SHAILABALA WOMEN'S AUTONOMOUS COLLEGE

Syllabus

for

Masters in Computer Science

(2-Years Programme)

P. G. DEPARTMENT OF COMPUTER SCIENCE
SHAILABALA WOMEN'S AUTONOMOUS COLLEGE
CUTTACK

2023-24

SHAILABALA WOMEN'S AUTONOMOUS COLLEGE

Syllabus for Masters in Science (Computer Science)

(Applicable for Students Taking Admission from the Session 2023-24)

Objective of the Course

The objective of the M.Sc in Computer Science curriculum is to equip the students with the ability to analyze varieties of real-life problems and develop computer based solutions for effective and efficient computer based solutions. Keeping in view the requirements of the evolving software industry and also to provide a foundation for higher studies in Computer Science, efforts has been made in the choice of subjects to balance between theory and practical aspects of Computer Science.

Outcomes of the Course

On successful completion of this course a student can pursue a career as a technical professional in Software Industries and Government Organizations. Additionally, students can go for higher studies and continue their career in research in the core and applied fields of Computer Science and Applications.

Eligibility Criteria

The candidate should have passed BCA/ Bachelor Degree in Computer Science Engineering or equivalent Degree OR Passed B.Sc./ B.Com./ B.A. with Mathematics at 10+2 level or at Graduation Level (with additional bridge courses if required). Obtained at least 50% marks (45% marks in case of candidates belonging to reserved category) in the qualifying Examination.

SHAILABALA WOMEN'S AUTONOMOUS COLLEGE

PG SYLLABUS STRUCTURE

DEPARTMENT OF COMPUTER SCIENCE (W.E.F.2023-24)

Semester-I								
Sl. No.	**Nature of Course**	**Course Code**	**Paper Title**	**Units**	**Credits**	**Marks**		
						Mid-Sem.	**End-Sem.**	**Total**
1	Hard Core	HC-101	Discrete Mathematical Structure	5	5	30	70	100
2	Hard Core	HC-102	Computer System Architecture	5	5	30	70	100
3	Hard Core	HC-103	Database Systems Implementation	5	5	30	70	100
4	Hard Core	HC-104	a) Database Systems Practical b) Python Practical	--	5	30	70	100
5	Allied Core	AC-101	Computer Application (Course to be offered by e-learning center)	3	3	Mid-Sem 10 + Practica1 10 = 20 marks	30	50
	Total				**23**	**135**	**315**	**450**

Semester-II								
Sl. No.	**Nature of Course**	**Course Code**	**Paper Title**	**Units**	**Credits**	**Marks**		
						Mid-Sem.	**End-Sem.**	**Total**
6	Hard Core	HC-201	Data structure & Algorithm	5	5	30	70	100
7	Hard Core	HC-202	Operating Systems	5	5	30	70	100
8	Hard Core	HC-203	Theory of Computation	5	5	30	70	100
9	Hard Core	HC-204	a) Data Structure & Algorithm Practical b) Operating System Practical	--	5	30	70	100
10	Core Elective	CE-201	Artificial Intelligence **OR** Data Science	5	5	30	70	100
11	Open Elective	OE-201	E-Commerce **OR** MOOCs (From SWAYAM/ NPTEL etc.)	--	4	--	50	50
	Total				**29**	**150**	**400**	**550**

Semester-III								
Sl. No.	**Nature of Course**	**Course Code**	**Paper Title**	**Units**	**Credits**	**Marks**		
						Mid-Sem.	**End-Sem.**	**Total**
12	Hard Core	HC-301	Computer Networks	5	5	30	70	100
13	Hard Core	HC-302	Software Engineering	5	5	30	70	100
14	Hard Core	HC-303	a) Computer Networks Practical b) Software Engineering Practical	--	5	30	70	100
15	Core Elective	CE-301	Compiler Design **OR** Graph Theory	5	5	30	70	100
16	Core Elective	CE-302	Cloud Computing **OR** Soft Computing	5	5	30	70	100
17	Field Internship	FI-201	Field Internship	--	3	--	50	50
	Total				**28**	**150**	**400**	**550**

Semester-IV								
Sl. No.	**Nature of Course**	**Course Code**	**Paper Title**	**Units**	**Credits**	**Marks**		
						Mid-Sem.	**End-Sem.**	**Total**
18	Hard Core	HC-401	Applied Cryptography	5	5	30	70	100
19	Hard Core	HC-402	a) Java Practical b) Applied Cryptography Practical	--	5	30	70	100
20	Hard Core	HC-403	Dissertation	--	5	--	100	100
21	Core Elective	CE-401	Data Mining **OR** Internet of Things	5	5	30	70	100
22	Allied Core	AC-401	Women and Society (For All PG Subjects/ Programs)	3	3	15	35	50
	Total				**23**	**105**	**345**	**450**

Summary

HC-Hard Core	14 x100	1400
CE-Core Elective	4 x100	400
OE-Open Elective	1x50	50
AC-Allied Core	2x50	100
FI-Field Internship	1x50	50
Total Marks:		**2000**

Summary

Semester	Credits	Total Marks
Sem-I	23	450
Sem-II	29	550
Sem-III	28	550
Sem-IV	23	450
TOTAL	**103**	**2000**

SEMESTER-I HC-101 [30+70= 100 Marks]

DISCRETE MATHEMATICAL STRUCTURE

Course Objectives

- To learn the concepts of mathematical logic and various proof methods.
- To understand the concepts of sets, relations, functions, chains, and lattices.
- To solve counting problems.
- To understand the fundamentals in graph theory.
- To introduce basic structures of abstract algebra.

Learning Outcomes

Upon successful completion of this course, students will be able to:

- apply mathematical logic to solve problems and prove theorems.
- understand sets, relations, functions and discrete structures.
- solve counting problems by applying counting techniques, permutations, combinations, pigeonhole principle.
- understand the concepts of graph theory.
- understand the algebraic structure: Group, Ring, Field.

UNIT I

Propositional Logic, Propositional equivalences, Rules of Inference for Propositional Logic. Predicates and Quantifiers, Nested Quantifiers, Rules of Inference for Quantified Statements. Methods of Proof: Direct Proofs, Proof by Contraposition, Proofs by Contradiction.

UNIT II

Sets, Set Operations, Set Identities. Functions, One-to-One and Onto Functions, Inverse Functions and Compositions of Functions. Relations, Properties of Binary Relations, Composition of Relations, Equivalence Relations, Equivalence Classes and Partitions, Partial Orderings, Chain, Hasse Diagrams, Lattices.

UNIT III

Mathematical Induction, Strong Induction. The Basics of Counting, Principle of Inclusion-Exclusion, The Pigeonhole Principle. Permutations and Combinations.

UNIT IV

Introduction to Graphs, Graph Terminology, The Handshaking Theorem, Special Simple Graphs (Complete Graphs, Cycles, Wheels, Bipartite Graphs). Representing graphs, Graph Isomorphism.
Euler Paths and Circuits, Hamilton Paths and Circuits, Planar Graphs, Euler's Formula for Planar Graphs.

UNIT V

Groups, Subgroups, Cosets and Lagrange's Theorem. Codes and Group codes. Ring, Integral Domains and Fields.

Text Books

1. Kenneth H. Rosen, Discrete Mathematics and its Applications, McGraw Hill International.
2. C.L. Liu, "elements of Discrete Mathematics", McGraw Hill International.

SEMESTER-I HC-102 [30+70=100 Marks]

COMPUTER SYSTEM ARCHITECTURE

COURSE OBJECTIVES:

- To understand the structure, function and characteristics of computer systems.
- To understand the design of the various functional units and components of computers.
- To identify the elements of modern instructions sets and their impact on processor design.
- To explain the function of each element of a memory hierarchy.
- To identify and compare different methods for computer I/O.

LEARNING OUTCOMES:

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

- understand the advanced concepts of computer architecture.
- analyze to the major differentials of RISC and CISC architectural characteristics.
- investigate modern design structures of Pipelined and Multiprocessors systems.
- acquainted with recent computer architectures and I/O devices, as well as the low-level language required to drive/manage these types of advanced hardware.
- prepare selected reports that imply some emergent topics supporting material essence.

UNIT I

Computer Function and Interconnection: Computer Components, Computer Function. Interconnection Structures, Bus Interconnection, PCI. Cache Memory: Computer Memory System, Cache Memory Principles, Elements of Cache Design.Pentium-4 Cache Organization.

UNIT II

External Memory: Magnetic Disk, RAID, Optical Memory, Magnetic Tape, External Devices. I/O Module, Programmed I/O, Interrupt-Driven I/O, Direct Memory Access. I/O Channels and Processors, Fire Wire and InfiniBand.

UNIT III

CPU Structure and Function: Processor Organization, Register Organization, Instruction Cycle, Instruction Pipelining. Reduced Instruction Set Computer (RISC): Instruction Execution Characteristics, Use of a large Register File, Compiler-Based Register Optimization, Reduced Instruction Set Architecture, RISC Pipelining, MIPS R4000, SPARC, RISC Versus CISC Controversy.

UNIT IV

Instruction Level Parallelism and Superscalar Processors, Overview and Design Issues of Pentium-4.
IA-64 Architecture: Motivation, General Organization, Prediction, Speculation, and Software Pipelining.
IA-64 Instruction Set Architecture, Itanium Organization.

UNIT V

Parallel Organization: Multiple Processors Organizations, Symmetric Multiprocessors, Cache Coherence and MESI Protocol. Clusters, Non-Uniform Memory Access (NUMA), Vector Computation.

Text Books

1. Stalling W.Computer Organization and Architectrure**.** (PHI)
2. C. Hamacher G. Vranesic, S. ZAky – Computer Organization McGraw Hill 1996

References

1. M. M. Mano – Computer System Architecture, 3rd Edition, PHI 1993
2. K. Hwang – Advanced Computer Architecture, McGraw Hill, 1993

SEMESTER-I_ HC-103 [30+70=100 Marks]

DATABASE SYETEMS IMPLEMENTETAION

Course Objectives

The objective of the course is to present an introduction to database management systems, with an emphasis on how to organize, maintain and retrieve data efficiently from a Data Base Management System (DBMS).

Learning Outcomes

Upon successful completion of this course, students will be able to:

- Describe the fundamental elements of relational database management systems**.**
- Design ER-models to represent simple database application scenarios and convert the ER-model to relational tables, populate relational database and formulate SQL queries on data.
- Improve the database design by normalization**.**
- Familiar with query processing and transaction processing in data base systems**.**
- Understand various Concurrency Control techniques implemented in data base systems.

UNIT I: Introduction to Database, 3-Schema Architecture and Data independence, Schema, and instances.
Conceptual Modeling and Database Design: Entity Relationship (ER) Model: Entity Types, Entity sets, Attributes, keys, Relationship types, Relationship Sets, Roles and structural constraints, Weak entity types, Refining the ER design for the company database, ER diagram Naming conventions and design issues.
ER to relational mapping and Enhanced Entity-Relationship (EER) and object modeling subclasses.
UNIT II: Relational Algebra and Calculus: Relational Algebra operations, Tuple relational calculus, Domain relational calculus.
SQL- The Relational Database Standard: Data Definition, Constraints and schemas, Insert, Delete and Update statement in SQL.
SQL queries: Basic and complex SQL queries, Aggregate functions.
UNIT III: Database Design Theory: Functional Dependencies, Armstrong's Axioms.
Closure of attributes, Dependency preservation, Lossless design.
Normalization: Normal Forms based on Primary Keys, Second and third Normal Forms, Boyce-Codd Normal Form.
UNIT IV: Query processing and Optimization: Translating SQL queries into relational algebra, Basic diagram for executing query operations, Using Heuristics in query optimization.
Transaction processing concepts: Introduction, Transaction and system concepts, Desirable properties of transaction. Schedules and recoverability: Types of Schedule, Serializability of schedules, Checking serializability of schedules.
UNIT V: Concurrency Control Techniques: Locking techniques for concurrency control, Concurrency control based on time stamp ordering, Multi version concurrency control techniques, Validations concurrency control techniques.
Database Security and Authorization: Introduction to database security issues, Discretionary access control based on granting and revoking privileges, Mandatory access control for multilevel security.
Distributed Database Systems: Client Server architecture, Distributed database concepts, Data fragmentation, Replication and allocation technique for distributed database, Types of distributed database systems.

Text Books

1. Ramez Elmasri and Shamkant B. Navathe, *"Fundamentals of Database Systems"*, Pearson Education, 7th Ed., 2016.
2. Rajeeb C. Chatterjee, *"Learning Oracle SQL and PL/SQL: A simplified Guide",* PHI Learning Private Limited, 2012.

Reference Books

1. A. Silberschatz, H.F. Korth, S. Sudarshan, *"Database System Concepts",* McGraw Hill, 7th Ed., 2021.
2. Raghu Ramakrishnan and Johannes Gehrke, *"Database Management Systems",* McGraw Hill, 3rd Ed., 2014.

(a)Database Systems Practical
(b) Python Practical

Course Objectives

1. Learn how to build a database and query through it using ORACLE.
2. Understand the SQL Commands
3. Demonstrate working on multiple tables using joins
4. Learn to write simple PL/SQL programs.
5. Emphasize hands-on experience working with all ORACLE.
6. To write, test, and debug simple Python programs.
7. To implement Python programs with conditionals and loops.
8. Use functions for structuring Python programs.

Learning Outcomes

Upon successful completion of this course, students will be able to:

1. Ability to understand the various kinds of SQL commands
2. Demonstrate the operation on database table.
3. Ability to make customized query efficiently on a database.
4. Ability to apply query techniques for realistic data.
5. Write, test, and debug simple Python programs.
6. Ability to understand Python code, develop medium difficulty applications in Python

List of Experiments

a) Database System Practical

1. Creating/Altering/Deleting Tables.
2. Retrieving data using SQL Queries.
3. Creating tables with constraints.
4. Working on Multiple tables with join.
5. WAP a PL/SQL program to find the factorial of a number.
6. WAP a PL/SQL program to print the Fibonacci series upto n terms.

b) Python Practical

1. Write a python program to convert temperature from Celsius to Fahrenheit and vice versa
2. Write a python program to find largest of three numbers.
3. Write a program to compute distance between two points taking input from the user (Pythagorean Theorem).
4. Write a python program to print prime numbers less than 100.
5. Write a python program to find factorial of a number using recursion.
6. Create a list and perform the following methods
   a) insert( )
   b) remove( )
   c) append( )
   d) len( )
   e) pop( )
   f) clear( )

SEMESTER II HC-201 [30+70= 100 Marks]

DATA STRUCTURE & ALGORITHM

Course Objectives

- To design algorithms and analyze the efficiency.
- To learn about different data structures.
- To make students understand the limitations of algorithms and to get familiar with NP-Complete problems.

Learning Outcomes

Upon successful completion of this course, students will be able to:

- design and analyze algorithms using techniques like divide and conquer, greedy and dynamic programming.
- use standard data structures like heaps, trees and graphs for designing algorithms.
- understand the notions of P, NP and NP complete.
- understand the NP complete reductions.

UNIT I

Introduction to algorithms, Analysis of Insertion sort. Asymptotic Notations, Divide and Conquer Approach, Merge Sort, Recurrence Relations. Solving Recurrences: Substitution methods, Recursion tree method, and Master method.

UNIT II

Quick Sort, Heap Sort. Hash Tables, Hash Functions, Chaining, Open Addressing: Linear probing, Quadratic probing and Double hashing. Binary search trees, AVL tree, B-trees.

UNIT III

Representation of Graphs, Breadth-First Search, Depth-First search, Topological sort, Minimum spanning trees: Prim's and Kruskal's Algorithm. Single source shortest paths: The Bellman-Ford algorithm, Dijkstra's algorithm.

UNIT IV

Dynamic programming: Matrix Chain multiplication, Longest Common Subsequence. Greedy algorithms: Activity selection problem, Human codes. Amortized analysis: Aggregate Analysis, The Accounting Method, The Potential Method.

UNIT V

Decision Problems vs. Optimization Problems, Polynomial time, Polynomial-time Verification, Polynomial-time Reduction. Complexity Classes: P, NP, co-NP, NP-hard and NP-complete. Example of NP-Complete Problems (Satisfiability problem, 3-CNF satisfiability problem, Clique problems, Vertex cover problem, Hamiltonian cycle problem, Travelling salesman problem). NP-Completeness proofs: Clique to vertex cover problem reduction, Hamiltonian cycle to Travelling salesman problem reduction.

Text Books

1. Introduction to Algorithms, T.H. Corman, C.E.Leiserson, R.L.Rivest and C. Stein
2. Algorithm Design, Jon Kleinberg, Éva Tardos.

SEMESTER II HC-202 [30+70= 100 Marks]

OPERATING SYSTEMS

Course Objectives

The objective is to learn the concepts of Operating System and Process, illustrate the Scheduling of a processor for a given problem instance, to identify the dead lock situation and provide appropriate solution, to analyze memory management techniques and implement page replacement Algorithm and to understand the implementation of file systems and directories.

Learning Outcomes

Upon successful completion of this course, students will be able to:

- Analyze the concepts of Operating System.
- Analyze the concepts of process, thread and deadlock situation and Illustrate the Scheduling of a processor for a given problem instance.
- Analyze memory management techniques and implement page replacement Algorithm.
- Understand the implementation of file systems and directories.
- Understand the implementation of Disk Structure and illustrate different case studies of different OS

UNIT I

Introduction: Definition of OS, History of DOS and UNIX. Operating System Process: Definition of Process, Process States. Deadlock: Definition of Deadlock, Causes of Deadlock, Avoidance of Deadlock, and Recovery from Deadlock.

UNIT II

Processor management: Preemptive versus non-preemptive scheduling - priorities -deadline scheduling - FIFO - RR – SJF. Distributed computing: Classification of sequential and parallel processing-array processors. Dataflow computers - multiprocessing - fault tolerance.

UNIT III

Memory Management: Background, Logical versus Physical Address space, swapping, contiguous Allocation. Paging, Segmentation. Virtual Memory: Background, Demand paging, performance of Demand paging. Page Replacement: Page Replacement Algorithms. Allocation of frames, Thrashing, Demand Segmentation.

UNIT IV

Device and information management: Operation of moving head disk storage - need for disk scheduling. Optimization - FCFS - SSTF - SCAN - RAM disks - optical disks. Files and database systems: File system - function - organization - allocating and freeing space - file descriptor - access control matrix.

UNIT V

Case studies: DOS - memory management - overlaying - extended and expanded memory - memory allocation. File system and allocation method - internal and external command memory management functions - file management functions. UNIX: Process in UNIX - memory management - I/O systems - file systems and allocation method.

Text Book

1. Willam Stallings, "Operating Systems", 5/e PHI/Pearson Education.
2. Silberschatz, Peterson, Galvin, "Operating System Concepts", Addisson Wessely, Fifth Edition,
3. H. M. Deital, “An introduction to operating systems”, Addison Wisely,

References

1. Charles Crowley, "Operating systems - A Design Oriented Approach", Tata McGraw Hill.
2. Andrew S. Tannenbaum, "Operating Systems: Design and Implementation", PHI, 2006.
3. Mukeshsinghal, Niranjan G shivaratri, “Advanced concepts in operating systems”, MGH.

SEMESTER II HC-203 [30+70= 100 Marks]

THEORY OF COMPUTATION

Course Objectives

- To learn the mathematical foundations of computation including automata theory
- To learn the theory of formal languages and grammars; the notions of algorithm, decidability, complexity, and computability.
- To learn about how really computers works and what kind of activities can be computed mechanically within a computer.

Learning Outcomes

Upon Completing the Course, Students will able to

- Model, compare and analyse different computational models using combinatorial methods.
- Apply rigorously formal mathematical methods to prove properties of languages, grammars and automata.
- Construct algorithms for different problems and argue formally about correctness on different restricted machine models of computation.
- Identify limitations of some computational models and possible methods of proving them.
- Have an overview of how the theoretical study in this course is applicable of application like designing the compilers.

UNIT I: Background materials: Alphabets, Strings, Empty Strings, Sets, Empty Set. Proof Methods: Induction, Contradiction, Hypothesis. Introduction to Theory of Computation: Finite State Machine, Deterministic Finite Automata, Non-deterministic Finite Automata, Equivalence of NFA and DFA, Minimization of Finite state Machine

UNIT II: Regular Expressions and Languages: Introduction, Conversion of DFAs to Regular Expressions, and Vice versa, Pumping Lemma. Closure Properties of Regular Languages: Union, Intersection, Complement, Difference, Reversal, Homomorphism, and Inverse Homomorphism. Context Free Languages: Context Free Languages, Context Free Grammars, Derivation, Ambiguity.

UNIT III: Push Down Automata: Definition of PDAs, Acceptance of PDAs by final state and by empty stack, Conversion of CFG to PDA and vice versa, DPDA and NPDA. Simplification of CFG: Chomsky Normal Form, Greibach Normal Form. The Pumping Lemma for CFL's. Closure properties CFL. Turing Machines: TM Definition and Notation, Instantaneous Descriptions, NTM & DTM. Extensions and Restrictions to Basic TM Model: Multi Tape, Multi Dimensional, Counter machine, Two Stack PDAs.

UNIT IV: Decidability Theory: The Church-Turing Thesis, Universal Turing Machines and TM Encoding, Decidable and Semi-decidable languages: Recursive Enumeration and Decidability, Many-one Reductions, Hardness, Undecidability. Language Properties: Closure Properties, The Diagonalization Language, The Halting Problem, Post's Correspondence Problem, Undecidable Problems from Language Theory, Linear Bounded /automata (LBA).

UNIT V: Complexity Theory: Measuring Complexity, The O, Ω, Θ notations. Time Complexity classes: P, NP, NP – Completeness, Some NP-Complete Problems: SAT, 3-SAT, Hamiltonian Path, Vertex Cover, Independent Set, Space Complexity classes: PSPACE, L, NL.

Text Book

1. Introduction to Automata Theory, Languages & Computation – Hopcroft, Motwani& Ullman.
2. Introduction to Theory of Computation: M.Siper, Thompson Learning

Reference Book

1. Fundamental of the theory of Computation, Priciples and Practice- R.Greenland, H.J.Hoorer.
2. K.L.P. Mishra and N. Chandrasekaran, Theory of Computer Science, PHI.
3. Michael Sipser, Introduction to the Theory of Computation, Thomson Learning.

SEMESTER II **HC-204** **[30+70= 100 Marks]**

(a) Data Structure & Algorithm Practical
(b) Operating System Practical

Course Objectives

- Identify, formulate, and analyze simple/complex problems.
- Design solutions for simple/complex problems and design system processes that meet the specified needs
- Emphasize hands-on experience working with various algorithm associated with Data Structure and Operating Systems

Learning Outcomes

Upon Completing the Course, Students will able to

- For a given sorting problem (Merge/Insertion/Quick) student will able to implement it and analyze the same to determine the time and computation Complexity.
- Student will able to implement program for Graph traversal Algorithm& Hashing Technique.
- Implements various OS and Memory Scheduling Algorithms.

List of Experiments

a) Data Structure & Algorithm

1. WAP in C to implement Quick Sort
2. WAP in C to implement Insertion Sort
3. WAP in C to implement Merge Sort
4. WAP in C to implement Binary Search Tree
5. WAP in C to implement Shortest Path Method (Dijkstra's Method)
6. WAP in C to implement Implementing Hashing (Linear & Quadratic)

b) Operating System Practical

1. Implementing FCFC Scheduling.
2. Implementing SJF Scheduling.
3. Implementing Priority Scheduling.
4. To Simulate MFT Memory Management Technique
5. To Simulate Memory Management Technique
6. To Simulate FIFO page replacement

SEMESTER II CE-201 [30+70= 100 Marks]

ARTIFICIAL INTELLIGENCE

Course Objectives

• To learn the basic concepts of AI principles and approaches.
• To develop the basic understanding of the building blocks of AI.

Learning Outcomes

Upon Completing the Course, Students will able to:

- Have fundamental understanding of the basic concepts of artificial intelligence (AI).
- Apply basic principles of AI in solutions that require problem solving, inference, perception, knowledge representation, and learning.
- Have fundamental understanding of various applications of AI techniques in intelligent agents, expert systems, artificial neural networks and other machine learning models.
- Have knowledge of current scope and limitations, and societal implications of AI.
- Have basic foundation of machine learning.

UNIT I

Introduction to AI: Foundations of AI, History of AI, State of Art. Intelligent agents: Agents and Environments, The concept of Rationality, Structure of Intelligent Agent Problem Solving by Searching: Problem-Solving Agents, Searching for Solutions, Uninformed Search Strategies, Informed (Heuristic) Search Strategies, Heuristic Functions.

UNIT II

Beyond Classical Search: Local Search Algorithms and Optimization Problems, Searching with Nondeterministic Actions, Searching with Partial Observations. Adversarial Search: Games, Optimal Decisions in Games, Alpha–Beta Pruning Constraint Satisfaction Problems: Defining Constraint Satisfaction Problems, Local Search for CSPs, The Structure of Problems.

UNIT III

Logical Agents: Knowledge-Based Agents, The Wumpus World, Logic and Propositional Logic.
First-Order Logic: Syntax and Semantics of First-Order Logic, First-Order Logic, Knowledge Engineering in First-Order Logic. Inference in First-Order Logic: Propositional vs. First-Order Inference, Unification and Lifting, Forward Chaining and Backward Chaining, Resolution.

UNIT IV

Classical Planning: Definition of Classical Planning, Planning Graphs, Other Classical Planning Approaches. Knowledge Representation: Ontological Engineering, Categories and Objects, Events, Mental Events and Mental Objects, Reasoning Systems for Categories. Quantifying Uncertainty: Acting under Uncertainty, Basic Probability Notation, Bayes' Rule and its use.

UNIT V

Learning from Examples: Forms of Learning, Supervised Learning, Learning Decision Trees. Reinforcement Learning: Introduction, Passive Reinforcement Learning, Active Reinforcement Learning, Generalization in Reinforcement Learning. Natural Language Processing: Language Models, Text Classification, Information Retrieval, Information Extraction.

Text Book

1. Stuart Russel& Peter Norvig: Artificial Intelligence A Modern Approach. (Person Education Asia.)
2. Artificial Intelligence – Mishra, PHI
3. D.W. Patterson, "Introduction to A.I and Expert Systems", PHI,
4. Rich & Knight, "Artificial Intelligence", Tata McGraw Hill.
5. W.F. Clocksin and Mellish, "Programming in PROLOG", Narosa Publishing House, 3/e

SEMESTER II CE-201 [30+70= 100 Marks]

DATA SCIENCE

Course Objectives

- To explicate data analysis techniques and quantitative modelling for the solution of real-world business problems.
- To report findings of analysis and effectively present them using data visualization techniques.
- To demonstrate knowledge of statistical data analysis techniques utilized in business decision making.
- To provide insights about the roles of a Data Scientist, such as a developer, an analyst, a statistical expert etc.
- To understand the techniques and tools for transformation of data.

Learning Outcomes

Upon Completing the Course, Students will able to:

- Use proper techniques for understanding and presenting data.
- Learn how to use optimization techniques and software tools for data analysis.
- Sharpen their analytical skills and develop the ability of analyzing data properly.
- Acquire problem solving ability of complex business decisions, quantitative literacy and critical thinking in seeking solutions to complex business problems.

UNIT I

Data Scientist's Tool Box: Turning data into actionable knowledge. Introduction to the tools that will be used in building data analysis software. Version control, markdown, git, GitHub, R, and RStudio.

UNIT II

Overview of R, R data types and objects, reading and writing data. Control structures, functions, scoping rules, dates and times, Loop functions. Debugging tools, Simulation, code profiling.

UNIT III

Getting and Cleaning Data. Obtaining data from the web, from APIs, from databases and from colleagues in various formats. Basics of data cleaning and making data "tidy".

UNIT IV

Exploratory Data Analysis, Essential exploratory techniques for summarizing data, applied for formal modelling commences, Eliminating or sharpening potential hypotheses about the world that can be addressed by the data. Common multivariate statistical techniques used to visualize high-dimensional data.

UNIT V

Make beautiful visualizations using the ggplot2 library. Create commonly used data visualizations for each data type including histograms, scatter plots, and box plots, improve your data visualizations using facets. Create reference variables using appropriate scope, Use the popular diamonds dataset to put your R skills to work.

Text Books

1. Rachel Schutt, Cathy O'Neil, "Doing Data Science: Straight Talk from the Frontline" by Schroff/O'Reilly, 2013.
2. Foster Provost, Tom Fawcett, "Data Science for Business" What You Need to Know About Data Mining and Data-Analytic Thinking by O'Reilly, 2013.

Reference Books

1. John W. Foreman, "Data Smart: Using data Science to Transform Information into Insight" by John Wiley & Sons, 2013.
2. Eric Seigel, "Predictive Analytics: The Power to Predict who Will Click, Buy, Lie, or Die", 1st Edition, by Wiley, 2013.

SEMESTER II **OE-201** **[50 Marks]**

E-COMMERCE

Course Objectives

- To explicate data analysis techniques and quantitative modelling for the solution of real-world business problems.
- To report findings of analysis and effectively present them using data visualization techniques.
- To demonstrate knowledge of statistical data analysis techniques utilized in business decision making.

Learning Outcomes

Upon Completing the Course, Students will able to:

- Use proper techniques for understanding and presenting data.
- Learn how to use optimization techniques and software tools for data analysis.
- Sharpen their analytical skills and develop the ability of analyzing data properly.

UNIT I

Introduction: Overview of Electronic Commerce, Definition of Electronic Commerce, E Business, Categories of E-Commerce Application. Benefits: Potential benefits of E Commerce, Advantages and Disadvantages of E-commerce, Impact of E Commerce on Business, Global trading Environment of E-commerce, The global information distribution networks. Legal and Ethical Issues: The regulatory environment for E Commerce, Legal Issues related to E-commerce.

UNIT II

Business Model of E-commerce, Internet Trading Relationships, Consumer to Business (C2B) Business to Consumer(B2C), Business to Business (B2B), Consumer to Consumer (C2C), Business to Government (B2G). Difference between B2C and B2B E-Commerce, Advantages and Disadvantages.

UNIT III

E-Commerce and EDI, introduction to Electronic Data Interchange (EDI), Benefits, Features of EDI, EDI Model, EDI Standards, Data Transfer and Standards, Cost of EDI, Electronic Funds Transfer, EFT, Combining EDI and EFT, Roles of Bank in EDI.

UNIT IV

Security Overview: Cryptography and Authentication Introduction, Messaging Security Issues, Confidentiality, Integrity, Authentication. Encryption Techniques, Digital Signatures. Good Encryption Practices, Key Management, key management tasks, Additional Authentication Methods. Firewalls: Definition, component, Functionality, securing the firewall, factors considered in securing the firewall, Limitations.

Text Books

1. Electronic Commerce - Security, Risk Management and Control, Greenstein and Feinman.
2. Electronic Commerce: From Vision to Fulfillment, Elias M Award (PHI)

Reference Books

1. E-Commerce and Mobile Commerce Technology- U.S. Pandey, SaurabhShukla, S. Chand
2. Electronic Commerce - A Managerial Perspective, Ed-aim Turban, Jae Lcc, David King Michael Chung, Addison Wesley, New Delhi.

COMPUTER NETWORKS

Course Objectives

The objective is to learn the fundamental principles of Data Communication and Computer Networking and get an exposure on standard OSI layers and protocols.

Learning Outcomes

Upon Completing the Course, Students will able to

- Identify data communications system components, network topologies, and protocols.
- Analyze different features of analog and digital transmission.
- Analyze the working principles and protocols of data link layer.
- Identify and differentiate working principles and protocols of network and transport layer.
- Identify and implement different types of application in application layer.

UNIT I

Overview of Data Communications: Network Topologies, Reference Models: OSI Model. Physical Layer: Analog and Digital Signals, Data Rate Limits, Transmission Impairment. Digital Transmission: Line Coding, Sampling, Transmission Modes.

UNIT II

Analog Transmission: Modulation of Digital Data, Modulation of Analog signals. Multiplexing: Frequency Division Multiplexing (FDM), Wave Division Multiplexing (WDM), Time Division Multiplexing (TDM).

Switching Techniques: Circuit Switching and Packet Switching.

UNIT III

Errors: Types of Errors, Error Detection, Error Correction. Data Link Control and Protocols: Stop-and-Wait ARQ, Go-Back-N ARQ, Selective Repeat ARQ. Wireless LANs: IEEE 802.11 and its architecture

UNIT IV

Host to Host Delivery: IP Addressing and Routing: Unicast, Multicast, Broadcast, and Anycast

Network Layer Protocols: Address Resolution Protocol (ARP), IPV4, Internet Control Message Protocol (ICMP), IPV6. Transport Layer: User Datagram Protocol (UDP), Transmission Control Protocol (TCP)

UNIT V

Client Server Model: Domain Name System (DNS): Electronic Mail (SMTP).

File Transfer: File Transfer Protocol (FTP), Post Office Protocol (POP), HyperText Transfer Protocol (HTTP) and World Wide Web (WWW). Network Security: Authentication, Digital Signatures and Certificates, Firewalls.

Text Books

1. Data Communications and Networking: Behrouz A. Forouzan, Tata McGraw-Hill, 4th Edition.
2. Computer Networks: A. S. Tannenbum, D. Wetherall, Prentice Hall, 5th Edition.

Reference Books

1. Data and Computer Communications: William Stallings, Prentice Hall, 9th Edition
2. Data Communication and Computer Networks: Ajit Pal, PHI Learning Pvt. Ltd

SEMESTER III HC-302 [30+70= 100 Marks]

SOFTWARE ENGINEERING

Course Objectives

- To provide the idea of decomposing the given problem into Analysis, Design, Implementation, Testing and Maintenance phases.
- To provide an idea of using various process models in the software industry according to given circumstances.
- To gain the knowledge of how to gather and specify requirements of the software projects.
- To differentiate different testing methodologies and their utilities.
- To understand and apply the basic project management practices in real life projects
- To enhance the ability to work in a team as well as independently on software projects

Learning Outcomes

On completion of this course, students will be able to:

- Students will be able to decompose the given project in various phases of a lifecycle.
- Students will be able to choose appropriate process model depending on the user requirements.
- Students will be able perform various life cycle activities like Analysis, Design,
- Students will be able to know various processes used in all the phases of the product.
- Students can apply the knowledge, techniques, and skills in the development of a software product.

UNIT I

Computer-Based system Engineering: Emergent System Properties, Systems and their Environment, System Modeling, System Engineering Process, System Procurement.
Software Process: Software Process Models, Process Iteration, Software Specification, Design and Implementation.
Software Validation and Evaluation, Automated Process Support.

UNIT II

Software Requirements: Functional and Non-Functional Requirements, Use Requirements. System Requirements, Software Requirements Document. Requirements Engineering Processes: Feasibility Studies, Requirements Elicitation and Analysis, Requirements Validation, Requirements Management. System Models: Context Models, Behavioral Models, Data models, Object models.

UNIT III

Architectural Design: System Structuring, Control Models, Modular Decomposition, Domain-Specific Architectures.
Distributed System Architectures: Multiprocessor Architectures, Client-Server Architectures, Distributed Architectures, CORBA. Patterns. Dependability: Critical Systems, Availability and Reliability, Safety, Security, Critical Systems

UNIT IV

Specifications: Software Reliability Specification, Safety Specification, Security Specification. Critical Systems Development: Fault Minimization, Fault tolerance, Fault Tolerance Architectures, Safe System Design. Object Oriented software design, Development process.

UNIT V

UML: Unified Modeling Language, Use case diagram, Class diagrams, Essential Interaction diagram, Object diagram, Packages and collaboration. State diagrams, Activity diagrams, Physical diagrams.

Text Books

1. Software Engineering – Sommerville, Addition -Wesley
2. Software Engineering- Pressman, R.S, MGH

Reference Books

1. Fundamental of Software Engineering—Rajib Mall, PHI
2. Software Engineering- Agarwal, K.K & Singh, New Age International.

SEMESTER III **HC-303** **[30+70=100 mark]**

(a) Computer Networks Practical
(b) Software Engineering Practical

Course Objective

- Upon successful completion of this Lab the student will be able to:
- Understand the client/server communication using socket programming
- Simulate ping and use of FTP
- To identify the role of the software in today's world across a few significant domains related to day to day life.
- To understand the suitable software development process model.

Learning Outcomes

On completion of this course, students will be able to:

- Acquire knowledge of Networking Parameters
- Able to do establish Client/Server Communication using Socket
- Learn how to use File Transfer protocol
- Learn to use Software Engineering Tools to develop various auotomated systems.

a) Computer Networks

List of Experiment (Using C/C++/Java)

1. Listing Network Interface Properties
2. Getting the own IP.
3. Implementing Ping Program.
4. Implementation of port scanner
5. Implements TCP Sockets for displaying date & time from server using
6. Implementing File Transfer.

b) Software Engineering (Using sample case study on Bank ATM System)

1. Introduction To software engineering
    a. Defining Problem definition
    b. Writing SRS
        i. Introduction
        ii. Overall Description
        iii. Specification Requirement
        iv. Front End ,Back End Specification
        v. Data Structure
        vi. DFD
        vii. Testing
        viii. Sample screen
2. OO Analysis & Design Using UML (Sample diagrams)
    a. Use Case
    b. Class diagram
    c. State chart diagram
    d. Activity Diagram
    e. Collaboration Diagram
    f. Package Diagram

SEMESTER III CE-301 [30+70= 100 Marks]

DATA MINING

Course Objectives

- To teach the basic principles, concepts and applications of data warehousing and data mining
- To introduce the task of data mining as an important phase of knowledge recovery process
- To familiarize Conceptual, Logical, and Physical design of Data Warehouses OLAP applications and OLAP deployment

Learning Outcomes

On completion of this course, students will be able to:

- design a data mart or data warehouse for any organization
- develop skills to write queries using DMQL
- extract knowledge using data mining techniques
- adapt to new data mining tools.

UNIT I

Introduction to data mining: Motivation, Importance, Definition of Data Mining, Kind of Data, Data Mining Functionalities, Kinds of Patterns, Classification of Data Mining Systems, Data Mining Task Primitives, Integration of A Data Mining System with A Database or Data Warehouse System, Major Issues in Data Mining

UNIT II

Types of Data Sets and Attribute Values, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity. Preprocessing: Data Quality, Major Tasks in Data Preprocessing, Data Reduction, Data Transformation and Data Discretization, Data Cleaning and Data Integration.

Data warehousing and on-line analytical processing: Data Warehouse basic concepts, Data Warehouse Modeling

UNIT III

Data Cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation, Data Generalization by Attribute-Oriented Induction. Data Cube Technology: Efficient Methods for Data Cube Computation, Exploration and Discovery in Multidimensional Databases.

UNIT IV

Mining frequent patterns, associations and correlations: Basic Concepts, Efficient and Scalable Frequent Item set Mining Methods, Are All the Pattern Interesting, Pattern Evaluation Methods, Applications of frequent pattern and associations. Frequent pattern and association mining: A Road Map, Mining Various Kinds of Association Rules, Constraint-Based Frequent Pattern Mining, Extended Applications of Frequent Patterns.

UNIT V

Classification: Basic Concepts, Decision Tree Induction, Bayesian Classification Methods, Rule-Based Classification, Model Evaluation and Selection, Techniques to Improve Classification Accuracy: Ensemble Methods, Handling Different Kinds of Cases in Classification, Bayesian Belief Networks, Classification by Neural Networks, Support Vector Machines, Pattern-Based Classification, Lazy Learners (or Learning from Your Neighbors), Other Classification Methods.

Text Books

1. Jiawei Han, Micheline Kamber, Jian Pei (2012), Data Mining: Concepts and Techniques, 3rd. edition, Elsevier, United States of America.

Reference Books

1. Margaret H Dunham (2006), Data Mining Introductory and Advanced Topics, 2nd edition, Pearson Education, New Delhi, India.
2. Amitesh Sinha (2007), Data Warehousing, Thomson Learning, India.

SEMESTER III CE-301 [30+70= 100 Marks]

COMPILER DESIGN

Course Objectives

- To provide a thorough understanding of the internals of Compiler Design.
- To explore the principles, algorithms, and data structures involved in the design and construction of compilers.
- Topics include context-free grammars, lexical analysis, parsing techniques, symbol tables, error recovery, code generation, and code optimization.

Learning Outcomes

On completion of this course, students will be able to

- Realize basics of compiler design and apply for real time applications.
- Introduce different translation languages
- Understand the importance of code optimization
- Know about compiler generation tools and techniques
- Working of compiler and non-compiler applications
- Compiler for a simple programming language

UNIT I

Introduction to Compiler Design: Introduction to Compiler Design, Phases of Compiler Design, Compiler Construction Tools. Lexical Analysis: DFA, NFA, Regular Expression, Equivalent to NFAs, Minimizing the States of DFA. CFG: Basics of CFG, Normal Forms, Implementation of Lexical Analyser.

UNIT II

Syntax analysis: Top down parsing concepts-Recursive Descent Parsing, FIRST and FOLLOW, LL(1) Grammars Left Recursion Elimination, Top-Down Recursive-Descent parsing. Bottom-Up Parsing: Reduction, Finding Handles, Shift-Reduce Parsing, Conflicts during Shift-Reduce Parsing. LR Parsers: Items and the LR(0) Automaton, The LR-Parsing Algorithms, SLR, CLR, LALR, Ambiguous Grammars.

UNIT III

Syntax-Directed Translation: Syntax Directed Definitions, Evaluating Orders for SDD, Applications of Syntax –Directed Translation Scheme. Intermediate code generation: Variations of Syntax, Three-Address Code- Addresses and Instructions, Quadraples, Triples. Translation of Expression: Operation within Expressions, Incremental Translation, Addressing Array Elements, Translation of Array References.

UNIT IV

Type Checking: Rules for Type Checking, Type Conversions, Overloading of Functions and Operators
Control Flow: Boolean Expressions, Short-Circuit Code, Flow of Control Statements, Control Flow Translation of Boolean Expressions, Backpatching. Run Time Environment: Storage organization, Stack Allocation of Space, Heap Management.

UNIT V

Code Generation: Issues in the Design of Code Generation, Addresses in the Target Code, Basic Blocks and Flow Graphs. Code optimization: Optimization of Basic Blocks, Peephole Optimization, Register Allocation and Assignment. Machine Independent Optimization: The Principle of Source of Optimization, Introduction to Dats Flow Analysis, Introduction to Interprocedural Analysis.

Text Books

1. Alfred Aho, Ravi Sethi, Jeffy D. Ullman, “Compilers - Principles, Techniques and Tools”, Pearson
2. Chattopadhyay Santhanu, “Compiler Design”, PHI.
3. HolubAllen, “Compilers in C”, PHI

SEMESTER III **CE-301** **[30+70= 100 Marks]**

GRAPH THEORY

Course Objectives

- To understand fundamentals of graph theory.
- To study proof techniques related to various concepts in graphs.
- To explore applications of graph theory.

Learning Outcomes

At the end of the course, students will be able to

- Explain the concept of formal graph-theoretic definitions and notations
- Apply basic theoretical concepts in solving real-life problems and address optimization issues

UNIT I

Graph Terminologies, Simple Graph, Multigraph, Special Graphs, Complement Graph, Regular Graph, Bipartite Graphs, Subgraphs: Proper Subgraph, Spanning Subgraph, Induced Subgraph, Vertx-degrees, Handshaking Theorem, Graphic Sequences, Havel-Hakimi Theorem. Graph Representation: Adjacency Matrix, Incidence Matrix, Adjacency List, Graph Isomorphism.

UNIT II

Connected Graphs, Disconnected Graphs and Components, Cut-vertices, Cut-edge, Blocks, Cut-set. Weighted Graphs, Shortest Paths, Dijkstra's Algorithm. Eulerian Graphs, Hamiltonian Graphs.

UNIT III

Trees, Properties of Trees, Pendant Vertices in a Tree, Distance and Centers in a tree, Rooted and Binary Trees. Counting Tree, Spanning Tree, Spanning Tree, Minimum Spanning Tree, Prim's Algorithm, Kruskal's Algorithm. Tree Traversal: Pre-order, Post-order and In-order Traversal.

UNIT IV

Combinatorial Vs Geometric Graphs, Planar Graphs, Kuratwoski Graphs, Detection of Planarity. Geometric and Combinatorial Dual, Thickness and Crossings. Matchings, Matchings and Coverings in Bipartite Graphs, Perfect Matching.

UNIT V

Independent Set, Clique. Graph Coloring, Chromatic Number, Chromatic Partitioning, Greedy Coloring Algorithm, Coloring of Chordal Graphs, Brooks Theorem. The Four Colour Conjecture and Five-Colour Theorem.

Text Books

1. Narsingh Deo, Graph Theory with Applications to Engineering and Computer Science.
2. D.B. West, Introduction to Graph Theory.
3. J. A. Bondy and U. S. R. Murty: Graph Theory.
4. Jon Kleinberg and Eva Tardos, Algorithm Design.

Reference Books

1. T.H. Corman, C.E.Leiserson, R.L.Rivest and C. Stein, Introduction to Algorithms.

CLOUD COMPUTING

Course Objectives

This course gives students an insight into the basics of cloud computing along with virtualization, cloud computing is one of the fastest growing domain from a while now. It will provide the students basic understanding about cloud and virtualization along with security in cloud computing environment.

Learning Outcomes

Upon successful completion of this course, students will be able to

- Explain the core concepts of the cloud computing paradigm: how and why this paradigm shift came about, the characteristics, advantages and challenges of Cloud.
- Develop applications using various models and services in cloud computing.
- Understand virtualization and outline their role in enabling the cloud computing system model.
- Implement different load balancing algorithms in cloud.
- Explain Service Management in Cloud Computing.
- Understand security mechanisms implemented at different levels.

UNIT I: Overview of Computing Paradigm: Recent trends in Computing: Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Cloud Computing. Introduction to Cloud Computing**:** Introduction, History of Cloud Computing, Characteristics of cloud computing, Benefits and limitations of Cloud Computing. Cloud Service Providers (CSPs), Cloud Data Centres, Components of data Centres, Cloud Computing applications.

UNIT II: Cloud Computing Architecture: Comparison with traditional computing architecture (client/server), Services provided at various levels. Service Models: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), How Cloud Computing Works. Deployment Models: Public cloud, Private cloud, Hybrid cloud, Community cloud, Case study of NIST architecture.

UNIT III: Virtualization: Introduction, Characteristics of Virtualized Environments. Taxonomy of Virtualization Techniques, Virtualization and Cloud Computing, Pros and Cons of Virtualization. Load balancing in Cloud Computing: Importance of load balancing, Types of load balancing, Load balancing algorithms.

UNIT IV: Case Studies: Case Study of Service Model using Google App Engine, Microsoft Azure, Amazon EC2. Service Management in Cloud Computing: Service Level Agreements (SLAs), Billing & Accounting. Comparing Scaling Hardware: Traditional vs. Cloud, Types of Scaling, Economics of Scaling.

UNIT V: Cloud Security: Infrastructure Security: Network level security, Host level security, Application level security. Data security and Storage: Aspects of data security, Provider data and its security, Data security issues. Jurisdictional issues raised by data location, Authentication in Cloud, Methods of Authentication.

Text Books

1. Rajkumar Buyya, James Broberg, Andrzej, M. Goscinski, "*Cloud Computing Principles & Paradigms*", Wiley, 2013.
2. Tim Mather, Subra Kumaraswamy, Shahed Latif, "*Cloud Security and Privacy",* O' Reilly, First Edition, 2011.

Reference Books

1. Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi, "*Mastering Cloud Computing*", McGraw Hill Education, 2018.
2. Barrie Sosinsky, "*Cloud Computing Bible*", Wiley, 2011.
3. Nick Antonopoulos, Lee Gillam, "*Cloud Computing: Principles, Systems and Applications*", Springer, 2010.

SOFT COMPUTING

Course Objectives

The primary objective of this course is to provide an introduction to the basic principles, techniques, and applications of soft computing. Upon successful completion of the course, students will have an understanding of the basic areas of Soft Computing including Artificial Neural Networks, Fuzzy Logic and Genetic Algorithms. The aim of this course is to develop some familiarity with current research problems and research methods in Soft Computing by working on a research or design project.

Learning Outcomes

Upon successful completion of this course, students will be able to

- Describe human intelligence and AI and explain how intelligent system works.
- Apply basics of Fuzzy logic and neural networks.
- Discuss the ideas of fuzzy sets, fuzzy logic and use of heuristics based on human experience Relate with neural networks that can learn from available examples and generalize to form appropriate rules for inference systems.
- Describe with genetic algorithms and other random search procedures useful while seeking global optimum in self-learning situations.
- Develop some familiarity with current research problems and research methods in Soft Computing Techniques.
- Apply different ANN techniques to real world problems.

UNIT I: Introduction: Concept of computing systems. Hard computing, Soft computing, Hybrid computing. Optimization and Some Traditional methods: Introduction to Optimization, Traditional methods of optimization. Some Applications of Soft Computing.

UNIT II: Fuzzy Logic: Introduction to Fuzzy logic. Fuzzy sets and membership functions. Operations on Fuzzy sets. Fuzzy relations, rules, propositions, implications and inferences. Defuzzification techniques. Fuzzy logic controller design. Some applications of Fuzzy logic.

UNIT III: Genetic Algorithms (GA): Concept of "Genetics" and "Evolution" and its application to probabilistic search techniques. Basic GA framework and different GA architectures. GA operators: Encoding, Crossover, Selection, Mutation, etc. Solving single-objective optimization problems using GA.
Some specialized GAs: Real-coded GA, Micro-GA

UNIT IV: Multi-objective Optimization problem solving: Concept of multi-objective optimization problems (MOOPs) and issues of solving them. Multi-Objective Evolutionary Algorithm (MOEA).
Non-Pareto approaches to solve MOOPs. Pareto-based approaches to solve MOOPs. Some applications with MOEAs.

UNIT V: Artificial Neural Networks: Biological neurons and it's working. Simulation of biological neurons to problem solving. ANN architectures: Different ANN Architectures. ANN Training: Training techniques for ANNs. Applications of ANNs to solve some real-life problems.

Text Books

1. R. Rajasekaran and G. A. Vijayalakshmi Pai, Neural Networks, Fuzzy Logic, and Genetic Algorithms: Synthesis and Applications, Prentice Hall of India, New Delhi, 2003
2. D. K. Pratihar, Soft Computing, Narosa, 2008
3. L. Fausett, Fundamentals of Neural Networks, Prentice Hall, Upper Saddle River, N.J, 1994.
4. F. Martin, Mcneill, and Ellen Thro, Fuzzy Logic: A Practical approach, AP Professional, 2000.

Reference Books

1. J.-S. R. Jang, C.-T. Sun, and E. Mizutani, Neuro-Fuzzy and soft Computing, PHI Learning, 2009.
2. D. E. Goldberg, Genetic Algorithms in Search, Optimisation, and Machine Learning, Addison-Wesley, Reading, MA, 1989

SEMESTER IV HC-401 [30+70= 100 Marks]

APPLIED CRYPTOGRAPHY

Course Objectives

- Able to understand the basic concepts and goals of Information security such as Confidentiality, Integrity, Authentication, Non-Repudiation, Authorization, and Availability and their relevance in various Contexts.
- Able to understand the classical cryptosystems and techniques used to break them.
- Able to understand the ideas of public key cryptosystems and digital signature schemes.

Learning Outcomes

On completion of this course, students will be able to:

- Identify basic security attacks and services.
- Use of symmetric and asymmetric key algorithms for cryptography.
- Analyse key Management techniques and importance of number Theory.
- Understanding of Authentication functions, the manner in which Message Authentication Codes and Hash Functions works.
- To examine the issues and structure of Authentication Service and Electronic Mail Security.

UNIT I

Introduction, The need of Security, Security approaches, Principles of Security, Types of Security Attacks, Security Services, Security Mechanisms, A model for Network Security. Cryptography: Concepts and Techniques: Introduction, Plain text and Cipher text, Substitution Techniques, Transposition Techniques, Encryption and Decryption. Symmetric and Asymmetric Cryptography, Steganography, Key Range and Key Size, Possible types of Attacks.

UNIT II

Symmetric Key Ciphers: Block Cipher Principles and Algorithms. DES, AES, and Blowfish. Differential and Linear Cryptanalysis, Block Cipher Modes of Operations, Stream Ciphers, RC4, Location and Placement of encryption function, Key Distribution. Asymmetric Key Ciphers: Principles of Public Key Cryptosystems, Algorithms, RSA, Diffie- Hellman, ECC, Key Distribution.

UNIT III

Message Authentication Algorithms and Hash Function: Authentication Requirements, Functions, Message Authentication Codes, Hash Functions. Secure Hash Algorithms, Whirlpool, HMAC, CMAC, Digital Signatures. Authentication Applications: Kerberos, X.509 Authentication Services, Public-Key Infrastructure, Biometric Authentication.

UNIT IV

Web Security: Web Security Considerations, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET). Intruders. Viruses and Firewalls: Intruders, Intrusion Detection, Password Management, Virus and related threats, Countermeasures, Firewall Design Principles, Types of Firewalls. Case Studies on Cryptography and Security: Secure Inter Branch Transactions, Cross Site Vulnerability, Virtual Elections.

UNIT V

Introduction to Information Hiding, Steganography and Watermarking. Fragile watermarking, Reversible watermarking. Importance of digital watermarking, Applications, Properties, Evaluating watermarking systems.

Text Books

1. Cryptography and Network Security – Atul Kahate – TMH.
2. Data Communications and Networking – Behourz A Forouzan

Reference Book

1. Cyber Security Operations Handbook – J.W. Rittiaghouse and William M. Hancok, Elsevier.
2. Cryptography and Network Security Principles and Practice – W.Sta1lings, Pearson Education Asia.

(a) Java Practical
(b) Applied Cryptography Practical

Course Objectives

- Learn various Cipher Techniques like Caesar, Play fair etc
- Learn to implement the various cryptographic algorithms like AES, DES,
- Learn to create simple Java programs using various features of Java

Learning Outcomes

At the end of the course, the student should be able to:

- Implement the cipher techniques
- Develop the various security algorithms
- Use different open source tools for network security and analysis
- To Understand OOP concepts and basics of Java programming.

List of Experiments

a) Java Programming

1. WAP to display Fibonacci series upto n terms
2. WAP to demonstrate String class and its methods.
3. Program to demonstrate use of class and objects.
4. Write a java program to add two integers and two float numbers. When no arguments are supplied, give a default value to calculate the sum. Use function overloading.
5. Program to demonstrate use of inheritance
6. Program to demonstrate use of exception handling (Divide by Zero & Array out of Bound)

b) Applied Cryptography

Implement following Cryptography algorithm using JAVA

1. Caesar Cipher
2. Play fair Cipher
3. Vigenere Cipher.
4. Diffie-Hellman Key exchange
5. DES
6. AES

SEMESTER IV CE-401 [30+70= 100 Marks]

DATA MINING

Course Objectives

- To teach the basic principles, concepts and applications of data warehousing and data mining
- To introduce the task of data mining as an important phase of knowledge recovery process
- To familiarize Conceptual, Logical, and Physical design of Data Warehouses OLAP applications and OLAP deployment
- To impart knowledge of the fundamental concepts that provide the foundation of data mining

Learning Outcomes

Upon successful completion of this course, students will be able to

- design a data mart or data warehouse for any organization
- develop skills to write queries using DMQL
- extract knowledge using data mining techniques
- adapt to new data mining tools.
- apply the techniques of clustering, classification, association finding, feature selection and visualization to real world data

UNIT I: Introduction to data mining: Motivation, Importance, Definition of Data Mining, Kind of Data, Data Mining Functionalities, Kinds of Patterns, Classification of Data Mining Systems, Data Mining Task Primitives, Integration of A Data Mining System with A Database or Data Warehouse System, Major Issues in Data Mining

UNIT II: Types of Data Sets and Attribute Values, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity. Preprocessing: Data Quality, Major Tasks in Data Preprocessing, Data Reduction, Data Transformation and Data Discretization, Data Cleaning and Data Integration.

Data warehousing and on-line analytical processing: Data Warehouse basic concepts, Data Warehouse Modeling

UNIT III: Data Cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation, Data Generalization by Attribute-Oriented Induction. Data CubeTechnology: Efficient Methods for Data Cube Computation, Exploration and Discovery in Multidimensional Databases.

UNIT IV: Mining frequent patterns, associations and correlations: Basic Concepts, Efficient and Scalable Frequent Item set Mining Methods, Are All the Pattern Interesting, Pattern Evaluation Methods, Applications of frequent pattern and associations. Frequent pattern and association mining: A Road Map, Mining Various Kinds of Association Rules, Constraint-Based Frequent Pattern Mining, Extended Applications of Frequent Patterns.

UNIT V: Classification: Basic Concepts, Decision Tree Induction, Bayesian Classification Methods, Rule-Based Classification, Model Evaluation and Selection, Techniques to Improve Classification Accuracy: Ensemble Methods, Handling Different Kinds of Cases in Classification, Bayesian Belief Networks, Classification by Neural Networks, Support Vector Machines, Pattern-Based Classification, Lazy Learners (or Learning from Your Neighbors), Other Classification Methods.

Text Books

1. Jiawei Han, Micheline Kamber, Jian Pei (2012), Data Mining: Concepts and Techniques, 3rd. edition, Elsevier, United States of America.

Reference Books:

1. Margaret H Dunham (2006), Data Mining Introductory and Advanced Topics, 2nd edition, Pearson Education, New Delhi, India.
2. Amitesh Sinha (2007), Data Warehousing, Thomson Learning, India.

INTERNET OF THINGS

Course Objectives

- To study the fundamentals about IoT
- To study about IoT Access technologies
- To study the design methodology and different IoT hardware platforms.
- To study the basics of IoT Data Analytics and supporting services.
- To study about various IoT case studies and industrial applications.

Learning Outcome

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

- identify the Components that forms part of IoT Architecture.
- determine the most appropriate IoT Devices and Sensors based on Case Studies.
- setup the connections between the Devices and Sensors.
- evaluate the appropriate protocol for communication between IoT.
- analyse the communication protocols for IoT.

UNIT I

Fundamentals of IoT: Evolution of Internet of Things, Enabling Technologies, M2M Communication, IoT World Forum (IoTWF) standardized architecture, Simplified IoT Architecture, Core IoT Functional Stack, Fog, Edge and Cloud in IoT, Functional blocks of an IoT ecosystem, Sensors, Actuators, SmartObjects and Connecting Smart Objects.

UNIT II

IoT Protocols: IoT Access Technologies: Physical and MAC layers, topology and Security of IEEE 802.15.4, 802.11ah and Lora WAN, Network Layer: IP versions, Constrained Nodes and Constrained Networks,6LoWPAN. Application Transport Methods: SCADA, Application Layer Protocols: CoAP and MQTT.

UNIT III

Design and Development: Design Methodology, Embedded computing logic , Microcontroller, System on Chips, IoT system building blocks. IoT Platform overview: Overview of IoT supported Hardware platforms, Raspberry pi, Arduino Board details

UNIT IV

Data Analytics and Supporting Services: Introduction, Structured Versus Unstructured Data, Data in Motion vs Data at Rest, IoT Data Analytics Challenges, Data Acquiring, Organizing in IoT/M2M, Supporting Services: Computing Using a Cloud Platform for IoT/M2M

UNIT V

Applications/Services, Everything as a service and Cloud Service Models. Case Studies/Industrial Applications: IoT applications in home, infrastructures, buildings, security, Industries, Home appliances. Other IoT electronic equipment, Industry 4.0 concepts.

Text Books

1. IoT Fundamentals: Networking Technologies, Protocols and Use Cases for Internet of Things, David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Rob Barton and Jerome Henry, Cisco Press, 2017.
2. Internet of Things: Architecture, Design Principles and Applications, Rajkamal, McGraw Hill Higher Education

Reference Books

1. The Internet of Things – Key applications and Protocols, Olivier Hersent, David Boswarthick, Omar Elloumi and Wiley, 2012 (for Unit2).
2. Recipes to Begin, Expand, and Enhance Your Projects, 2nd Edition, Michael Margolis, Arduino Cookbook and O'Reilly Media, 2011.