditary Material: DNA structure, replication and protein synthesis. Structure and roles of RNA. Genetic code. Mechanism of protein synthesis: transcription and translation. Mutation.

Recombinant DNA Technology: Restriction enzymes. Hybridisation techniques. Gene cloning. Polymerase chain reaction.

Genomics and Structural Genomics: Genes, genomes, mapping and DNA sequencing.

**Unit II**

Introduction to bioinformatics: Definitions, Sequencing, Biological sequence/structure, Genome Projects, Pattern recognition and prediction, Folding problem, Sequence Analysis, Homology and Analogy. Protein Information Resources: Biological databases, Primary sequence databases, Protein Sequence databases, Secondary databases, Protein pattern databases, and Structure classification databases.

**Unit III**

Genome Information Resources: DNA sequence databases, specialized genomic resources DNA Sequence analysis: Importance of DNA analysis, Gene structure and DNA sequences, Features of DNA sequence analysis, EST (Expressed Sequence Tag) searches, Gene hunting, Profile of a cell, EST analysis, Effects of \EST data on DNA databases.

**Unit IV**

Pair wise alignment techniques: Database searching, Alphabets and complexity, Algorithm and programs, Comparing two sequences, sub-sequences, Identity and similarity, The Dotplot, Local and global similarity, different alignment techniques, Dynamic Programming, Pair wise database searching. Multiple sequence alignment: Definition and Goal, The consensus, computational complexity, Manual methods, Simultaneous methods, Progressive methods, Databases of Multiple alignments and searching.

Introduction to Secondary database searching

Text and Reference Books:


1. Introduction to Bioinformatics, T K Attwood & D J Parry-Smith ,Addison Wesley Longman
2. Bioinformatics- A Beginner‘s Guide Jean-Michel Claveriw, Cerdric Notredame , WILEY dreamlech India Pvt. Ltd
3. Introduction to Bioinformatics, M.Lesk , OXFORD publishers (Indian Edition)
4. Biochemistry - Lubert Stryer, WH Freeman and Co.
5. Bioinformatics and Functional Genomics, Jonathan Pevsner, 2009 (new edition Oct 2015)
6. Discovering Genomics, Proteomics, and Bioinformatics, Campbell et al, 2002
7. Bioinformatics - Genes, Proteins and Computers, C.A. Orengo, D.T. Jones and J.M. Thornton, BIOS Scientific Publishers, 2003
8. Mathematical Biology, J.D. Murray, Springer, 1993.

## Big data analytics

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-430-L/IT-430-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4<br><br>Mode: Lecture (L)<br><br>Examination Duration: 3 Hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Prerequisites**

Basic knowledge of databases, data mining.

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are:**

1. To introduce the students about knowledge of Data Management and data analysis
2. To provide a basic understanding of the issues and problems involved in massive on-line repository system.
3. To provide knowledge about Big Data stacks and practical techniques for satisfying the needs of such systems.
4. To understand, use, and build practical big data analytics systems.

**By the end of the course a student is expected to:**

CO1. Know distributed computing for big data.
CO2. Be familiar with virtualization of memory, network, server and applications.
CO3. Gain understanding of HDFS.
CO4. Get acquainted with big data tools

**Unit I**

Big Data Introduction: The Evolution of Data Management, Defining Big Data, Traditional and advanced analytics. Distributed Computing, need of distributed computing for big data, economics of computing, latency problem. Examining Big Data Types, Structured Data, sources of big structured data, role of relational databases in big data, Unstructured Data, sources of unstructured data, role of a CMS in big data management.

**Unit II**

Big Data Stack: Redundant Physical Infrastructure, Security Infrastructure, Operational Databases. Organizing Data Services and Tools, Analytical Data Warehouses, Big Data Analytics, Big Data Applications. Virtualization and big data: Server virtualization, Application virtualization, Network virtualization, Processor and memory virtualization, Data and storage virtualization, Managing Virtualization with the Hypervisor.

**Unit III**

MapReduce Fundamentals, Putting map and reduce Together, Optimizing MapReduce Tasks. Hadoop, Hadoop Distributed File System (HDFS), Name Nodes, Data nodes, Hadoop MapReduce.

**Unit IV**

Big Data Analytics: Basic analytics, Advanced analytics, Operationalized analytics, Monetizing analytics, Text Analytics and Big Data, Social media analytics, Text Analytics Tools for Big Data, Attensity, Clarabridge, OpenText. Integrating Data Sources: Dealing with Real-time Data Streams and Complex Event Processing, Operationalizing Big Data, Applying Big Data within Your Organization, Security and Governance for Big Data Environments.

**Text and reference books:**


1. Big Data For Dummies, Judith S. Hurwitz, Alan F. Nugent, Fern Halper, Marcia A. Kaufman, John Wiley & Sons, Inc. 2013
2. Hadoop For Dummies, Robert D. Schneider, John Wiley & Sons, Inc. 2012.
3. Understanding Big Data: Analytics for Enterprise Class Hadoop and streaming data, Paul Zikopoulos, ,McGraw Hill 2012.

## Operations Research

### General Course Information:

| | |
|---|---|
| Course Code: CSE-432-L/ IT-432-L<br><br>Course Credits: 3.5<br><br>Type: Compulsory<br><br>Contact Hours: 4 Hours/Week<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 Hours | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:**

Basic knowledge mathematical tools like graph.

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are to**

1. Develop proficiency in business study and decide the fusibility of system
2. To carry out the profitable solution for industries.

**By the end of the course a student is expected to able to:**

CO1. Make the decision about business system
CO2. Find the Maximum profit and Minimum lose for the business
CO3. Integer Programming with PERT and CPM
CO4. Understand dynamic Programming and Queuing models

### Course Contents

**Unit I**

**Introduction:** The Historical development, Nature, Meaning and Management application of Operations research. Modeling, Main characteristic and phases of OR, Impact of OR.
**Linear Programming:** Formulation, Graphical solution, Standard and Matrix forms of Linear Programming Problems, Simplex method, Big-M method, Two- Phase method, Degeneracy.
**Duality:** Introduction, Definition, General Rule for converting any primal into its Dual, Dual Simplex method and its flow chart.

**Unit II**

**Assignment problem:** Assignment problem and its mathematical formulation, solution of assignment problem (Hyngarian method).
**Transportation problem:** Transportation problem and its mathematical formulation, Initial basic feasible solution of transportation problem by North-West corner rule, Lowest Cost Entry method and Vogel's Approximation method, Optimal solution of transportation problem.

**Unit III**

**Integer Programming:** Importance and Applications, Gomorg's all integer programming problem technique, Branch and Bound Method.
**PERT and CPM:** Basic steps in PERT and CPM, Forward and Backward computation, Representation in tabular form, Slack and Critical path, Difference between CPM and PERT, Float.

**Unit IV**

**Dynamic Programming:** Introduction to Dynamic Programming, General inventory Model, Static Economic Order Quantity (EOQ) Models.
**Queuing Models:** Introduction, Applications, Characteristic Waiting and Ideal time costs, Transient and Steady states, Kendall's Notations, Pure Birth & Death model, Generalized Poisson Queuing model, Specialized Poisson Queues.

T**ext and Reference Books:**


1. Gupta P.K., Hira and D.S., Operation Research, Sultan Chand & Sons, New Delhi,2002.
2. Kanti Swarup, Gupta P.K. & Man Mohan, Operation Research, Sultan Chand & sons, New Delhi, 2014.
3. Mittal, K.V., Optimization Methods in Operations Research and System Analysis, New Age International (P) Ltd., New Delhi, 2009.
4. Rao S.S., Optimization Theory and Applications,Wiley Eastern Ltd. New Delhi, 1985.
5. Sharma, S.D., Operations Research, Kedar Nath and Ram Nath, Meerut, 2005.
6. Taha, H.A., Operation Research - An Introduction, McMillan Publishing Co, New York, 2010.
7. Bazara, Operation Research & Networking, Wiley, 2008.

## Multimedia Technology

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-434-L/IT-434-L<br><br>Course Credits: 3.5<br><br>Type:<br><br>Contact Hours: 4<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:** Basic programing skills**,** hardware knowledge**.**

**About the Course and its Objectives & Outcomes:** Multimedia is combined use of text, graphics, sound, animation and video. This course introduces students to current multimedia technology. Using skills with graphics, image animation student create dynamic multimedia projects

**The objectives of this course are to:**

1. To demonstrates how still images, sound and video can be digitized on computers.
2. To teach students how to develop multimedia programs.

**By the end of the course a student is expected to:**

CO1. Formulate a working definition of interactive multimedia.

CO2. Develop conceptual maps for content and process for interactive multimedia programs.

CO3. Plan and design multimedia projects using virtual reality environment

CO4. Work on action scripts in Macromedia flash

### Course Contents

**Unit I**

**Basic of Multimedia Technology:** Computer, Communication and Entertainment; Multimedia an introduction, Multimedia devices;CD-Audio,CD-ROM,CD-I,DVD, Presentation devices and the user interface, Multimedia Authoring Tools, Text in multimedia, Symbols and Icons, Choosing text fonts,3D graphics program, Animation Techniques,Antialiasing,Morphing,Video on demand, Application of multimedia.

**Unit II**

**Image Compression and Standards:** Making still images; editing and capturing images, scanning images, Computer color models, Color palettes; Vector drawing;3D Drawing and Rendering, JPEG objectives and Archietecture;JPEG-DCT encoding and quantization, JPEG performance, Overview of other image file formats GIF,TIFF,BMP,PNG etc

**Unit III**

**Audio and Video:** Digital audio, making digital audio files. MIDI audio, MIDI vs Digital audio, Audio file format, Adding sound to multimedia projects,How video works and displayed: Analog video, Digital video, digital video containers,MPEG,Shooting and Editing video,DVI technology, Virtual reality and Virtual Environment Displays, Planning and Designing the multimedia project.

**Unit IV**

**Introduction to flash:** Flash Environment:-,interface;menubar,frames,timeline,Drawing and Working with color,symbols,buttons and movie clips, Layers and its working, introduction to action script in flash, action panel, operators object and properties

T**ext Reference Books:**


1. An introduction, Villamil &Molina, Multimedia Mc Milan,1997.
2. Multimedia: sound & video,Lozano,PHI,(Que), 1997.
3. Multimedia: Production, Planning and Delivery, Vilamil & Molina, Que, 1997.
4. Multimedia on the PC,Sinclair, BPB, 1995.
5. Multimedia: Making it work, Tay Vaughan, fifth edition,TMH, 1994.
6. Multimedia System By Koegel, AWL, 2010.
7. Multimedia Communication by Halsall & Fred,2001.

## Natural Language Processing

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-436-L/ IT-436-L<br><br>Course Credits: 3.5<br><br>Type:<br><br>Contact Hours: 4<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:** A course on compiler design and artificial intelligence

**About the Course and its Objectives & Outcomes:**

**The objectives of this course are to:**

1. To learn the techniques in natural language processing
2. To understand the syntax and semantics of natural languages.
3. To understand semantics and machine translation for any compiler.

**By the end of the course a student is expected to:**

CO1. Analyse the natural language text
CO2. Generate the natural language
CO3. Do machine translation
CO4. Do semantic Analysis such as syntax driven analysis and semantic grammars etc

**Course contents**

**Unit I**

Introduction: Need for Processing Natural languages, Issues in NLP and Complexity of Processing NLP, Brief history of NLP application development. Language Modeling: Various types of Languages and its modeling, Grammar based language models, Government and Binding, Lexical Functional Grammar and Paninian Grammar for handling natural languages, Statistical modeling

**Unit II**

Word Level Analysis: Regular expressions, Finite State Automata, Morphological parsing, Spelling Error Detection and Correction, Words and word classes (Hindi and English), Part of speech tagging : Rule-based tagger, Stochastic tagger, Hybrid tagger, Unknown words
Syntactic Analysis: Context Free Grammar, Phrase and sentence level Constructions, Parsing: Top-down Parsing, Bottom-up parsing, A Basic Top-down Parser, The Earley Parser, The CYK Parser, Probabilistic Parsing : Estimating Rule Probabilities, Parsing PCFGs, Problems with PCFG

**Unit III**

Semantic Analysis: Meaning Representation, Characteristics of Meaning Representation Languages, Meaning structure of languages, Syntax-driven semantic analysis, Semantic Grammars, Lexical Semantics, Relationships, Internal structure of words, Ambiguity, Word Sense Disambiguation, Selectional Restriction in Word sense Disambiguation, Context-based Word Sense Disambiguation Approaches, Knowledge sources in WSD, Applications of WSD, WSD Evaluation Discourse Context and World Knowledge: Local discourse Context and Anaphora Resolution, World Knowledge, Discourse Structure, Discourse Analysis
Language Generation: Architecture of language generators, Template-based, Phrase-based and Feature-base d Natural language generation, Knowledge-based Approaches

**Unit IV**

Machine Translation: Problems in Machine Translation, Characteristics of Indian Languages, Machine Translation Approaches: Direct Machine Translation, Rule-based
Machine Translation: Transfer-based and Interlingua based Machine Translation, Corpus-based Machine Translation: Statistical and Example-based Machine Translation, Semantic or Knowledge based MT systems

**Text and reference books:**


1. Natural language Processing and Information Retrieval, T. Siddiqui and U. S. Tiwary, Oxford Univ. Press, 2008.
2. Natural Language Understanding, James Allen, 2/e, Pearson Education, 2003
3. Statistical Language Learning: E. Charniac, MIT Press, 2000.
4. Natural Language Processing with Python, Steven Bird, 1st Edition, 2009.
5. Foundations of Statistical Natural Language Processing, Christopher Manning, 1999.
6. Speech and Language Processing, D. Jurafsky, J. H. Martin, Pearson Education, 2002
7. Natural Language Processing and Language Representation, L.M. Ivansca, S. C. Shapiro.
8. Language as a Cognitive Process, T. Winograd, Addison-Wesley, 1997.
9. An introduction to Linguistics, language grammar and semantics, P.Syal , D.V.Jindal, Eastern Economy Edition, 2009.

# Management Information System

**General Course Information:**

| | |
|---|---|
| Course Code: CSE-438-L/ IT-438-L<br><br>Course Credits: 3.5<br><br>Type:<br><br>Contact Hours: 4<br><br>Mode: Lecture(L)<br><br>Examination Duration: 3 | **Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks.<br><br>For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks. |

**Pre-requisites:** Knowledge of computer terminologies, databases and programming language(s) will help in understanding the given concepts very easily.

**Course Objectives:**

1. To study of people, technology, organizations, and the relationships among them.
2. To understand the basic needs of an organization required to design and develop an efficient and effective MIS.
3. To understand various control and security issues to be taken care at the time of development of an MIS
4. To learn various tools and methods required to develop an effective and efficient MIS.

**By the end of the course a student is expected to:**

CO1. Study of people, technology, organizations, developments & trends and the relationships among them.
CO2. Understand the basic needs of an organization to design and develop an efficient and effective MIS tool for decision-making.
CO3. Develop computer-based secure MIS that provides decision-makers with the tools to organize.
CO4. Evaluate and efficiently manage various functions within an organization.

**Course Contents**

**Unit-I**

Background Meaning, Nature, Need, Role, Importance, Evolution of management through information system; Relatedness of MIS with management process.Management functions and decision-making.Concept of balance MIS effectiveness and efficiency criteria.Development and trends in telecommunications and internet technologies.

**Unit-II**

Development of Management Information System: Introduction, Information system planning, Motivational forces behind development of information system, Principles for information system development, SDLC for MIS development process.

**Unit-III**

Development of MIS: Methodology and Tools techniques for systematic identification, implementation, evaluation, and maintenance of MIS.
Control and Security Issues in Management Information Systems: Control, Why need to Control MIS, Types of Control, Audit in MIS, Security Hazards, Security Techniques. Challenges of usage of IT.

**Unit-IV**

Introduction to ERP, CRM, SCM, Data Warehouse and Data Mining concepts and their relevancy with computerized MIS.
Case studies: To introduce business problems and to discuss various stages for understanding the systems development process.

T**ext and Reference Books:**


1. James A.O'Brien, Management Information Systems.
2. Kenneth C. Laudon, Jane P. Laudon, AhmedElragal, Management Information Systems: MANAGING THE DIGITAL FIRM, Pearson.
3. S. Sadagopan, Management Information Systems, PHI Learning; Second edition (2014)
4. Management Information Systems: A Computer oriented approach for business applications by Dharminder Kumar, Sangeeta Gupta, Excel books, 2006, New Delhi.
5. InderjitChatterjee, Management Information systems, PHI Learning Pvt. Ltd.(2010).
6. Davendranath G. Jha, Computer concepts and Management Information Systems, PHI Learning Pvt. Ltd.(2013).