APPENDIX
Syllabus (B.3.6)

SEMESTER – I

Course code: CS1302
Course title: **Fundamentals of Unix & C Programming**
Pre-requisite(s): A familiarity with high school algebra is expected, however the course assumes no prior programming knowledge

Course Outcomes
After the completion of this course, students will be able to:

1. of logging-in into UNIX operating system, create their own directories and files, move around among different files and directories, create, files in different ways.
2. of creating , editing, debugging and executing C programs using the Vi-editor.
3. To solve problems related to arrays, strings, record and file handling.

4 Classify a problem into functions and synthesize a complete program using Divide and conquer approach

5. To able to solve real life problem.

Syllabus

Module – I [6 lectures]
The Free Software Movement, Open source definition, Open source business strategy, Problem Solving and its tools, Flow chart, Pseudo code, Modular programming. Fundamentals of Unix Operating System, Login & Password, Different Commands, Unix directory, Structure and working with directories, Vi-editor, Basic Structure and execution of C programmes, Constants, Variables, and Data Types and various type of declarations, Different type operators and Expressions, Evaluation of Expressions, Operator Precedence and Associability, Mathematical Functions.

Module –II [4 lectures]
Managing Input and Output operations, Decision Making and Branching Decision Making and Looping.

Moule – III [5 lectures]
One – dimensional Arrays and their declaration and Initialisations, Two-dimensional Arrays and their initialisations, Multidimensional Arrays, String Variables, Reading and Writing Strings, Arithmetic Operations on characters, Putting Strings together, Comparison of Two Strings, String – handling functions.

Module –IV [5 lectures]
Need and Elements for user –defined Functions, Definition of Functions, Return values and their types, Function calls and Declaration, Arguments and corresponding return values, Functions that return multiple values, Nesting of functions, Recursion, Passing arrays and strings to functions, The Scope, Visibility and Life time of variables.

Module –V [5 lectures]
Defining Structure, Declaring Structure Variable and Accessing Structure Members, Initialisation of Structure, Comparing Structure Variables, Operation on Individual Members, Arrays of Structures, Structures

within structures, Structures and Functions, Unions, Size of Structures, Bit Fields.

Module – VI [6 lectures]

Understanding Pointers, Accessing the Address of a Variable, Declaration and Initialisation of Pointer Variables, Accessing a Variable through its Pointer, Chain of Pointers, Pointer Expressions, Pointer Increments and Scale Factor, Pointers and Arrays, Pointers and Character Strings, Arrays of Pointers, Pointers and Function Arguments, Functions Returning Pointers, Pointers to Functions, Pointers and Structures,

Module – VII [4 lectures]

File Management in C. use of fopen(), fclose(), Formatted file I/O, Searching through files using fseek(), ftell(), rewind().

Text Book :

1. Kernighan K. R., Ritchie D. M. - The C Programming Language, Ansi C Edition, Prentice Hall, India

Reference Books:

1. E. Balagurusamy – Programming in ANSI C, 3rd Edn. , TMH, New Delhi ; 2004
2. A. N. Kanthane – Programming with ANSI and TURBO C, Pearson Education, New Delhi; 2004
3. Y. Kanetkar – Let us C, 4th Edition, BPB Publication , New Delhi; 2002
4. Chris DiBona, Sam Ockman , Open Sources : Voices from the Open Source Revolution (Web book), Oreilly Press, 2nd edition,1999.

Course code: CS2301
Course title: **Fundamentals of Data Structures**
Pre-requisite(s): Knowledge of programming languages like C/C++.

Course Outcomes
After the completion of this course, students will be able to:

1. Understand the properties of various data structures
2. Identify the strength and weaknesses of different data structures
3. Design and employ appropriate data structures for solving computing problems
4. Analyze and compare the efficiency of algorithms
5. Think critically

Module – I [5 lectures]
Algorithms and Analysis of Algorithms: Definition, Structure and Properties of Algorithms, Development of an Algorithm, Data Structures and Algorithms, Data Structure – Definition and Classification, Efficiency of Algorithms, Apriory Analysis, Asymptotic Notations, Time Complexity of an Algorithm using O Notation, Polynomial Vs Exponential Algorithms, Average, Best and Worst case Complexities, Analyzing Recursive Programs, Open source software development process.
Module – II [5 lectures]
Arrays, Stacks and Queues: Array Operations, Number of Elements in an Array, Representation of Arrays in Memory, Applications of Array, Stack-Introduction, Stack Operations, Applications of Stack, Queues-Introduction, Operations on Queues, Circular Queues, Other Types of Queues, Applications of Queues.
Module – III [5 lectures]
Linked List, Linked Stacks and Linked Queues: Singly Linked Lists, Circularly Linked Lists, Doubly Linked Lists, Multiply Linked Lists, Applications of Linked Lists, Introduction to Linked Stack and Linked Queues, Operations on Linked Stacks and Linked Queues, Dynamic Memory Management and Linked Stack, Implementations of Linked Representations, Applications of Linked Stacks and Linked Queues.
Module – IV [6 lectures]
Trees, Binary Trees, BST, AVL Trees and B Trees: Trees: Definition and Basic Terminologies, Representation of Trees, Binary Trees: Basic Terminologies and Types, Representation of Binary Trees, Binary Tree Traversals, Threaded Binary Trees, Applications, BST & AVL Trees: Introduction, BST: Definition and Operations, AVL Trees: Definition and Operations, B Trees: Introduction, m-way search trees: Definition and Operations, B Trees: Definition and Operations.
Module – V [5 lectures]
Graphs: Introduction, Definitions and Basic Terminologies, Representations of Graphs, Graph Traversals, Single-Source Shortest-Path Problem, Minimum Cost Spanning Trees.
Module – VI [5 lectures]
Sorting: Introduction, Shell Sort, Quick Sort, Heap Sort.
Module – VII [4 lectures]
Searching: Introduction, Binary Search, Transpose Sequential Search, Interpolation Search.

Text Books:
1. G A V Pai – Data Structures and Algorithms: Concepts, Techniques and Applications, 2nd Edn, Tata McGraw-Hill, 2008
2. Horowitz E.Sahni, S., Susan A., Fundamentals of Data Structures in C, 2nd Edition, University Press, 2010

Reference Books:
1. J. P. Tremblay **,** P. G. Sorenson – An Introduction to Data Structures With Applications, 2nd

Edn, McGraw-Hill, Inc. New York, NY, USA.

2. Seymour Lipschutz – Data Structures, 6th Edn, 9th Reprint 2008, Tata McGraw-Hill.
3. Adam Drozdek – Data Structures and Algorithms in C++, Thomson Learning, New Delhi – 2007.
4. J. Feller, B. Fitzgerald -Understanding Open Source Software Development, Pearson Education Ltd. New Delhi

Course code: CS2302
Course title:**Fundamentals of Data Structures Lab**
Pre-requisite(s): Programming Language like C/C++.

Course Outcomes

After the completion of this course, students will be able to:

1. Arrays, Stacks and Queues
2. Various kinds of Linked Lists
3. Various kinds of Trees and perform appropriate operations.
4. Graphs
5. Sorting and searching techniques.

Text Books:

1. G A V Pai – Data Structures and Algorithms: Concepts, Techniques and Applications, 2nd Edn, Tata McGraw-Hill, 2008
2. Horowitz E.Sahni, S., Susan A., Fundamentals of Data Structures in C, 2nd Edition, University Press, 2010

Reference Books:

1. J. P. Tremblay **,** P. G. Sorenson – An Introduction to Data Structures With Applications, 2nd Edn, McGraw-Hill, Inc. New York, NY, USA.
2. Seymour Lipschutz – Data Structures, 6th Edn, 9th Reprint 2008, Tata McGraw-Hill.
3. Adam Drozdek – Data Structures and Algorithms in C++, Thomson Learning, New Delhi – 2007.
4. J. Feller, B. Fitzgerald -Understanding Open Source Software Development, Pearson Education Ltd. New Delhi.

Course code: CS3005
Course title: **Object Oriented Programming Using Java**
Pre-requisite(s): Data Structure and Programming skills

Course Outcomes

After the completion of this course, students will be able to:

1. Understand basic concepts of object oriented programming.
2. Improve their programming skills.
3. Apply the knowledge gained for their project work as well as to develop some GUI applications.
4. Students are expected to go through the websites for latest know-how related to the subject.

Module - 1

Introduction to OOP, Objects and classes, Characteristics of OOP, Difference between OOP and Procedure oriented programming. Introduction to Java Programming, Features of Java, Applications and Applets, JDK, Source File Structure

Module-II

Java language fundamentals, Building blocks of Java, Data Types, Variable declaration, Wrapper classes, Operators and Assignment, Control Structures, Arrays, Strings, StringBuffer class

Module -III

Java as an OOP Language, Defining classes, Modifiers, Packages, Interfaces, Exception Handling, Exception hierarchy, Constructors and methods of Throwable class, Unchecked and Checked Exceptions, Handling Exceptions in Java, Exception and Inheritance, Throwing user defined exceptions, Redirecting and rethrowing exceptions.

Module -IV

Multithreading, Overview of threading, Creating threads, Thread Life-cycle, Thread priorites and Thread scheduling, Thread synchronization, Daemon Threads, Thread groups, Communication of Threads

Module -V

Files and I/O Streams, Java I/O, File Streams, FileInputStream and FileOutputStreams, Filter streams, Random Access files, Serialization

Module - VI

Applets, Java Applications versus Java Application, Applet Life cycle, Working with Applets, The HTML APPLET Tag, java.Applet package

Module -VII

AWT, Basic classes in AWT, Drawing with Graphics Class, Class hierarchy of AWT, Event Handling, Adapter classes, AWT Controls, Layout Managers, Swings, Swings packages, Hierarchy of Swing classes, Advanced layout Managers, Additional Swing Components.

Text Book:

1. Krishna P. R., Object Oriented Programming through JAVA, 1st Edition, Universities Press, 2008.

Reference Books:

1. Dietel,Dietel - Java How to program , 7th edition; Pearson Education , New Delhi.
2. C. Horstmann,G. Cornell - Core Java 2 Vol I & Vol II ; Pearson Education , New Delhi.
3. Balagurusamy -Programming in Java, 2nd Edition; Tata McGraw Hill Publication; New Delhi.
4. Patrick Naghton & H. Schildt – The Complete Reference Java 2, Tata McGraw Hill Publication, New Delhi.

Course code: CS6101
Course title: **Design and analysis of Computer Algorithms**
Pre-requisite(s): Knowledge of basic data structures

Course Outcomes

After the completion of this course, students will be able to:

1. Understand basic algorithm designing techniques such as divide and conquer, greedy, dynamic programming, backtracking and branch and bound and analyze them. .
2. Various sorting algorithms, Travelling salesperson problem, shortest path problem, sum of subset, 0/1 knapsack problem.
3. Understand and apply fundamental algorithmic problems including job scheduling, N queen problem, 8 puzzle problem, flow shop scheduling, optimal binary search tree, matrix multiplication, minimum weight spanning tree.
4. Exploration of various research problems in algorithm like NP-hard and NP-complete problems
5. Student will able to solve problems and analyze them using randomized approach also.

Module -I

Basic Tools on Designing Algorithms: What is an algorithm? Algorithm specification and performance analysis, randomized algorithms.

Module –II

Divide-and-Conquer: The general method, application to binary search, finding the maximum and minimum, merge sort, quick sort, the problem of selection and Strassen's matrix multiplication.

Module -III

The Greedy Method: The general method, application to optimal storage on tapes, job sequencing with deadlines, optimal merge patterns and minimum weight spanning trees.

Module –IV & V

Dynamic Programming: The general method, application to multistage graphs, all pairs shortest paths, optimal binary search trees,0/1-Knapsack and traveling salesman problem, Flow shop scheduling

Backtracking: The general method, application to 8- puzzle problem, 8- queen problem and sum of subsets.

Module -VI

Branch and Bound: The method, application to 0/1 Knapsack traveling salesman problems, and efficiency considerations.

Module -VII

NP-Hard and NP-Complete Problems: Introduction and basic concepts, non-deterministic turing machine, the classes of P and NP, NP-hard graph problems, NP-completeness of the satisfiability problem, and polynomial- space-bounded problem.

Text Book:

1. E. Horowitz. et.al., Fundamentals of computer Algorithms, Universities Press, 2008, 2nd Edition.

Reference Books:

1. J.Kleinberg & E. Tardos – Algorithm Design, Pearson Education, New Delhi, 2006.
2. G.Brassard & P. Bratley – Fundamentals of Algorithms, PHI, New Delhi, 2005.
3. T.H. Cormen et.al. – Introduction to Algorithms – PHI, New Delhi, 2005.
4. S.Dasgupta et.al. – Algorithms, TMH, New Delhi – 2007.

Discrete

Course code: CS3006
Course title: **Object Oriented Programming Using Java Lab**
Pre-requisite(s): Data Structure and Programming skills

Course Outcomes

After the completion of this course, students will be able to:

1. Students are expected to perform well in viva-voce/ sessional tests/ class assignments examination.
2. Students are expected to create simple programs using basic programming and OOPs concepts.
3. Students are expected to improve their programming skills.
4. Students are expected to apply the knowledge gained for their project work.
5. Students are expected to go through the websites for latest know-how related to the subject.

Text Book:

1. Krishna P. R., Object Oriented Programming through JAVA, 1st Edition, Universities Press, 2008.

Reference Books:

1. Dietel,Dietel - Java How to program , 7th edition; Pearson Education , New Delhi.
2. C. Horstmann,G. Cornell - Core Java 2 Vol I & Vol II ; Pearson Education , New Delhi.
3. Balagurusamy -Programming in Java, 2nd Edition; Tata McGraw Hill Publication; New Delhi.
4. Patrick Naghton & H. Schildt – The Complete Reference Java 2, Tata McGraw Hill Publication, New Delhi.

Course code: CS4205
Course title: **Database Management System**
Pre-requisite(s): Fundamental of Data Structures

Course Objectives

This course enables the students:

1. Understand basic concepts of modelling and structuring of databases
2. Have knowledge of ‘indexing and hashing’ mechanisms in a database management system.
3. Have idea of the backend activities involved in extracting data from a database.
4. Have knowledge of transaction and concurrency control mechanisms in a database management system.
5. Know how to compute cost of query and normalize tables for efficient operations

Course Outcomes

After the completion of this course, students will be able to:

1. Understand the fundamental and advanced concepts required for modelling and designing the database.
2. Formulate, using relational algebra and SQL, solutions to a broad range of query problems.
3. Apply the concept of normalization techniques to effectively model and design the database systems.
4. Analyze Indexing and Hashing mechanisms in a database management system.
5. Elaborate and evaluate the concepts of Transaction and Concurrency control.

Module – I

Introduction: Purpose of Database System; View of Data, Database Languages, Transaction Management, Database architecture, Database Users Administrator

Database Design and Entity - Relational Model: Overview of Design process, E-R model, Constraints, E-R diagrams, Week Entity Sets, Extended E – R Features.

Module – II

Relational Model: Structure of Relational Database, Fundamental Relational Algebra, Operation, Additional Operations, Tuple Relational, Calculus.

Module – III

SQL and Advanced SQL: Data definition, Basic structure of SQL queries, Set Operations, Aggregate Functions, NULL values, Nested Sub-queries, complex queries, views, modification of database, SQL data types & schemas, Integrity constraints, Authorization, Embedded SQL.

Module – IV

Relational Database Design: Atomic domains and First Normal form, Decompositions using functional dependencies, Functional dependencies, Decomposition using multivalued dependencies, more normal forms.

Module – V

Indexing and Hashing: Basic concepts, Ordered Indices, B+ Tree Index Files, B Tree Index files, Multiple Key Access, Hashing, Comparison of Ordered Indexing and Hashing

Module VI

Query processing- Overview, measures of query cost, selection operation, sorting join operations

Module – VII

Transaction & Concurrency Control: Transaction Concepts & ACID Properties, Transaction States, Concurrent Executions, Serializability & Its Testing, Recoverability, Introduction to Concurrency Control, Locked Base Protocol & Deadlock Handling. Timestamp-Based Protocols. Validation-Based Protocols. Multiple Granularity.

Text Book:

1. A.Silberschatz et.al - Database System Concepts, 5th Ed^{n}, Tata Mc-Graw Hill, New Delhi – 2000.

Reference Books:

1. Date C.J. - An Introduction to Database System, Pearson Education, New Delhi, 2005.
2. R.Elmasri, Fundamentals of Database Systems, Pearson Education, New Delhi, 2005.

Course code: CS4109
Course title: **Computer System Architecture**
Pre-requisite(s): Digital Logic Design

Course Outcomes

After the completion of this course, students will be able to:

1. Describe the merits and pitfalls in computer performance measurements and analyse the impact of instruction set architecture on cost-performance of computer design.
2. Explain Digital Logic Circuits ,Data Representation, Register and Processor level Design.
3. Solve problems related to computer arithmetic and Determine which hardware blocks and control lines are used for specific instructions.
4. Design a pipeline for consistent execution of instructions with minimum hazards.
5. Explain memory organization ,I/O organization and its impact on computer cost/performance.

Module- I

Design Methodology: System Design, System Representation, Design Process, Gate, Registers, Register-Level Components, Programmable Logic Devices, Register-Level Devices, Processor-level Components, Processor-level Design.

Module- II

Processor Basics: CPU Organization, Fundamentals, Additional Floating-Point Numbers, Data Representation Basic Format, Fixed-Point Numbers, Floating-Point Numbers, Instruction Sets, Instruction Formats and Types.

Module- III

Datapath Design: Fixed-Point Arithmetic, Addition, Subtraction, Multiplication and Division, Arithmetic Logic Units, Combinational ALUs, Sequential ALUs.

Module - IV

CONTROL DESIGN: Basic Concepts, Introduction, Hardwired Control, Microprogramming, Basic Concepts, Multiplier Control Unit, Control, Pipeline Control, Instruction Pipeline, Arithmetic Pipeline.

Module - V

Memory Organization

Memory Technology, Memory Device Characteristics, Random Access Memories, Serial Access Memories, Memory Systems Multilevel Memories, Address Translation, Memory Allocation, Cache, Main Features, Address Mapping.

Module - VI

System Organization: Communication Methods, Basic concepts Bus Control, System Control, DMA and Interrupts.

Module- VII

Advanced Topics: Pipeline Processing, Parallel Processing.

Text Book

1. Hayes, J.P., "Computer Architecture and Organization", 3rd ed McGraw-Hill, London , 2000

R**eference Books**

1. Mano, M.M., "Computer System Architecture" , Prentice Hall of India, New Delhi, 1995
2. Heuring V.P., etal., " Computer System Design and Architecture", Addision Wesley Indian Reprint, 2000
3. Hamacher.V., etal, (Computer Organzation" ,4th edition, McGraw Hill, Singapore, 1996
4. Ram. B."Computer Fundamentals: Architecture and Organization",3rd ed New Age International Publication, New Delhi, 2000.
5. David A. Patterson, John L. Hennessy, "Computer Organization and Design: The Hardware / Software Interface", 5th Edition, Elsevier

Course code: CS4107
Course title: **Operating System**
Pre-requisite(s): Fundamental of Data Structures, Computer System Architecture, Design and Analysis of Computer Algorithms.

Course Outcomes

After the completion of this course, students will be able to:

1. Understand the role and working mechanisms of the main components of OS
2. Explain the concepts of process, their scheduling policies and handle synchronization and deadlock issues
3. Understand directory structures and file handling
4. Understand different techniques for managing main-memory and secondary storage
5. Compare and analyse performance of various design options of OS components

Syllabus

Module – I

Introduction: What is an Operating System? Simple Monitor, Performance, Multiprogramming, Time-sharing, Real Time systems, Protection.

File Systems: File Concept and support, Access and allocation methods, directory systems, File protection.

Module – II

CPU Scheduling: Scheduling concepts and algorithms, Algorithms evaluation, and Multiple processor scheduling.

Module – III

Memory Management: Preliminaries, Bare Machine, Resident Monitor, Swapping, Multiple Partitions, Paging, Segmentation, Combined systems.

Module – IV

Virtual Memory : Overlays, Demand paging, Performance of demand paging, Page replacement, Virtual memory concepts, Page replacement algorithms, Allocation algorithms, and Thrashing.

Module – V

Disk Scheduling: Physical characterization, Disk Management, Swap-Space Management, RAID structure, FCFS scheduling and Shortest-Seek-Time-First.

Module – VI

Deadlocks: The deadlock problem, Deadlock characterization, Deadlock prevention, Deadlock avoidance; Deadlock detection, Recovery from deadlock and combined approach to deadlock handling.

Module – VII

Process Synchronization: Semaphores, OS Synchronization, Atomic Transaction.

Security: The Security Problem, User Authentication, Cryptography.

Text Book:

1. Silver Schatz, A and Golvin, P.B. ‘Operating System Concepts’, 5th Edn. John Wiley, New Yark, 2000.

Reference Books:

1. Deitel H.M., ‘An Introduction to Operating System’, Addison Wesley, Inc., London, 1995.
2. Mandinck S.E., ‘Operating System’ McGraw Hill., London, 1993.

Course code: CS4108
Course title: **Operating System Lab**
Pre-requisite(s): Fundamental of Data Structures, Computer System Architecture, Design and Analysis of Computer Algorithms.

Course Outcomes
After the completion of this course, students will be able to:

1. Understand working mechanism of operating system components and their design choices.
2. Implement and evaluate performance of CPU scheduling algorithms.
3. Implement and evaluate performance of contiguous memory allocation strategies.
4. Implement and evaluate performance of page replacement policies used in demand paging.
5. Implement and evaluate performance of disk scheduling algorithms.

Reference Books:

1. Operating Systems: A Modern Perspective, Lab Update, 2nd Edition, 2001, Addison Wesley.
2. Operating systems - Internals and Design Principles, W. Stallings, 8th Edition, 2014, Pearson
3. Operating System Principles, Abraham Silberchatz, Peter B. Galvin, Greg Gagne, 9th Edition, 2013, Wiley Student Edition.
4. Modern Operating Systems, Andrew S Tanenbaum 4th Edition, 2014, Pearson.

Course code: **CS6107**
Course title: **Computer Networks**
Pre-requisite(s): **Computer System Architecture**

Course Outcomes

After the completion of this course, students will be able to:

1. Students are expected to know the characteristics of different types of computer networks.
2. To gain knowledge of the computer network design choices and their application areas.
3. To understand the data communication techniques used in local area networks and internetworks.
4. To understand the operations of different network protocols.
5. Knowledge of contemporary issues in computer networks.

Module -I

Foundation: Applications, Requirements, Network Architecture, Implementing Network Software, Performance.

Module –II

Direct Link Networks: Hardware Building Blocks, Encoding (NRZ, NRZI, Manchester, 4B/5B), Framing, Error Detection, Reliable Detection, Reliable Transmission,

Module - III

Ethernet (802.3), Token Rings (802.5, FDDI), Wireless (802.11), Network adaptors.

Module –IV

Internetworking: Simple Internetworking (IP), Routing, Global Internet,

Module - V

Multicast, Multiprotocol Label Switching (MPLS).

Module -VI

End-to-End Protocols: Simple Demultiplexer (UDP), Reliable Byte Stream (TCP), Remote Procedure Call.

Module -VII

Congestion Control and Resource Allocation: Issues in Resource Allocation, Queuing Disciplines, TCP Congestion Control, Congestion-Avoidance Mechanisms, Quality of Service.

Text Book:

1. L.L. Peterson & B.S. Davie- Computer Networks: A Systems Approach, 3rd Edition, Morgan Kaufman Publication, New Delhi, 2006

Reference Books:

1. A. Forouzan **-** Data communications and Networking, 4th Edn., TMH, New Delhi, 2006
2. P.C. Gupta- Data Communications and Computer Networks, PHI, New Delhi, 2006**.**

Course code: CS5101
Course title: **Formal Languages and Automata Theory**
Pre-requisite(s): Discrete Mathematics

Course Outcomes

After the completion of this course, students will be able to:

1. Demonstrate an ability to apply the knowledge for solving real world problems.
2. Relate practical problems to languages and automata.
3. Design an appropriate abstract machine to solve a problem.
4. Design a grammar for a given formal language.
5. Develop a clear understanding of undecidability and computational efficiency.

Module - I

Introduction to Automata: Study and Central concepts of automata theory, An informal picture of finite automata, deterministic and non-deterministic finite automatas, applications of finite automata, finite automata with epsilon – transitions.

Module - II

Regular expression and languages: Regular expressions, finite automata and regular expressions, applications of regular expressions, algebraic laws of regular expressions.

Module - III

Properties of Regular Languages: Proving languages not to be regular, closure properties of regular languages, equivalence and minimization of automata.

Module - IV

Context – free Grammars and Languages: Parse trees, Applications of context free grammars, Ambiguity in grammars and languages.

Module - V

Pushdown Automata: Pushdown automation (PDA), the language of PDA, equivalence of PDA's and CFG's, Deterministic Pushdown Automata.

Module - VI

Properties of Context – Free Languages: Normal forms of context free grammars, pumping lemma for context free languages, close properties of context free languages.

Module - VII

Introduction to Turing Machine: The Turing machine, programming techniques for Turing machine, extensions to the basic Turing machine, restricted Turing Machines, Turing Machines and Computers.

Text Books:

1. J.E. Hopcroft , et.al. - Introduction to Automata Theory, Languages and Computation, 2nd Edn. Pearson Education , New Delhi 2001.

Reference Books:

1. K.L.P. Misra – et.al. - Theory of Computer Science, 2nd Edn. PHI, New Delhi, 2000.
2. J.C. Martin - Introduction to Languages and the Theory of Computation 2nd Edn, TMH, New Delhi, 2000.

Course code: CS8101
Course title: **Artificial Intelligence & Expert Systems**
Pre-requisite(s): Data Structure and Data Base Management systems

Course Outcomes

After the completion of this course, students will be able to:

1. To provide students with in-depth theoretical base for the generation of artificial intelligence and development of the expert systems.
2. To provide information regarding the design and Implementation methods for Knowledge representation, organization, manipulation and acquisition.
3. The principles on dealing with inconsistencies and uncertainties.
4. Proper understanding of heuristic search techniques.
5. To familiarize the students regarding the various types of expert systems available and implementation details for implementing an expert system.

Module - I

Overview of Artificial Intelligence: Definition & Importance of AI.

Knowledge: General Concepts: Introduction, Definition and Importance of Knowledge, Knowledge-Based Systems, And Representation of Knowledge, Knowledge Organization, Knowledge Manipulation, And Acquisition of Knowledge.

Module - II

LISP and Other AI Programming Languages: Introduction to LISP : Syntax and Numeric Function, Basic List Manipulation Functions in LISP, Functions, Predicates and Conditionals, Input, Output and Local Variables, Iteration and Recursion, Property Lists and Arrays, Miscellaneous Topics, PROLOG and Other AI Programming Languages.

Module - III

Knowledge Representation: Introduction, Syntax and Semantics for Propositional logic, Syntax and Semantics for FOPL, Properties of Wffs, Conversion to Clausal Form, Inference Rules, The Resolution Principle, No deductive Inference Methods, Representations Using Rules.

Module - IV

Dealing with Inconsistencies and Uncertainties: Introduction, Truth Maintenance Systems, Default Reasoning and the Closed World Assumption, Predicate Completion and Circumscription, Modal and Temporal Logics.

Probabilistic Reasoning: Introduction, Bayesian Probabilistic Inference, Possible World Representations, Dumpster-Shafer Theory, Ad-Hoc Methods.

Module - V

Structured Knowledge: Graphs, Frames and Related Structures: Introduction, Associative Networks, Frame Structures, Conceptual Dependencies and Scripts.

Object-Oriented Representations: Introduction, Overview of Objects, Classes, Messages and Methods, Simulation Example using an OOS Program.

Module - VI

Search and Control Strategies: Introduction, Preliminary Concepts, Examples of Search Problems, Uninformed or Blind Search, Informed Search, Searching And-Or Graphs.

Matching Techniques: Introduction, Structures Used in Matching, Measures for Matching, Matching Like Patterns, Partial Matching.

Module - VII

Knowledge Organization and Management: Introduction, Indexing and Retrieval Techniques, Integrating Knowledge in Memory, Memory Organization Systems.

Expert Systems Architectures: Introduction, Rule Based System Architecture, Non-Production System Architecture, Dealing with uncertainty, Knowledge Acquisition and Validation, Knowledge System Building Tools.

Text Book:

1. Dan W. Patterson - Introduction to Artificial Intelligence and Expert Systems, PHI, New Delhi, 2006.

Reference Books:

1. E. Rich & K. Knight - Artificial Intelligence, 2/e, TMH, New Delhi, 2005.
2. P.H. Winston - Artificial Intelligence, 3/e, Pearson Edition, New Delhi, 2006.
3. D.W. Rolston,- Principles of AI & Expert System Development, TMH, New Delhi.

Course code: CS6108
Course title: Computer Networks Lab
Pre-requisite(s): Computer System Architecture

Course Outcomes

After the completion of this course, students will be able to:

1. Gain knowledge of usage of networking commands.
2. Apply encoding/decoding, Encryption/decryption techniques.
3. Develop programs to implement basic communication techniques.
4. Develop simple client server programs using sockets.
5. Solve network based real life problems.

Reference Books:

1. J. Graba, An introduction to network programming with Java, 3rd edition, Springer Science and Business Media, 2013.
2. E. R. Harold, Java network programming: Developing networked applications, O’Reilly Media Inc, 2013.

Course code: CS8102
Course title: **Artificial Intelligence & Expert Systems Lab**
Pre-requisite(s): Data Structure and Data Base Management systems

Course Outcomes

After the completion of this course, students will be able to:

1. Students are expected to apply AI techniques to different complex problems using programs.
2. Students will be able to create knowledge base and serve their applications in different fields.
3. Students are expected to implement perfect and better heuristics for different applications.
4. Students are able to create Expert systems.

Text Book:

1. Prolog Programming by Bratko, PHI publishing.

Course code: CS6105
Course title: **Compiler Design**
Pre-requisite(s): Data Structures, Basics of Automata theory (regular expressions, context free grammar)

Course Objectives
This course enables the students:

1. Understand the need of compiler in Computer Engineering.
2. Provide a thorough understanding of design, working, and implementation of programming Languages.
3. Trace the major concept areas of language translation and compiler design.
4. Create an awareness of the functioning and complexity of modern compilers.

Course Outcomes
After the completion of this course, students will be able to:
1. Analyze the need of compiler for interfacing between user and machine.
2. Explain the role of several phases of compilation process.
3. Create an awareness of the function and complexity of modern compilers.
4. Outline the major concept areas of languages translation and Compiler design.
5. Develop a comprehensive Compiler for a given language.

Syllabus

Module - I
Introduction to Compilers and its Cousins, Structure of a Compiler, Science of building Compiler and its Application, Lexical Analyzer, Input Buffering, Specification and Recognition of Tokens, Introduction to Lex.

Module - II
Introduction to Syntax Analysis, Elimination of Ambiguity, Left Recursion and Left Factoring, Recursive and Non-Recursive Top-Down Parsers, Bottom-up Parsers: Shift Reduce Parser techniques and conflicts, all variants of LR Parsers, Handling Ambiguous grammar in Bottom-Up Parsing, Error handling while parsing, The Parser generator YAAC.

Module - III
Syntax-Directed Definition(SDD), Evaluation Order of SDD’s and its application, Syntax-Directed Translation Schemes and their Implementation.

Module - IV
Intermediate code Generation: Variants of Syntax Tree, Three Address Code, Translation of Expressions, Control flow, Back Patching , Run Time Environment: Storage Organization

Module - V
Code Generation: Issues in its Design, Target Language, Addresses in Target Code, Basic Blocks and Flow Graphs, Optimization of Basic BlocksMachine Independent Optimization: Sources of Optimization, Data Flow analysis.

Text book:
1. Aho A. V., Lam M. S., Sethi R., Ullman J. D., Compilers, Principles, Techniques, and Tool, 2nd Edition, Pearson Education Asia.

Reference books:
1. Fischer C. N., LeBlanc R. J., Crafting a Compiler with C, Pearson Education Asia.
2. Louden K. C., Compiler Construction, Principles and Practice, Thomson, Brooks/Cole.

Course code: CS6011
Course title: **Computer Graphics And Multimedia**
Pre-requisite(s): Programming skills

Course Outcomes

After the completion of this course, students will be able to:
1. To perform visual computations for geometrical drawings.
2. To display 3D objects in a 2D display devices using various graphics data structures.
3. To model 3D objects.
4. To develop simple Graphical User Interface.
5. To implement the concept of graphics for different frameworks.

Module-I

Introduction and Overview of Graphics Systems:- Use of Computer graphics, Video Display Devices, Refresh Cathode-Ray Tubes, Raster and Random Scan Displays, Colour CRT Monitors, Direct View Storage Tubes, Flat Panel Displays, Three-Dimensional Viewing Devices, Stereoscopic & Virtual Reality Systems, Raster and Random Scan Systems, Different Input and Hard Copy Devices, Graphics Softwares.

Module-II

Output Primitives: - Points and Lines, Line Drawing Algorithms (DDA & Bresenham's), Circle and Ellipse Generating Algorithms, Conic Sections.

Module-III

Two-Dimensional Geometric Transformations:- Different types of transformations and their matrix representations, Homogeneous Coordinates, Composite Transformations, transformations between Coordinate Systems, Affine transformations, Window-to-Viewport Coordinate transformation, Clipping-Point, Line, Polygon, Curve and Text Clipping.

Module-IV

Three-Dimensional Concepts and Object Representation:- Three Dimensional Display Methods, Polygon Surfaces, Curved Lines & Surfaces, Quadric Surfaces, Spline Representations, Cubic Spline interpolation methods, Bezier Curves and Surfaces.

Module-V

Three Dimensional Transformations and Viewing: Translation, Rotation, Scaling, Reflection, Shears, Composite Transformations, Projections- Parallel and Perspective, Projection Transformations, Clipping.

Module-VI

Visible Surface Detection Methods: Classification of Visible Surface Detection Algorithms, Back Face Detection, Depth Buffer Method, A-Buffer Method, Scan-Line Method, Depth Sorting Method, BSP-Tree Method & Area Subdivision Method. Polygon- Rendering Methods.

Module-VII

Introduction to Multimedia Systems Design:

An Introduction – Multimedia applications – Multimedia System Architecture – Defining objects for Multimedia systems – Multimedia Data interface standards – Multimedia Databases
Compression & Decompression – Data & File Format standards – Multimedia I/O technologies - Digital voice and audio – video image and animation – Full motion video – Multimedia Authoring & User Interface – Hypermedia messaging -

Text Book:

1. D. Hearn & M.P. Baker - Computer Graphics, 2/e , Pearson Education, New Delhi, 2005.
2. Prabat K Andleigh and Kiran Thakrar, "Multimedia Systems and Design", PHI, 2005.

Reference Books:

1. W.M. Newman. et. al.- Principle of Interactive Computer Graphics, Mc Graw Hill Publication, New Delhi, 1995.
2. S. Harrington -Computer Graphics- A Programming Approach, Mc Graw Hill Publication, New Delhi, 1994.
3. J.D. Foley et. al- A Fundamental of Computer Graphics Addition Wesley, London, 1993.

Course code: CS6109
Course title: **Software Engineering**
Pre-requisite(s): Programming skills

Course Outcomes
After the completion of this course, students will be able to:
1. Explain process models and their suitable applications for creating and maintaining software in different development environment.
2. Apply project management tools and techniques for planning and managing software projects.
3. Choose a suitable requirement engineering process and design approach to apply them for efficient coding of software.
4. Evaluate and develop the software using various testing methods.
5. Use techniques and principles of quality management and software change management.

Module – I
Introduction: Some Definitions, FAQs about software engineering, the evolving role of software, Software characteristics, SW applications
Software Processes: Software process models, Waterfall model, the prototyping model, spiral model, RAD and Incremental model.
Module – II
Project Management: Management activities, Project planning, Project scheduling, Risk Management.
Module – III
Software Requirements: Functional and non-functional requirements, User requirements, System requirements, the software requirements document. IEEE standard of SRS, Quality of good SRS.
Requirement Engineering Process: Feasibility study, Requirements elicitation and analysis, Requirements validation, Requirement management.
Module – IV
Software Design: Design Concepts and Principles, Architectural Design, Object oriented Design, User interface design
UML: Class diagram, Sequence diagram, Collaboration diagram
Module – V
Verification and Validation: Verification and Validation Planning, S/W inspection, static analysis.
Software Testing : Testing functions, Test case design, White Box testing, Black box testing, Unit testing, Integration Testing, System testing, Reliability.
Module – VI
Management: SW cost estimation: Estimation techniques, Algorithmic cost modeling, Project duration and staffing.
Quality Management: Quality assurance and standards, Quality planning, Quality control.
Module – VII
Software Change: Program Evolution Dynamic, S/W Maintenance in detail.

Text Book:
1. I. Sommerville : Software Engineering, Pearson Education Publication, 7th ed.

Reference Books:
1. R. S. Pressman: Software Engineering: A Practitioners Approach, 5th Edn., TMA, New Delhi.
2. J. F. Peters & W. Pedrycz– Software Engineering, John Wiley & Sons,Inc. 2000
3. A. Behforooz & F.J. Hudson – Software Engineering Fundamentals, Oxford Univ. Press, New York, 2000.
4. R.Mall: Fundamentals of Software Engineering. Prentice Hall, 3rd Edn, 2004
5. Naresh Chauhan: Software Testing, Oxford Univ.Press,

Course code: **CS6103**
Course title: **System Programming**
Pre-requisite(s): Operating system, Computer system architecture, Microprocessor

Course Outcomes
After the completion of this course, students will be able to:

1. Support, dependency and limitations of system softwares on available system architecture.
2. Clarity about the concrete view on the theoretical and practical aspects of design system softwares.
3. Understand the working of various system software such as assemblers, linkers, loaders and macro-processors.
4. Abilty to design and develop system softwares (complete life cycle).

Module – I
Background: Introduction, System Software and Machine Architecture, The Simplified Instructional Computer (SIC), Traditional (CISC) machines, RISC Machines.
Module – II & III
Assemblers: Basic Assembler Functions, Machine – Dependent Assembler Features, Machine – Independent Assembler Features, Assembler Design Options, Implementation Examples.
Module – IV & V
Loaders and Linkers: Basic Loader Functions, Machine - Dependent Loader Features, Machine – Independent Loader Features, Loader Design Options, Implementation Examples.
Module – VI
Macro Processors: Basic Macro Processor Functions, Machine – Independent Macro Processor Features, Macro Processor Design Options, Implementation Examples.
Module – VII
Software Engineering Issues: Introduction to Software Engineering Concepts, System Specifications, Procedural System Design, Object – Oriented Design, System Testing Strategies.

Text Book:

1. L. L. Beck – System Software – An Introduction to Systems Programming, 3rd Edn., Pearson Education, New Delhi, 2004.

Reference Book:

1. J.J. Donovan – System Programming, McGraw Hill , New Delhi, 1993.
2. D.M. Dhamdhere – System Programming and Operating Systems, 2nd Edn., Tata McGraw Hill , New Delhi, 2000.

Course code: **CS6106**
Course title: **Compiler Design Lab**
Pre-requisite(s): C Programming, Data Structures, Basics of Automata theory (regular expressions, context free grammar)

Text Book:

1. Compilers: Principles, Techniques and Tools. Alfred.V.Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman. 2nd Edition. Pearson.

Reference Books:

1. The C Programming Language. Kernighan & Ritchie. 2nd Edition. PHI.
2. Flex Programming Manual.
3. Bison Programming Manual.

Course code: CS6110
Course title: **Software Engineering Lab**
Pre-requisite(s): Programming skills

Course Outcomes

After the completion of this course, students will be able to:

1. To prepare an IEEE-SRS from the given case study and draw use case diagrams, DFD, architectural design, and also prepare use case description document.
2. To create a complete UML class diagram for the given case study and apply iterative process to refine the class diagram if there is a change in requirements.
3. To create sequence, collaboration and state chart diagrams to analyze the dynamic behavior of the system.
4. To create activity diagram, component and deployment diagram to analyze the environmental and deployment view of the system..
5. Test the programs using software testing tools Like JUnit,Selenium,LDRA,Jumble.

Text Book:

1. Jason T .Roff,UML :A begineer’s guide,TMH,N.Delhi,Ed:2012
2. M.Blaha,James Rumbaugh,Object Oriented Modeling and Design Using UML,PHI,New Delhi,Ed:2nd,2006.

Course code: **CS6012**
Course title: **Computer Graphics and Multimedia Lab**
Pre-requisite(s): Data Structures and Programming skills

Course Outcomes

After the completion of this course, students will be able to:

1. To understand the 3D transformation of objects into 2D displays.
2. To understand the different kinds of transformations by animating objects.
3. To understand the algorithms on projection, shear, coloring and modeling of various shapes.
4. To understand and apply multimedia effect on objects and realize its importance in the graphics industry.

Text Books:

1. D. Hearn & M.P. Baker - Computer Graphics, 2/e , Pearson Education, New Delhi, 2005.
2. Prabat K Andleigh and Kiran Thakrar, “Multimedia Systems and Design”, PHI, 2005.

Reference Books:

1. W.M. Newman. et. al.- Principle of Interactive Computer Graphics, Mc Graw Hill Publication, New Delhi, 1995.
2. S. Harrington -Computer Graphics- A Programming Approach, Mc Graw Hill Publication, New Delhi, 1994.
3. J.D. Foley et. al- A Fundamental of Computer Graphics Addition Wesley, London, 1993.

Course code: CS5105
Course title: **Soft Computing**
Pre-requisite(s): Programming skills

Course Outcomes

After the completion of this course, students will be able to:

1. Solve numericals on Fuzzy sets and Fuzzy Reasoning.
2. Develop Fuzzy Inference System (FIS).
3. Solve problems on Genetic Algorithms.
4. Explain concepts of neural networks.
5. Develop neural networks models for various applications.

Fuzzy Logic

Module -I

Fuzzy Set Theory: Basic Definition and Terminology, Set Theoretic Operations, MF Formulation and Parameterization, MF of two dimensions, Fuzzy Union, Intersection and Complement.

Module -II

Fuzzy Rules and Fuzzy Reasoning: Extension Principles and Fuzzy Relations, Fuzzy IF THEN Rules, Fuzzy Reasoning.

Module –III

Fuzzy Inference System: Introduction, Mamdani Fuzzy Models, Other Variants, Sugeno Fuzzy Models, Tsukamoto Fuzzy Models.

Genetic Algorithms

Module –IV

Fundamentals of Genetic Algorithms: Basic Concepts, Creation of Offsprings, Encoding, Fitness Functions, Reproduction, Genetic Modelling: Inheritance Operators, Cross over, Inversion and detection, Mutation operator, Bitwise operators.

Artificial Neural Networks

Module -V

Introduction, Architecture, Back Propagation and Feed Forward Networks, Offline Learning, Online Learning.

Module -VI

Supervised Learning of Neural Networks: Introduction, Perceptrons, Adaline, Back Propagation Multilayer Perceptrons, Back Propagation Learning Rules, Methods of Speeding. Radical Basis Function Networks, Functional Expansion Networks.

Module -VII

Unsupervised Learning: Competitive Learning Networks, Kohonen self-organising networks, Hebbian Learning, The Hopfield Network

Text Book :

1. J.S.R. Jang, C.T.Sun and E.Mizutani, "Neuro-Fuzzy and Soft Computing" PHI/Pearson Education, New Delhi 2004.

Reference Books:

1. T. J. Ross, "Fuzzy Logic with Engineering Applications." TMH, New York, 1997.
2. K.S.Ray ,"Soft Computing and Its application", Vol 1, Apple Academic Press.2015
3. K.H.Lee ,"First Course on Fuzzy Theory and App.", Adv in Soft Computing Spinger.2005
4. H.Z.Zimmermann ,"Fuzzy Set Theory and its App " , 4th Ed.Spinger Science,2001
5. S. Rajasekaran & G.A. Vijayalakshmi Pai, PHI, New Delhi 2003

Course code: **CS5106**
Course title: **Soft Computing Lab**
Pre-requisite(s): Programming skills

Course Outcomes

After the completion of this course, students will be able to:
1. Solve numericals on Fuzzy sets and Fuzzy Reasoning.
2. Develop Fuzzy Inference System (FIS).
3. Solve problems on Genetic Algorithms.
4. Explain concepts of neural networks.
5. Develop neural networks models for various applications.

Reference Books:

1. T. J. Ross, “Fuzzy Logic with Engineering Applications.” TMH, New York, 1997.
2. K.S.Ray ,”Soft Computing and Its application”, Vol 1, Apple Academic Press.2015
3. K.H.Lee ,”First Course on Fuzzy Theory and App.”, Adv in Soft Computing Spinger.2005
4. H.Z.Zimmermann ,”Fuzzy Set Theory and its App “ , 4th Ed.Spinger Science,2001
5. S. Rajasekaran & G.A. Vijayalakshmi Pai, PHI, New Delhi 2003

Course code: CS8031
Course title: **Data Mining & Data Warehouseing**
Pre-requisite(s): Object oriented programming concepts.

Course Outcomes

After the completion of this course, students will be able to:

1. Assess the problem and decide what data mining activities are required to obtain the desired objectives.
2. Distinguish between the specific operations, data structures and schema that are used in data warehouses vis a vis databases.
3. Mathematically perform preprocessing operations on datasets to ensure the validity of the data is improved.
4. Design schema and write DMQL corresponding to elementary data warehouses.
5. Understand, analyze and evaluate algorithms for performing common data mining activities like Association rule mining, Classification, Clustering etc.

Module - I

Data Mining : Introduction, Relational Databases, Data Warehouses, Transactional databases, Advanced database Systems and Application, Data Mining Functionalities, Classification of Data Mining Systems, Major Issues in Data Mining.

Module - II

Data Warehouse : Introduction, A Multidimensional data Model, Data Warehouse Architecture, Data Warehouse Implementation, Data Cube Technology, From Data warehousing to Data Mining.

Module - III

Data Processing : Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and concept Hierarchy Generation.

Data Mining Primitives, Languages and System Architecture : Data Mining Primities, DMQL, Architectures of Data Mining Systems.

Module – IV

Concept Description : Data Generalization & Summarization – Based Characterization, Analytical Characterization, Mining class Comparisons, Mining Descriptive Statistical Measures in Large Databases.

Module - V

Mining Association Rules in Large Databases : Association Rule Mining, Single – Dimensional Boolean Association Rules, Multilevel Association Rules from Transaction Databases, Multi Dimensional Association Rules from Relational Databases, From Association Mining to Correlation Analysis, Constraint – Based Association Mining.

Module - VI

Classification and Prediction : Classification & Prediction, Issues Regarding Classification & Prediction, Classification by decision Tree Induction, Bayesian Classification, Classification by Back propagation, Classification based on concepts & Association Rule, Other Classification, Prediction, Classification Accuracy.

Module - VII

Cluster Analysis : Types of Data in Cluster Analysis, Partitioning methods, Hierarchical methods, Density – Based Methods, Grid – Based Methods, Model – Based Clustering Methods, Outlier Analysis.

Mining Complex Types of Data.

Text Books :

1. Jiawei Han & Micheline Kamber - Data Mining Concepts & Techniques Publisher Harcout India. Private Limited.

Reference Books :

1. G.K. Gupta – Introduction to Data Mining with case Studies, PHI, New Delhi – 2006.
2. A. Berson & S.J. Smith – Data Warehousing Data Mining, COLAP, TMH, New Delhi – 2004.
3. H.M. Dunham & S. Sridhar – Data Mining, Pearson Education, New Delhi, 2006.

Course code: **CS7029**
Course title: **Free/Open Source Software**
Class: B. Tech
Semester / Level: VII
Branch: Computer Science & Engineering

Syllabus

Module I
Open Standards for Enterprise Applications Application examples from Banking and e-Governance, Enterprise level requirements, Layered Architecture applications

Module II
Open Standards, Royalties, FOSS Licensing & Business models

Module III
FOSS Development Methodology & Tools,
Collaborative Software Development: FOSS Community structure & dynamics

Module IV
Mailing lists, chat, wiki, messaging, Integrated Development Environments, Version Control (tagging, branching, merging), Issue tracking (bugs, new features), Documentation

Module V
Presentation Layer, Client Environment: Browsers (extensions, cookies, applets), Hand-held devices, Thin clients, Web page design & Client
side scripting (XHTML, WML, AJAX, CSS, XSL)

Module VI
Data store Layer,
Data modeling refresher: RDBMS fundamentals, Dublincore, Database Administration, Access control, Security; Disaster recovers (Backups, Replication)

Module VII
Business Logic Layer Model View Controller (MVC) Framework, Server-side programming (CGI, page tempaltes), Data access, database connectivity, Report generation

Text Book(s):

1. Understanding Open Source Software Development, J Feller and B Fitzgerald, Addison Wesley, 2002
2. Enterprise Application Development using FOSS, Ed. NRCFOSS Team, An NRCFOSS (Anna University) Publication

Course code: **CS7117**
Course title: **Optimization Techniques**
Pre-requisite(s): Fundamentals of Mathematical Modeling and basic computer programming.

Course Outcomes

After the completion of this course, students will be able to:

1. To be able to understand the importance of Operation Research techniques and mathematical modeling in solving practical problems.
2. To be able to formulate decision making problems into mathematical models.
3. To be able to use computer tools in solving real-world problems.

Module - I

Introduction to Linear Programming: Prototype Example, the Linear Programming Model, Assumptions of Linear Programming, Additional Examples, Some Classic Case Studies.

Solving Linear Programming Problems - The Simplex Method: The Essence of the Simplex Method, Setting Up the Simplex Method, the Algebra of the Simplex Method, the Simplex Method in Tabular Form, Tie Breaking in the Simplex Method, Adapting to Other Model Forms, Postoptimality Analysis.

Module - II

The Theory Of The Simplex Method: Foundations of the Simplex Method, The revised Simplex Method, A Fundamental Insight.

Duality Theory And Sensitivity Analysis: The Essence of Duality Theory, Economic Interpretation of Duality, Primal-Dual relationships, Adapting to Other Primal Forms, the Role of Duality Theory in Sensitivity Analysis.

Module - III

Other Algorithms for Linear Programming: The Dual Simplex Method, Parametric Linear Programming, the Upper Bound Techniques, an Interior-Point Algorithm.

Network Optimization Models: Prototype Example, the Terminology of Networks, The Shortest-Path Problem, The Minimum Spanning Tree Problem, The Maximum Flow Problem, The Minimum Cost flow Problem, The Network Simplex Method.

Module - IV

Dynamic Programming: A Prototype Example for Dynamic Programming, Characteristics of Dynamic Programming Problems, Deterministic Dynamic Programming, Probabilistic Dynamic Programming.

Module - V

Integer Programming: Prototype Example, Some BIP Applications, Innovative Uses of Binary Variables in Model Formulation, Some Formulation examples, Some Perspectives on Solving Integer Programming problems, The Branch-and-Bound Technique and Its Application to Binary Integer Programming, A Branch-and-Bound Algorithm for Mixed Integer.

Module - VI

Nonlinear Programming: Sample Applications, Graphical Illustration of Nonlinear Programming Problems, Types of Nonlinear Programming Problems, One-Variable Unconstrained Optimization, Multivariable Unconstrained Optimization, The Karush-Kuhn-Tucker (KKT) Conditions for Constrained Optimization, Quadratic Programming, Separable Programming , Convex Programming.

Module - VII

Queuing Theory: Prototype Example, Basic Structure of queuing Models, Examples of Real Queuing Systems, The role of the Exponential Distribution, the Birth-and-Death Process, Queuing Models Based on the Birth-and Death Process, Queueing Models Involving Non-exponential Distributions.

-

Text Book:

1. S. Hiller & G.J. Lieberman – Operations Research, 8th Edn, TMH, New Delhi–2006.

Reference Books:

1. H.A.Taha – Operations Research, 8/e , Pearson Education , New Delhi-2007.
2. J.K. Sharma – Operations Research, 3/e, Mcmillan , India Ltd, 2007.

Course code: **CS7121**
Course title: **Cryptography & Network Security**
Pre-requisite(s): Computer Networks, Number Theory

Course Outcomes

After the completion of this course, students will be able to:

1. Understand theory of fundamental cryptography, encryption and decryption algorithms, identify the appropriate cryptography scheme & security mechanism for different computing environment and information systems.
2. Explain the basic concepts of symmetric & asymmetric cryptography, Demonstrate a systematic and critical understanding of the theories, principles and practices.
3. Design mathematical models to solve specified security problems. Apply the security techniques in solving security problems in practical systems.
4. Understand the fundamental concepts of different digital signature schemes, Build secure authentication systems by use of message authentication techniques.
5. Apply the cryptographic systems learnt so far to build information and network security mechanisms.

Module - I

Security Services, Mechanisms and Attacks, The OSI Security Architecture, A Model for Network Security. Symmetric Cipher Model, Substitution Techniques, Transposition Techniques, Rotol Machines, Steganography.

Module - II

Simplified DES, Block Cipher Principles, The Data Encryption Standard, The Strength of DES, Differential and Linear Cryptanalysis, Block Cipher Design Principles, Block Cipher Modes of Operation.

Module - III

Finite Fields and Confidentiality : Groups, Rings, and Fields, Modular Arithmetic, Euclid's Algorithm, Finite Fields of the Form GF (p), Polynomial arithmetic, Finite Fields of the Form GF(2"), Placement of Encryption Function, Traffic Confidentially, Key Distribution, Random Number Generation.

Module - IV

Encryption Standard and Ciphers : Evaluation criteria for AES, AES cipher, Multiple encryption and Triple DES, Block ciper Modes of operation, Stream ciphers and RCG.

Module - V

Number Theory and Public-Key Cryptography: Prime Numbers, Fermat's and Euler's Theorems, Testing for Primality, The Chinese Remainder Theorem, Discrete Logarithms, Principles of Public-Key Cryptosystems, The RSA Algorithm,

Module - VI

Message Authentication, Function, Algorithms and Digital System :Authentication Requirements, Authentication Functions, Message Authentication Codes, Hash Functions,

MODULE - VII

Security of Hash Functions and MACs, Secure Hash Algorithm, HMAC, Digital Signatures, Authentication Protocols.

Text Book:

1. W.Stallings : Cryptography and Network Security : Principles and Practice, 4/e Pearson Education, New Delhi, 2006.

Reference Books:

1. B.A. Forouzan – Cryptography and Network Security, TMH, New Delhi, 2007
2. B. Schneier – Applied Cryptography, John Wiley, Indian Edition, 2006.

Course code: **CS7029**
Course title: **Information Security and Assurance**
Class: B. Tech
Semester / Level: VII
Branch: Computer Science & Engineering

Syllabus

Module I
Network Security:
Introduction, Network Attacks and Security Issues, Network Communications, Some Example Security Attacks, Security Attacks, Services, and Architecture, Protection and Prevention, Firewalls and Perimeter Security, Cryptographic Protocols, Detection, Assessment and Response

Module II
Introduction, Identification and Authentication, Password-Based Authentication, Insecure Communication Channels, Challenge-Response Systems, Authentication in Distributed Systems, Access Control, Access Control Based on Subject–Object, Relationships, Protection Matrix Model

Module III
An Information Flow Policy for Confidentiality, Bell-LaPadula Model, Clark-Wilson Model, Role-Based Access Control, Access Control in Distributed Systems, Overview of Relevant Standards, Advanced Approaches, Digital Identity Management, Shibboleth, CardSpace, Higgins Trust Framework

Module IV
Network Survivability:
Introduction, Prevention Techniques, Survivable Network Design and Traffic Restoration Concepts, Typical Network Architecture, Basic Survivability Concepts, Basic Network Management Concepts , Protection versus Restoration, Other Issues, Transport Network Recovery Techniques, Automatic Protection Switching, Ring-Based Survivability, Span Restoration, Shared Path Protection, Shared Path Restoration, p-Cycles, Survivable Network Design Techniques, Multilayer Issues

Module V
System Survivability:
Introduction and Background, Survivability and the Impact of Fault Models, Dependability Considerations, Survivability Considerations, Design for Survivability, Identification of Essential Functionalities, Tolerating Faults, Dealing with Common-Mode Faults, Applying the Notion of Optimality, Decentralized Storage, Survivability of Large Distributed Systems, Borrowing from Well-established Fields, Problem Transformation, Scheduling Problems, Case Study: Autonomous Mobile Agents

Module VI
Taxonomy and Framework for Integrating Dependability and Security:
Introduction, Basic Concepts and Related Work, Dependability, Integration of Dependability and Security, Proposed Taxonomy and Framework, Key Notations of the Feedback Control System Model, Definitions of Basic Concepts of Dependability and Security within the Proposed Framework , Dependability, Security, and their Attributes, Taxonomy of Faults ,The Means to Attain Dependability and Security ,Fault Prevention,Fault Tolerance ,Fault Removal

Module VII
Stochastic Modeling Techniques for Secure and Survivable Systems:

Introduction, Survivability and Security, Analytical Modeling Techniques, Markov Models, Semi-Markov Process, Higher-Level Model Formalisms, Security Modeling, Intrusion-Tolerant Systems, Security Modeling of SITAR Security System, Survivability Modeling, System Description

Text Book(s):

1. Qian Y., Joshi J., Tipper D., Krishnamurty P., “Information Assurance : Dependability and Security in Networked Systems”, 1st Edition, Morgan Kauffman Publication, 2008

Course code: **CS7101**
Course title: **Principles of Programming Languages**
Class: B. Tech
Semester / Level: VII
Branch: Computer Science & Engineering

Syllabus

Module I
Introduction:
Programming Linguistics, Historical Development
Data Abstraction:
Program units, packages, encapsulation, Abstract types, Objects and classes, Implementation notes

Module II
Generic Abstraction:
Generic units and instantiation, Type and class parameters, Implementation notes
Type Systems:
Inclusion polymorphism, Parametric polymorphism, Overloading, Type conversions, Implementation notes

Module III & IV
Control Flow:
Sequencers, Jumps, Escapes, Exceptions, Implementation notes
Concurrency:
Why concurrency, Programs and processes, Problems with concurrency, Process interactions, Concurrency primitives, Concurrent control abstractions
Imperative Programming:
Imperative programming: Key concepts, Pragmatics, Case study: C, Case study: ADA

Module V
Object-Oriented Programming:
Key concepts, Pragmatics, Case study: C++, Case study: JAVA, Case study: ADA95

Module VI & VII
Concurrent Programming:
Key concepts, Pragmatics, Case study: ADA95, Case study: JAVA, Implementation notes
Functional Programming:
Key concepts, Pragmatics, Case study: HASKELL
Logic Programming:
Key concepts, Pragmatics, Case study: PROLOG
Scripting:
Pragmatics Key concepts, Case study: PYTHON

Text Book(s):

1. David A. Watt – Programming Language Design Concepts, John Wiley, India Edition, 2004.

Reference Book(s):

1. R. Sethi and K.V. Viswanatha – Programming Languages: Conupt & Constructs, 2/e, New Delhi-2007.
2. D.Appleby & J.J.Vanda Kopple – Programming Languages : Paradigm and Practice 2/e, TMH, New Delhi-2005.
3. K.C. Louden – Programming Languages : Principles and Practice, 2/e, Thomson Learning India Edition, 2005.

Course code: **CS7107**
Course title: **Digital Image Processing**
Pre-requisite(s): Data Structures**,** Computer Graphics

Course Outcomes

After the completion of this course, students will be able to:

1. Apply the concepts of sampling, quantization in the context of digital images. They should also be able to measure pixel distances and evaluate neighbourhood properties in an image.
2. Represent an image in spatial and frequency domains and transform from one to the other. The student should acquire the requisite mathematical knowledge to perform the transform.
3. Understand the statistical tools used in image enhancement, perform spatial transforms, design masks for enhancing images and apply them in efficient ways.
4. Differentiate between constrained and unconstrained restoration, develop the mathematical model for restoring images and solve the models for special cases.
5. Calculate mathematical parameters related to compression, understand lossy and non lossy compression and compute evaluation criterion for compression methods.

Module - I

Introduction: Background, Digital Image Representation, Fundamental Steps in Image Processing, Elements of a Digital Image Processing System.

Digital Image Fundamentals: Elements of Visual Perception, A Simple Image Model, Sampling and Quantization, Some Basic Relationships between Pixels, Imagining Geometry.

Module - II

Image Transforms: Introduction to the Fourier Transform, The Discrete Fourier Transform, Some Properties of the Two-Dimensional Fourier Transform, Other Separable Image Transforms.

Module - III

Image Enhancement : Spatial Domain Methods, Frequency Domain Methods, Some Simple Intensity Transformations, Histogram Processing, Image Subtraction, Image Averaging, Background, Smoothing Filters, Sharpening Filters, Lowpass Filtering, Highpass Filtering, Generation of Spatial Masks from Frequency Domain Specifications.

Module - IV & V

Image Restoring: Degradations Model - Definitions, Degradation Model for Continuous Functions, Diagonalization of Circulant and Block-Circulant Matrices, Circulant Matrices, Block Circulant Matrices, Effects of Diagonalization on the Degradation Model, Algebraic Approach to Restoration, Unconstrained Restoration, Constrained Restoration, Inverse Filtering – Formulation, Removal of Blur Caused by Uniform Linear Motion, Restoration in the Spatial Domain, Geometric Transformation.

Module - VI & VII

Image Compression: Fundamentals – Coding Redundancy, Interpixel Redundancy, Psychovisual Redundancy, Fidelity Criteria. Image Compression Models – The Source Encoder and Decoder, The Channel Encoder and Decoder. Elements of Information Theory – Measuring Information, The Information Channel, Fundamental Coding Theorems, Using Information Theory. Error-Free Compression – Variable-Length Coding, Bit-Plane Coding, Lossless Predictive Coding. Lossy Compression – Lossy Predictive Coding, Transform Coding.

Text Book:

1. Rafael. C. Gonzalez & Richard E.Woods.- Digital Image Processing, 2/e Pearson Education, New Delhi - 2006

Reference Books:

1. W.K.Pratt.-Digital Image Processing ,3/e Edn., John Wiley & sons, Inc. 2006

2. M. Sonka et.al Image Processing, Analysis and Machine Vision, 2/e, Thomson, Learning, India Edition, 2007.

Course code: **CS7033**
Course title: **Multimedia Technology**
Class: B. Tech
Semester / Level: VII
Branch: Computer Science & Engineering

Syllabus

Module I
Overview:
Introduction, Multimedia presentation and production,Characteristics of a multimedia presentation, Multiple media, Utilities of multisensory perception, Hardware and Software requirements, Uses of multimedia, Promotion of multimedia based content, Steps for creating a multimedia presentation
Digital Representation:
Introduction, Analog representation, Waves, Digital representation, Need for digital representation, Analog to Digital conversion, Digital to Analog conversion, Relation between sampling rate and Bit Depth, Quantization error, Fourier representation, Pulse modulation, Importance and drawbacks of digital representation

Module II
Text:
Introduction, Types of text, Unicode standard, Font, Insertion of text, Text compression, File formats
Images:
Introduction, Image types, Seeing color, Color models, Basic steps for image processing, Scanner, Digital camera, Interface standards, Specifications of digital images, Color management system(CMS), Device independent color models, Gamma and Gamma correction, Image processing software, File formats, Image output on monitor, Image output on printer

Module III
Audio:
Introduction, Acoustics, Nature of sound waves, Fundamental characteristics of sound, Musical note and pitch, PsychoAcoustics, Elements of audio systems, Microphone, Amplifier, Loudspeaker, Audio Mixer, Digital Audio, Synthesizers, Musical instrument digital interface (MIDI), MIDI messages, MIDI connections, General MIDI (GM) specifications, Basics of staff notation, Sound Card, Audio transmission, Audio recording devices, Audio file formats and CODECs, Software audio players, Audio recording systems, Digital audio broadcasting, Audio and Multimedia

Module IV
Video:
Introduction, Analog Video camera, Transmission of video signals, Video signal formats, Television broadcasting standards, Digital Video, Digital Video standards, PC Video, Video recording formats and systems, Video file formats and CODECs, Video editing

Module V
Animation:
Introduction, Historical background, Uses of Animation, Keyframes and Tweening, Types of Animation, Computer assisted animation, Creating movement, Principles of animation, Some techniques of animation, Animation on the web, 3D

animation, Camera, Special effects, Creating animation, Rendering algorithm

Module VI
Compression:
Introduction, CODEC, Types of compression, Types of redundancies, Lossless/Statistical compression techniques, GIF image coding standard, Lossy/Perceptual compression techniques, JPEG image coding standard, MPEG standards overview, MPEG1 Audio, MPEG-1 Video, MPEG2 Audio, MPEG2 Video,MPEG4, MPEG7, Fractals

Module VII
Multimedia Architecture:
Introduction, User interfaces, Windows multimedia support,Hardware support, Distributed multimedia applications, Realtime protocols, Playback Architectures, Streaming technologies, Temporal relationships,
Synchronization,Multimedia database systems(MMDBS), Feature extraction of image, Feature extraction of audio, Feature of extraction of video, Benchmarking of MMDBS

Text Book(s):

1. R. Parekh – Principles of Multimedia, 2nd Edition, TMH, New Delhi, 2006

Reference Book(s):

1. R.Steinmetz & K. Nahrstedt Multimedia: Computing, Communications and Applications, Pearson Edn., New Delhi, 2006.
2. P.K.Andleigh & K.Thakrar Multimedia Systems Design, PHI, New Delhi, 2005

Course code: **CS8029**
Course title: **Parallel and Distributed Systems**
Pre-requisite(s): Data Structure and algorithms, Operating System, Computer Architecture

Course Objectives

This course enables the students:
1. Motivation for study of parallel and distributed computing.
2. Eevolution of parallel algorithms for abstract model of parallel computation.
3. To design and analyse parallel algorithms for parallel architectures.
4. Basics of Message-Passing model of Distributed Systems.
5. Handling problems of synchronization, mutual exclusion, Broadcast/Multicast and Distributed Shared Memory in distributed environment

Course Outcomes

After the completion of this course, students will be able to:

1. Develop and apply knowledge of parallel and distributed computing techniques.
2. Understand and account for models, limitations and fundamental concepts in the area of parallel and distributed computing.
3. Adapt and design algorithms for execution in parallel and distributed settings.
4. Analyse the algorithms for correctness, efficiency, reliability, security and performance.
5. Apply these understandings to example systems and algorithms.

Module-I

Introduction: Computational Demand of Modern Science, Advent of Practical rocessing, Parallel Processing Terminology- Contrasting Pipelining and Data arallelism, Control Parallelism, Scalability, Control-Parallel Approach, Data-Parallel
Approach with I/O.

Module-II

PRAM Algorithm: A Model of Serial Computation, The PARAM Model of Parallel Computation, PARAM Algorithm- Parallel Reduction, Prefix Sums, List Ranking, Preorder Tree Traversal, Merging Two Sorted Lists, Graph Colouring, Problem defining Fast Solutions on PRAMS.

Module-III

Elementary Parallel Algorithm: Classifying MIMD Algorithm, Reduction. Matrix Multiplication: Sequential Matrix Multiplication, Algorithms for Processor Array, Algorithms for Multiprocessors.

Module-IV

Solving Linear Systems: Terminology, Back Substitution, ODD-EVEN Reduction, Gaussian Elimination, The JACOBI Algorithm, The Gauss-Seidel Algorithm, Jacobi Overrelaxation and Successive Overrelaxation, Mulyigrid Methods, Conjugate Gradient.

Module-V

Basic Algorithms in Message-Passing Systems: Formal Models for Message Passing Systems, Broadcast and Convergecast on a Spanning Tree, Flooding and Building a Spanning Tree, Constructing a Depth-First Search Spanning Tree for a
Specified Root, Constructing a Depth-First Search Spanning Tree without a Specified Root.Leader Election in Rings: The Leader Election Problem, Anonymous Rings, Asynchronous Rings, Synchronous Rings.

Module-VI

Mutual Exclusion in Shared Memory: Formal Model for Shared Memory Systems, The Mutual Exclusion Problem, Mutual Exclusion using Powerful Primitives, Mutual Exclusion using Read/Write Registers. A formal Model for Simulations: Problem Specifications, Communication Systems, Processes, Admissibility, Simulations, Pseudocode Conventions.

Module- VII

Broadcast and Multicast: Specification of Broadcast Services, Implementing a Broadcast Service, Multicast in Groups, An Application: Replication Distributed Shared Memory: Linearizability Shared Memory, Sequentially Consistent Shared Memory, Algorithms, Lower Bounds.

Text Books:

1. H.Attiya & J. Welch- Distributed Computing- Fundamentals, Simulations and Advanced Topics,2nd Edn., Wiley India Publication, New Delhi, 2006.

Course code: **CS8111**
Course title: **Natural Language Processing**
Class: B. Tech
Semester / Level: VIII
Branch: Computer Science & Engineering

Syllabus

Module I
Introduction:
Knowledge in Speech and Language Processing, Ambiguity, Models and Algorithms, Language, Thought, and Understanding, The State of the Art and the Near-Term Future
Regular Expressions and Automata:
Regular Expressions, Finite-State Automata, Regular Languages and FSAs

Module II
Word Classes and Part-of –Speech Tagging:
(Mostly) English Word Classes, Tagsets for English, Part-of –Speech Tagging, Rule-Based Part-of –Speech Tagging, Stochastic Part-of –Speech Tagging, Transformation-Based Tagging, Other Issues

Module III
Context-Free Grammars for English:
Constituency, Context-Free Rules and Trees, Sentences-Level Constructions, The Noun Phrase, Coordination, Agreement, The Verb Phrase and Sub categorization, Auxiliaries, Spoken Language Syntax, Grammar Equivalence and Normal Form, Finite-State and Context- Free Grammars, Grammars and Human Processing

Module IV
Parsing with Context-Free Grammars: Parsing as Search, A Basic Top-Down Parser, Problems with the Basic Top-Down Parser, The Early Algorithm, Finite – State Parsing Methods

Module V
Features and Unification: Feature Structures, Unification of Features Structures, Features Structures in the Grammar, Implementing Unification, Parsing with Unification Constraints, Types and Inheritance

Module VI
Representing Meaning: Computational Desiderata for Representations, Meaning Structure of Language, First Order Predicate Calculus, Some Linguistically Relevant Concepts.
Semantic Analysis: Syntax-Driven Semantic Analysis, Attachments for a Fragment of English, Integrating Semantic Analysis into the Early Parser, Idioms and Composionality, Robust Semantic Analysis

Module VII
Discourse: Reference Resolution, Text Coherence, Discourse Structure, Psycholinguistic Studies of Reference and Coherence
Natural Language Generation: Introduction to Language Generation, An Architecture for Generation, Surface Realization, Discourse Planning, Other Issues

Text Book(s):

1. D.Jurafsky & J.H.Martin- Speech and Language Processing, 4th Edn, Pearson Education, 2005

Reference Book(s):

1. J. Allen – Natural Language Understanding, Pearson Education, New Delhi, 2006

Course code: **CS8121**
Course title: **Pattern Recognition**
Pre-requisite(s): Basic Calculus, Elementary Linear Algebra, Theory of Probability and Statistics and Optimization Theory.

Course Outcomes

After the completion of this course, students will be able to:

1. The computer application to different areas of real problems.
2. Classifying the objects using Priori knowledge.
3. Character recognition in automation and information handling.
4. Computer aided diagnosis aiming at assisting doctors in making diagnostic decisions.
5. Building intelligence machines for speech recognition, data mining and biomedical applications.

Module - I

Pattern Recognition Overview: Overview, Pattern Recognition, Classification and Description, Patterns and Feature Extraction, Training and Learning in PR Systems, Pattern Recognition Approaches.

Module - II

Statistical Pattern Recognition: Introduction, The Gaussian case and Class Dependence Discriminate Functions, Extensions, Classifier Performance, RISK and Errors.

Module - III

Supervised Learning: Parametric Estimation and Supervised Learning, Maximum Likelihood Estimation Approach, Bayesian Parameter Estimation Approach, Non – Parametric Approaches, Parzen Windows, K-nn Non-Parametric Estimation. Nearest Neighbour Rule.

Module - IV

Linear Discriminate Functions and The Discrete and Binary Feature Cases: Introduction, Discrete and Binary Classification Problems, Techniques to Directly Obtain Linear Classifiers.

Module - V & VI

Syntactic Pattern Recognition: Overview Quantifying Structure in Pattern Description and Recognitions, Grammar Based Approach and Application, String Generation as Pattern Description. Recognition by String Matching and Parsing. The Cocke-Younger Kasami ((ck) parsing algorithm.

Module - VII

Neural Pattern Recognition: Introduction to Neural Networks, Neural Network Structure from Pattern Recognition Applications. Physical Neural Network. The Artificial Neural Network Model, Neural Network Based Pattern Associators.

Text Book:

1. Robert Schalkoff - Pattern Recognition, Statistical, Structural and Neural Approach, John Wiley, Indian Edition, 200.

Reference Books:

1. R. U. Duda – Pattern Classification, John Wiley, Indian Edition, 2006.

Course code: **CS8113**
Course title: **Computational Geometry**
Class: B. Tech
Semester / Level: VII
Branch: CS

Syllabus

Module I
Geometric Data Structures:
Points, Polygons, Edges, Geometric objects in space, finding intersection of a line and a triangle.

Module II & III
Incremental Insertion:
Finding star shaped polygons, finding convex hulls, point enclosure : The ray – shooting and the signed angle method, line clipping, polygon clipping, triangulating monotone polygons.
Module IV
Incremental Selection:
Different methods for finding convex hulls, removing hidden surfaces, Intersection of convex polygons, Finding Delaney triangulations.

Module V
Plane-sweep algorithms:
Finding the intersections of line segments, finding convex hulls, contour of the union of rectangles, Decomposing polygons into monotone pieces.

Module VI
Divide and conquer Algorithms:
Computing the intersection of half planes, finding the kernel of polygon, finding voronoi regions, Merge Hull, closest points, polygon triangulation.
Module VII
Spatial subdivision Methods:
The Grid method, quad trees, Two-dimensional search tree, removing hidden surfaces.

Text Book(s):

1. Michael J. Laszlo - Computational Geometry and computer graphics in C++, PHI, New Delhi, 1999.

Reference Book(s):

1. Kernighan Brian W. and Ritchie Dennis M., The C Programming Language, Prentice.

Course code: **CS8123**
Course title: **Real Time Systems**
Class: B. Tech
Semester / Level: VIII
Branch: CS

Syllabus

Module I
Basic Real-Time Concepts:
Terminology, Real-Time System Design Issues, Example Real-Time Systems, Common Misconceptions, Brief History.

Module II
Hardware Considerations:
Basic Architecture, Hardware Interfacing, Central Processing Unit, Memory, Input/Output, Enhancing Performance, Other Special Devices, Non-von-Neumann Architectures.

Module III
Real Time Operating Systems:
Real-Time Kernels, Theoretical Foundations of Real-Time Operating Systems, Intertask Communication and Synchronization, Memory Management, Case Study: POSIX.

Module IV
Software Requirements Engineering:
Requirements-Engineering Process, Types of Requirements, Requirements Specification for Real-Time Systems, Formal Methods in Software Specification, Structured Analysis and Design, Object-Oriented Analysis and the Unified Modeling Language, Organizing the Requirements Document, Organizing and Writing Requirements, Requirements Validation and Review.

Module V
Software System Design:
Properties of Software, Basic Software Engineering Principles, The Design Activity, Procedural-Oriented Design, Object-Oriented Design, Appendix: Case Study in Software Requirements Specification for Four-Way Traffic Intersection Traffic Light Controller System.

Module VI
Programming Languages and the Software Production Process:
Introduction, Assembly Language, Procedural Languages, Object-Oriented Languages.

Module VII
Performance Analysis and Optimization:
Theoretical Preliminaries, Performance Analysis, Application of Queuing Theory, I/O Performance, Performance Optimization.

Text Book(s):

1. Phillip A. Laplante- Real-Time Systems: Design and Analysis, John Wiley- India Edition, 2006

Reference Book(s):

1. Rajib Mall- Real Time Systems; Theory and Practice, Pearson Edition, New Delhi-2007.

2. J. W.S.Liu – Real Time Systems, Pearson Education, New Delhi – 2004.
3. C.M. Krishna & K.G. Shiv – Real Time Systems, Mc Graw Hill – 1997.

Course code: **CS8125**
Course title: **Computer Vision**
Class: B. Tech
Semester / Level: VIII
Branch: CS

Syllabus

Module I
Cameras:

Pinhole Cameras, Cameras with Lenses, The Human Eye, Sensing.

Radiometry- Measuring Light:

Light in Space, Light at Surfaces, Important Special Cases, Notes.

Module II
Sources, Shadows and Shading:
Qualitative Radiometry, Sources and Their Effects, Local Shading Models, Application: Photometric Stereo, Interreflections: Global Shading Models.

Module III
Color:
The Physics of color, Human Color Perception, Representing Color, A Model for Image Color, Surface Color from Image Color.

Module IV
Linear Filters:
Linear Filters and Convolution, Shift Invariant Linear Systems, Spatial Frequency and Fourier Transforms, Sampling and Aliasing, Filters as Templates.

Edge Detection:

Noise, Estimating Derivatives, Detecting Edges. .

Module V
Segmentation By Clustering:
What Is Segmentation, Human Vision: Grouping and Gestalt , Applications: Shot Boundary Detection and Background Subtraction, Image Segmentation by Clustering Pixels, Segmentation by Graph-Theoretic Clustering.

Module VI
Segmentation by Fitting a Model:
The Hough Transform, Fitting Lines, Fitting Curves, Fitting as a Probabilistic Inference Problem.

Module VII
Tracking with Linear Dynamic Models:
Tracking as an Abstract Inference Problem, Linear Dynamic Models, Kalman Filtering, Data Association.

Text Book(s):

1. D.A. Forsyth & J. Ponce – Computer Vision: A Modern Approach, Pearson Education, New Delhi- 2003.

Course code: **CS8127**
Course title: **Computing and Complexity Theory**
Class: B. Tech
Semester / Level: VIII
Branch: CS

Syllabus

Module I
The Church-Turing Thesis:
Turing Machines- Formal definition of a Turing machine, Examples of Turing machines; Variants of Turing Machines- Multitape Turing machines, Nondeterministic Turing machines, Enumerators, Equivalence with other models; The Definition of Algorithm- Hilbert's problems, Terminology for describing Turing machines.

Module II
Decidability:
Decidable Languages- Decidable problems concerning regular languages, Decidable problems concerning context-free languages; The Halting Problem- The diagonalization method, The halting problem is undecidable..

Module III
Reducibility:
Undecidable Problems from Language Theory- Reductions via computation histories; A Simple Undecidable Problem; Mapping Reducibility-Computable functions, formal definition of mapping reducibility.

Module IV
Advanced Topics in Computability Theory:
The Recursion Theorem, Self-reference, Terminology for the recursion theorem, applications; Decidability of logical theories- A decidable theory, An undecidable theory; Turing Reducibility; A Definition of Information- Minimal length descriptions, Incompressible strings and randomness.

Module V & VI
Time Complexity:
Measuring Complexity- Big-O and small-o notation, Analyzing algorithms, Complexity relationships among models; The Class P- Polynomial time, Examples of problems in P; The Class NP- Examples of problems in NP, The P versus NP question; NP-completeness- Polynomial time reducibility, Definition of NP-completeness, The Cook-Levin Theorem; Additional NP-complete Problems- The vertex cover problem, The Hamiltonian path problem, The subset sum problem.

Module VII
Space Complexity: Savitch's Theorem, The Class PSPACE, PSPACH-completeness – The TQBF problem, Winning strategies for games, Generalized geography; The Classes L and NL, NL-completeness- Searching in graphs; NL equals coNL.
Intractability: Hierarchy Theorems- Exponential space completeness; Relativization- Limits of the diagonalization method; Circuit Complexity.

Text Book(s):

Michael Sipser – Introduction to the Theory of Computation, 2/e, Thomson Learning – India Edition 2006.

Reference Book(s):

a. R.G.Taylor – Models of Computation and Formal Languages, Oxford University New York, 1998.
b. B.M. Moret – The Theory of Computation, Pearson Education, New Delhi – 2002.

Course code: **CS7127**
Course title: **Software Project Management**
Pre-requisite(s): Fundamentals of software engineering

Course Outcomes
After the completion of this course, students will be able to:
1. To be able to create project plan for software projects effectively.
2. To be able to implement software project plans and handle real-world project management challenges.
3. To be able to develop skills for project scheduling, tracking and controlling software project.
4. To be able to handle activities necessary to successfully complete and close the software Projects.

Module - I
Managing Software Project: Process & Project Management, Project Management and the CMM, Project Management at Infosys, Introduction to CMMI, PCMM.
The Project Planning Infrastructure: The process data base, The process capability Baseline, Process Assets and the Body of Knowledge System.
Module - II
Process Planning: The Infosys Develelopment Process, Requirement Change Management
Effort Estimation & Scheduling: Estimation and Scheduling Concepts, Effort – Estimation, Scheduling.
Module -III
Quality Planning: Quality Concepts, Quantitative quality Management Planning, Defect Prevention Planning.
Risk Management: Concepts of Risks and Risk Management, Risk Assessment, Risk Control, Examples.
Module- IV
Measurement and Tracking Planning: Concepts in measurement, Measurements, Project tracking.
Project Management Plan: Team Management, Customer Communication and Issue Resolution, Structure of the Project Management Plan.
Module- V
Configuration Plan: Concepts in Configuration Management, Configuration Management Process.
Reviews: The Reviews, Review process Data Collection, Monitoring & Control, Intruduction of Reviews & the NAH Syndrome.

Module -VI
Project Monitoring & Control: Project tracing, Milestone Analysis, Activity Level Analysis using SPC, Defect Analysis & Prevention Process Monitoring & audit.
Module -VII
Project Closure: Project closure Analysis.

Text Book:
1. Pankaj Jalote – Software Project Management in Practice, Pearson Education, New Delhi, 2002

Reference Books:
1. B.Huges and M.Cotterell – Software Project Management, 3/e, Tata Mcgraw Hill, New Delhi, 2004.
2. Pankaj Jalote – CMM in Practice, Pearson Education, New Delhi, 2002

3. W. Humph Grey – Managing the Software Process, Addition – Wesley, 1989.

Course code: **CS7123**
Course title: **Bio-Informatics**
Pre-requisite(s): Basics of Probability and Statistics, Basics of various data structures, Techniques of Algorithm design, Theory of Permutation and Combination, Basics of Biology, Matrix Algebra.

Course Outcomes

After the completion of this course, students will be able to:

1. Design drugs to control human disease.
2. Solve challenges in biological problems through various algorithms.
3. Derive rules for biological phenomenon.
4. Focus in redundant discoveries.
5. Awareness/Creation of web resources for biological problems.

Module - I

Molecular Biology and Biological Chemistry: The Generic Material: Nucleotides, Orientation, Base Pairing, The Central Dogma of Molecular Biology, Gene Structure and Information Content: Promoter Sequences, The Genetic Code, Open Reading Frames, Introns and Exons, Protein Structure and Function: Primary Structure, Secondary, Tertiary, and Quaternary Structure, The Nature of Chemical Bonds: Anatomy of an Atom, Valence, Electronegativity, Hydrophilicity and Hydrophobicity, Molecular Biology Tools: Restriction Enzyme Digests, Gel Electrophoresis, Blotting and Hybridization, Cloning, Polymerase Chain Reaction, DNA Sequencing, Genomic Information Content: C-Value Paradox, Reassociation Kinetics.

Module - II

Data Searches and Pairwise Alignments: Dot Plots, Simple Alignments, Gaps: Simple Gap Penalties, Origination and Length Penalties, Scoring Matrices, Dynamic Programming: The Needleman and Wunsch Algorithm, Global and Local Alignments: Semiglobal Alignments, The Smith-Waterman algorithm, Database Searches: BLAST and Its Relatives, FASTA and Related Algorithms, Alignment Scores and Statistical Significance of Database Searches, Multiple Sequence Alignments.

Module - III

Substitution Patterns: Estimating Substitution Numbers: Jukes-Cantor Model, Transitions and Transversions, Kimura’s Two-Parameter Model, Models With Even More Parameters, Substitutions Between Protein Sequences, Variations in Evolutionary Rates Between Genes.

Module -IV

History of Molecular Phylogenetics: Advantages to Molecular Phylogenies, Phylogenetic Trees: Terminology of Tree Reconstruction, Rooted and Unrooted Trees, Gene vs. Species Trees, Character and Distance Data, Distance Matrix Methods: UPGMA, Estimation of Branch Lengths, Transformed Distance Method, Neighbor’s Relation Method, Neighbor-Joining Methods, Maximum Likelihood Approaches, Multiple Sequence Alignments.

Module - V

Character – Based Methods of Phylogenetics: Parsimony: Informative and Uninformative Sites, Unweighted Parsimony, Weighted Parsimony, Inferred Ancestral Sequences, Strategies for Faster Searches: Branch and Bound, Heuristic Searches, Consensus Trees, Tree Confidence: Bootstrapping, Parametric Tests, Comparison of Phylogenetic Methods, Molecular Phylogenies: The Tree of life, Human Origins.

Module - VI

Genomics and Gene Recognition: Prokaryotic Genomes, Prokaryotic Gene Structure: Promoter Elements, Open Reading Frames, Conceptual Translation, Termination Sequences, GC Content in Prokaryotic Genomes, Prokaryotic Gene Density, Eukaryotic Genomes, Eukaryotic Gene Structure: Promoter Elements, Regulatory Protein Binding Sites, Open Reading Frames: Introns and Exons, Alternative Spicing, GC Content in Eukaryotic Genomes: CpG Islands, Isochores, Codon Usage Bias, Gene Expression: cDNAs and ESTs, Serial Analysis of Gene Expression, Microarrays.

Module -VII

Protein and RNA Structure Prediction: Amino Acids, Polypeptide Composition, Secondary Structure: Backbone Flexibility, Accuracy of Predictions, The Chou-Fasman and GOR Methods, Tertiary and Quaternary Structure: Hydrophobicity, Disulfide Bonds, Active Structures vs. Most Stable Structures, Algorithms for Modeling Protein Folding: Lattice Models, Off-Lattice Models, Energy Functions and Optimization, Structure Prediction: Comparative Modeling, Threading : Reverse Protein Folding, Predicting RNA Secondary Structures.

Text Book:

1. D.E. Krane & M.L. Raymer - Fundamental Concepts of Bioinformatics, Pearson Education, New Delhi-2003.

Reference Books:

1. S.C. Rastogi et.al.- Bioinformatics: Methods and Applications, PHI, New Delhi-2005.
2. V.R. Srinivas - Bioinformatics: A Modern Approach, PHI, New Delhi-2005.
3. A.M. Lesk – Introduction to Bioinformatics, Oxford (Indian Edn), New Delhi-2004.

Course code: **CS7125**
Course title: **Object Oriented Analysis and Design**
Class: B. Tech
Semester / Level: VII
Branch: CS

Syllabus

Module I
What Is Object-Orientation:
Introduction, Basic Concepts, The Origins of Object-Orientation, Object-Oriented languages Today.
Modelling Concepts:
Introduction, Models and diagrams, Drawing Activity Diagrams.

Module II & III
Requirements Capture:
Introduction, User Requirements, Fact Finding Techniques, User Involvement, Documenting Requirements, Use Cases, Requirements Capture and Modelling.
Requirements Analysis:
Introduction, What Must a Requirements Model Do, Use Case Realization, The Class Diagram, Drawing a Class Diagram, CRC (Class Responsibility Collaboration) Cards, Assembling the Analysis Class Diagram.

Module IV
Object Interaction:
Introduction, Object Interaction and Collaboration, Interaction Sequence Diagrams, Collaboration Diagrams, Model Consistency.
Specifying Operations:
Introduction, The Role of Operation Specifications, Contracts, Describing Operation Logic, Object Constraint Language, Creating an Operation Specification.

Module V
Specifying Control:
Introduction, States and Events, Basic Notation, Further Notation, Preparing a Statechart, Consistency Checking, Quality Guidelines, Summary.
Moving Into Design:
Introduction, How is Design Different from Analysis, Logical and Physical Design, System Design and Detailed Design, Qualities and Objectives of Analysis and Design, Measurable Objectives of Analysis and Design, Measurable Objectives in Design, Planning for Design.

Module VI
System Design:
Introduction, The Major Elements of System Design, Software Architecture Concurrency, Processor Allocation, Data Management Issues, Development Standards, Prioritizing Design Trade-offs, Design for Implementation.
Object Design:
Introduction, Class Specification, Interfaces, Criteria for Good Design, Designing Associations, Integrity Constraints, Designing Operations, Normalization

Module VII
Design Patterns:
Introduction, Software Development Patterns, Documenting Patterns-Pattern Templates, Design

Patterns, How to Use Design Patterns, Benefits and Dangers of Using Patterns.

Designing Boundary Classes:

Introduction, The Architecture of the Presentation Layer, Prototyping the User Interface, Designing Classes, Designing Interaction with Sequence Diagrams, The Class Diagram Revisited, User Interface Design Patterns, Modelling the Interface Using Statecharts.

Text Book(s):

1. S.Bennett, S.Mc Robb and R.Farmer – Object- Oriented Systems Andlysis and Design U**sing** UML 2nd **edn,** TMH, New Delhi – 2007.

Reference Book(s):

1. M.Blaha and J.Runbangh – Object- Oriented Modeling and Design with UML 2./e, Pearson Education, New Delhi,2007.
2. J.W. Satzinger, B.R. Jackson and S.D. Burd – Object –Oriented Analysis and Design, Thomson Learning, India Edition, 2007.
3. G. Booch – Object Oriented Analysis and Design with Applications,2/e, CA;Benjamin/Cumming,1994.