<table>
<thead>
<tr>
<th>Code No.</th>
<th>Course Title</th>
<th>Credits*</th>
<th>Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCS8101</td>
<td>Computer Organization</td>
<td>4</td>
<td>100</td>
</tr>
<tr>
<td>DCS8102</td>
<td>Python Programming</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8103</td>
<td>Advanced Database Technology</td>
<td>4</td>
<td>100</td>
</tr>
<tr>
<td>DCS8104</td>
<td>Object Oriented Software Engineering</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8105</td>
<td>Mathematical Foundations of Computer Science</td>
<td>4</td>
<td>100</td>
</tr>
<tr>
<td>DCS8111</td>
<td>Python Programming Lab</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8112</td>
<td>Advanced Database Technology Lab</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td><strong>TOTAL</strong></td>
<td><strong>24</strong></td>
<td><strong>700</strong></td>
</tr>
</tbody>
</table>

**SEMESTER - II**

<table>
<thead>
<tr>
<th>Code No.</th>
<th>Course Title</th>
<th>Credits*</th>
<th>Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCS8201</td>
<td>Computer Networks</td>
<td>4</td>
<td>100</td>
</tr>
<tr>
<td>DCS8202</td>
<td>Advanced Java Programming</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8203</td>
<td>Advanced Data Structures and Algorithms</td>
<td>4</td>
<td>100</td>
</tr>
<tr>
<td>DCS8204</td>
<td>Operating System</td>
<td>4</td>
<td>100</td>
</tr>
<tr>
<td>E1</td>
<td>Elective I</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8211</td>
<td>Advanced Java Programming Lab</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8212</td>
<td>Advanced Data Structures and Algorithms Lab</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td><strong>TOTAL</strong></td>
<td><strong>24</strong></td>
<td><strong>700</strong></td>
</tr>
</tbody>
</table>

**SEMESTER - III**

<table>
<thead>
<tr>
<th>Code No.</th>
<th>Course Title</th>
<th>Credits*</th>
<th>Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCS8301</td>
<td>Web Design</td>
<td>4</td>
<td>100</td>
</tr>
<tr>
<td>DCS8302</td>
<td>Data Warehousing and Data Mining</td>
<td>4</td>
<td>100</td>
</tr>
<tr>
<td>DCS8303</td>
<td>Mobile Application Development</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8304</td>
<td>Object Oriented Analysis and Design</td>
<td>4</td>
<td>100</td>
</tr>
<tr>
<td>E2</td>
<td>Elective II</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8311</td>
<td>Software Development Lab</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8312</td>
<td>Mobile Application Development Lab</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td><strong>TOTAL</strong></td>
<td><strong>24</strong></td>
<td><strong>700</strong></td>
</tr>
</tbody>
</table>

**SEMESTER - IV**

<table>
<thead>
<tr>
<th>Code No.</th>
<th>Course Title</th>
<th>Credits*</th>
<th>Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>E3</td>
<td>Elective III</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>E4</td>
<td>Elective IV</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8411</td>
<td>Project Work</td>
<td>12</td>
<td>400</td>
</tr>
<tr>
<td></td>
<td><strong>TOTAL</strong></td>
<td><strong>18</strong></td>
<td><strong>600</strong></td>
</tr>
</tbody>
</table>

**Total No. of Credits and Marks**

<table>
<thead>
<tr>
<th>Code No.</th>
<th>Course Title</th>
<th>Credits*</th>
<th>Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>TOTAL</strong></td>
<td><strong>90</strong></td>
<td><strong>2700</strong></td>
</tr>
</tbody>
</table>

*Each credit is equivalent to 30 hours of student study comprising of all learning activities.
### ELECTIVE - I

<table>
<thead>
<tr>
<th>Code No.</th>
<th>Course Title</th>
<th>Credits*</th>
<th>Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCS8001</td>
<td>Open Source Systems</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8002</td>
<td>Soft Computing</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8003</td>
<td>Cryptography and Network Security</td>
<td>3</td>
<td>100</td>
</tr>
</tbody>
</table>

### ELECTIVE - II

<table>
<thead>
<tr>
<th>Code No.</th>
<th>Course Title</th>
<th>Credits*</th>
<th>Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCS8004</td>
<td>Cloud Computing Technologies</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8005</td>
<td>Ethical Hacking and Cyber Forensics</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8006</td>
<td>Software Testing and Quality Assurance</td>
<td>3</td>
<td>100</td>
</tr>
</tbody>
</table>

### ELECTIVE - III

<table>
<thead>
<tr>
<th>Code No.</th>
<th>Course Title</th>
<th>Credits*</th>
<th>Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCS8007</td>
<td>Social Network Analysis</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8008</td>
<td>Data Science</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8009</td>
<td>Big Data Analytics</td>
<td>3</td>
<td>100</td>
</tr>
</tbody>
</table>

### ELECTIVE - IV

<table>
<thead>
<tr>
<th>Code No.</th>
<th>Course Title</th>
<th>Credits*</th>
<th>Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCS8010</td>
<td>Artificial Intelligence</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8011</td>
<td>E-Commerce</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>DCS8012</td>
<td>Machine Learning</td>
<td>3</td>
<td>100</td>
</tr>
</tbody>
</table>
COURSE OBJECTIVES
Upon Completion of the course, the students should be able to:

- Master the binary and hexadecimal number systems including computer arithmetic.
- Design and implement digital systems with basic gates and other components using combinational and sequential circuits.
- Be familiar with the Von Neumann architecture and the functional units of the processor and addressing modes, instruction sets.
- Be familiar with the memories and cache subsystem.
- Be familiar with different ways of communicating with I/O devices and standard I/O interfaces.

COURSE OUTCOMES
On completion of the course, the student will be able to:

- To have a thorough understanding of the basic structure and operation of a digital computer.
- Design and realize digital systems with basic gates and other components using combinational and sequential circuits.
- To study the Von Neumann architecture and the functional units of the processor and addressing modes.
- To study the hierarchical memory system including cache memories and virtual memory.
- To study the different ways of communicating with I/O devices and standard I/O Interfaces.

UNIT I  Digital systems, binary numbers, octal, hexadecimal conversions, signed binary numbers, complements, logic gates, Boolean algebra, K-maps, standard forms, NAND-NOR implementation.

UNIT II  Combinational circuits, adder, subtractor, ALU design, decoder, encoder, multiplexers, Sequential circuits: latches, flip-flops, registers, memories, up- down counters.

UNIT III Von-neumann architecture, processor: definition, structure, category, technology, ALU concept, stored programs, fetch execute cycle, instruction formats, clock rate instruction rate, pipeline, current processors, multi core processors.

UNIT IV  Physical memory, addressing, virtual memory, address translation, paging, cache, L1,L2,L3 cache memories, cache mapping, LRU replacement.

UNIT V  Data transfer, Serial and Parallel data transfer, Full duplex- half duplex interaction, Bus interface, Programmed I/O, Polling, Interrupt driven I/O, Hardware interrupt mechanism, Interrupt vectors, Multi-level of interrupts, DMA, buffer chaining, operation chaining.
REFERENCE BOOKS:


DCS8102 PYTHON PROGRAMMING CREDIT: 3

COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

- To develop Python programs with conditionals and loops.
- To use Python data structures - lists, tuples, dictionaries.
- To define Python functions and use function calls.
- To develop Python programs with OOP’s concepts.
- To do input/output with files in Python.

COURSE OUTCOMES

On completion of the course, the student will be able to

- Develop algorithmic solutions to simple computational problems
- Represent compound data using Python lists, tuples, dictionaries.
- Decompose a Python program into functions.
- Structure simple OOPs concepts in Python programs for solving problems.
- Read and write data from/to files in Python Programs.


UNIT IV  OBJECT ORIENTED PROGRAMMING IN PYTHON: Creating a Class, Class methods, Class Inheritance, Encapsulation, Polymorphism, class method vs. static methods, Python object persistence.

UNIT V  FILE HANDLING AND EXCEPTION HANDLING: Files: Introduction - File Path - Opening and Closing Files - Reading and Writing Files - File Position - Exception: Errors and Exceptions, Exception Handling, Multiple Exceptions

REFERENCE BOOKS:

DCS8103  ADVANCED DATABASE TECHNOLOGY  CREDIT: 4

COURSE OBJECTIVES
Upon Completion of the course, the students should be able to:

- Understand the importance of Modeling an Entity Relationship Diagram, Map the Entity Relationship Diagram to Relations and Database Normalization.
- Gain Knowledge on Designing Parallel Databases and Distributed Databases.
- Understand the Basics of XML Databases, Web Databases, Active Databases and Temporal Databases.
- Gain Basic Knowledge on MongoDB NoSQL Database.
- Understand the Basics of Data Warehousing and Data Mining.
COURSE OUTCOMES

On completion of the course, the student will be able to

- Design a Relational Database for an Enterprise.
- Design a Parallel Database and Distributed Database for an Enterprise.
- Apply Knowledge of XML Database, Web Database, Active Database and Temporal Database for maintaining Data of an Enterprise.
- Model a Data Warehouse and Integration of a Data Mining System with a Data Warehouse.

UNIT I  RELATIONAL MODEL: Entity Relationship Model - Relational Data Model - Mapping Entity Relationship Model to Relational Model - Relational Algebra - Structured Query Language - Database Normalization - First Normal Form - Second Normal Form - Third Normal Form - Boyce Codd Normal Form - Fourth Normal Form - Fifth Normal Form.

UNIT II  PARALLEL AND DISTRIBUTED DATABASES: Parallel Databases - I/O Parallelism - Inter-Query and Intra-Query Parallelism - Inter-Operation and Intra-Operation Parallelism - Distributed Database Architecture - Distributed Data Storage - Distributed Transactions - Distributed Query Processing - Distributed Transaction Management - ACID Properties - Concurrency Control.


UNIT IV  NoSQL DATABASES: NoSQL Database vs. SQL Databases - CAP Theorem - Migrating from RDBMS to NOSQL - MongoDB - CRUD Operations - MongoDB Sharding - MongoDB Replication - Web Application Development using MongoDB with PHP and Java.


REFERENCE BOOKS:

4. Jiawei Han, Jian Pei and Hanghang Tong, Data Mining Concepts and Techniques, Fourth Edition, Morgan Kaufmann Publishers, 2022.
COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

- To understand the phases in object oriented software development
- To gain fundamental concepts of requirements engineering and analysis.
- To know about the different approach for object oriented design and its methods
- To learn about how to perform object oriented testing and how to maintain software
- To provide various quality metrics and to ensure risk management.

COURSE OUTCOMES

On completion of the course, the student will be able to

- Able to identify the appropriate process model to develop the object oriented software
- Gain knowledge about requirement elicitation and analyzing techniques
- Able to choose and design suitable UML diagrams and methods
- Able to apply correct testing methods and maintain software systems.
- Able to estimate the object oriented application by applying metric data.


REFERENCE BOOKS:


DCS8105  MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE  CREDIT: 4

COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

- To introduce mathematical logic
- To learn Combinatorics concepts.
- To learn the counting techniques, Algebraic structures.
- Understand Propositions, tautologies and inference rules.
- Able to formulate problems and apply testing of hypothesis.

COURSE OUTCOMES

On completion of the course, the student will be able to

- Understand mathematical logic.
- To develop analytical solutions for logical problems.
- Equipped with counting techniques to Solve combinatorial problems.
- Comprehend the algebraic structure.
- Understand the Inference concepts.


UNIT II  COMBINATORICS: Permutations and Combinations - Mathematical Induction - Pigeonhole principle - Principle of Inclusion and Exclusion - Recurrence relations - Solution by generating functions and characteristics equations.

UNIT IV ALGEBRAIC STRUCTURES: Groups - Cyclic group - Permutation group (Sn and Dn) - Substructures - Homomorphism - Cosets and Lagrange’s Theorem - Normal Subgroups - Rings and Fields (definition and examples).

UNIT V LATTICES: Partial order relation - Posets - Hasse diagram - Lattices - Special Lattices.

REFERENCE BOOKS:


OBJECTIVES:

- To understand the problem solving approaches.
- To learn the basic programming constructs in Python.
- To practice various computing strategies for Python-based solutions to real world problems.
- To use Python data structures - lists, tuples, dictionaries.
- To do input/output with files in Python.

OUTCOMES:

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

- Develop algorithmic solutions to simple computational problems
- Develop and execute simple Python programs.
- Implement programs in Python using conditionals and loops for solving problems.
- Deploy functions to decompose a Python program.
- Process compound data using Python data structures.
- Utilize Python packages in developing software applications.

EXPERIMENTS:

- Python programming using simple statements and expressions (exchange the values of two variables, circulate the values of n variables, distance between two points).
- Scientific problems using Conditionals and Iterative loops. (Number series, Number Patterns, pyramid pattern)
- Implementing real-time/technical applications using Lists, Tuples. (Items present in a library/Components of a car/ Materials required for construction of a building -operations of list & tuples)
- Implementing real-time/technical applications using Sets, Dictionaries. (Language, components of an automobile, Elements of a civil structure, etc.- operations of Sets & Dictionaries)
- Implementing programs using Functions. (Factorial, largest number in a list, area of shape)
- Implementing programs using Strings. (reverse, palindrome, character count, replacing characters)
- Implementing programs using written modules and Python Standard Libraries (pandas, numpy, Matplotlib, scipy)
- Implementing real-time/technical applications using File handling. (copy from one file to another, word count, longest word)
• Implementing real-time/technical applications using Exception handling. (divide by zero error, voter’s age validity, student mark range validation).
OBJECTIVES:
The student should be able:

- Gain Knowledge on Data Definition Language, Data Manipulation Language and Transaction Control Language.
- Understand the Types of Joins, Aggregate Functions, Nested Queries, Creating Views and Creating Stored Procedures.
- Gain Basic Knowledge on Implementing Distributed, XML and Temporal Databases.
- Gain Basic Knowledge on Accessing and Updating a Relational Database using PHP and Java.
- Gain Basic Knowledge on Accessing and Updating a MongoDB NoSQL Database using PHP.

EXPERIMENTS IN THE FOLLOWING TOPICS:

- Data Definition Language - Create - Alter - Drop - Enforcing Primary Key and Foreign Key Constraints - Data Manipulation Language - Insert - Delete - Update - Transaction Control Language - Commit - Rollback - Save Points.
- Cartesian Product - Equi Join - Left Outer Join - Right Outer Join - Full Outer Join.
- Creating Triggers and Stored Procedures.
- Distributed Database Implementation.
- XML Database Implementation.
- Temporal Database Implementation.
- Accessing and Updating a Relational Database using PHP.
- Accessing and Updating a Relational Database using JDBC.
- MongoDB - CRUD Operations
- Accessing and Updating MongoDB using PHP

Exercises 1 to 9 should be implemented using a Relational Database (Oracle / MySQL / PostgreSQL). Exercises 10 and 11 should be implemented using MongoDB NoSQL Database.

OUTCOMES:
Upon completion of this course, the student should be able to:

- Create a Relational Database Enforcing Integrity Constraints and perform Data Manipulation Language Operations.
- Create Views and Stored Procedures.
• Implement Distributed, XML and Temporal Databases.
• Access and Update a Relational Database using PHP and Java.
• Access and Update a MongoDB NoSQL Database using PHP
COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

- Understand data communication techniques.
- To know network Fundamentals.
- To know network layer functions.
- Understand transport layer and its functions.
- Explore network applications in various domains.

COURSE OUTCOMES

On completion of the course, the student will be able to

- Trace the flow of information from one node to another node in the network.
- Identify the component required to build different types of networks.
- Understand the division of network functionalities in network layers.
- Identify solution for each functionality in transport layer.
- Choose the required functionality at each layer for given application.

UNIT I INTRODUCTION: Communication model - Data communications and Networking - Data transmission concepts and terminology - Transmission media - Data Encoding Techniques - Digital Data communication Techniques - Data link Control Protocols.

UNIT II NETWORK FUNDAMENTALS: Protocol architecture - OSI - TCP/IP - LAN Architecture - Topologies - MAC - Ethernet, Fast Ethernet, Token ring, FDDI, Wireless LANS: 802.11/ WiFi / Bluetooth / WiMAX.


REFERENCE BOOKS:

COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

- To get familiar with the concept of packages, interface.
- Able to understand Inheritance and Exception handling in java.
- To learn the concept of Graphical User Interface (GUI).
- Student will be able to develop web application using Java Servlet and Java Server Pages technology.
- To learn database connectivity and network programming.

COURSE OUTCOMES

On completion of the course, the student will be able to

- Identify classes, objects, members of a class and the relationships among them needed for a finding the solution to specific problem.
- Use the Java language for writing well-organized, complex computer programs with both command line and graphical user interfaces.
- Identify and describe common abstract user interface components to design GUI in Java using Applet & AWT along with response to events.
- Apply Servlets and JSP for creating Web based applications using JDBC.
- Design and Develop various application by integrating any of Servlets, JSPs, Swing and Applet using Database.


UNIT V USING RELATIONAL DATABASES: Introduction - JDBC Drivers for RDBM Systems-
REFERENCE BOOKS:

COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

• Understand and apply linear data structures-List, Stack and Queue.
• Apply data structures and algorithms in real time applications.
• Understand the graph algorithms.
• Learn different algorithm analysis techniques.
• Analyze the efficiency of an algorithm.

COURSE OUTCOMES

On completion of the course, the student will be able to

• Implement a program using stack, queue, linked list data structures
• Design and Implement Tree data structures and Sets
• Apply the Graph Data structure and to find shortest path among the several possibilities
• Perform analysis of various algorithms
• Analyze and design algorithms to appreciate the impact of algorithm design in practice.

UNIT I LINEAR DATA STRUCTURES:

UNIT II NON-LINEAR DATA STRUCTURES:

UNIT III GRAPHS:
Representation of graph - Graph Traversals - Depth-first and breadth-first traversal - Applications of graphs - Topological sort - shortest-path algorithms - Dijkstra’s algorithm - Bellman-Ford algorithm - Floyd's Algorithm - minimum spanning tree - Prim's and Kruskal's algorithms.

UNIT IV ALGORITHM DESIGN AND ANALYSIS:

UNIT V ADVANCED ALGORITHM DESIGN AND ANALYSIS:

REFERENCE BOOKS:


COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

- Learn the Operating System basics.
- Study the process management of Operating system.
- Gain knowledge in the storage management.
- To learn I/O systems of Operating system.
- To learn file systems and networking.

COURSE OUTCOMES

On completion of the course, the student will be able to

- Ability to discuss on the basics of OS.
- In depth knowledge in process management.
- In depth knowledge in memory management
- In depth knowledge in I/O Management of various operating systems.
- To explore the case studies with various operating systems.

UNIT I OPERATING SYSTEMS OVERVIEW:

Operating system - Types of Computer Systems - Computer-system operation - I/O structure - Hardware Protection - System components - System calls - System programs - System structure - Process concept - Process scheduling
  - Operations on processes - Cooperating processes - Inter process communication - Communication in client-server systems - Multithreading models - Threading issues.

UNIT II PROCESS MANAGEMENT:


UNIT III STORAGE MANAGEMENT:

Memory Management - Swapping - Contiguous memory allocation - Paging - Segmentation - Segmentation with paging. Virtual Memory: Background
  - Demand paging - Process creation - Page replacement - Allocation of frames - Thrashing.

UNIT IV I/O SYSTEMS:


UNIT V CASE STUDY:

REFERENCE BOOKS:

COURSE OBJECTIVES

- Design and develop GUI applications using Abstract Windowing Toolkit (AWT) and Event Handling.
- Design and develop Web applications
- Designing applications using pre-built frameworks.
- Design interactive web pages using JSP and Servlets.

COURSE OUTCOMES

On completion of the course, the student will be able to

- Create the Internet Programming, using Java Applets.
- Apply exception handling concepts in JAVA programs.
- Create a full set of UI widgets and other components, including windows, menus, buttons, checkboxes, text fields, scrollbars and scrolling lists, using Abstract Windowing Toolkit (AWT).
- Learn to access database through Java programs, using Java Data Base Connectivity (JDBC)
- Create dynamic web pages, using Servlets and JSP.
- Invoke the remote methods in an application using Remote Method Invocation (RMI).

LIST OF EXPERIMENTS:

1. Create UI applications using Java Applets
2. Create a full set of UI widgets and other components, including windows, menus, buttons, checkboxes, text fields, scrollbars and scrolling lists, using Abstract Windowing Toolkit (AWT) & Swings
3. Apply event handling on AWT.
4. Implement servlet concepts.
5. Create Dynamic Web pages using JSP.
6. Invoke the remote methods in an application using Remote Method Invocation (RMI).
7. Create web applications using Java Data Base Connectivity (JDBC)
COURSE OBJECTIVES

- Design ADT concepts.
- To know applications of binary tree.
- Design and develop graph applications.

COURSE OUTCOMES

On completion of the course, the student will be able to

- To develop applications of binary tree.
- To implement merge sort and quick sort concept.
- To develop graph applications using Java.

LIST OF EXPERIMENTS:

1. Create stack and queue operations using Java.
2. To implement Linked List concepts using Java.
3. To implement Binary Search.
4. To implement merge sort and quick sort.
5. To implement AVL Trees using Java.
6. To implement Splay Trees using Java.
7. To implement Red black Trees using Java.
8. To implement Graphs using Java.
COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

- To understand the concepts and architecture of the World Wide Web.
- To understand and practice markup languages
- To understand and practice embedded dynamic scripting on client-side Internet Programming
- To understand and practice web development techniques on client-side.
- The objective is to enable the students to understand the Organizational Behaviour, and Organizational Change and dynamic of groups.

COURSE OUTCOMES

On completion of the course, the student will be able to

- Create a basic website using HTML and Cascading Style Sheets.
- Create websites with complex layouts.
- Add interactivity to websites using simple scripts.
- Design rich client presentation using AJAX.
- Add business logic to websites using PHP and databases.


UNIT V SERVER-SIDE PROGRAMMING WITH PHP: PHP basic syntax-PHP Variables and basic data structures-Using PHP to manage form submissions-File Handling -Cookies and Sessions with PHP-Working with WAMP and PHPMYADMIN-Establishing connectivity with MySQL using PHP.
REFERENCE BOOKS:


COURSE OBJECTIVES
Upon Completion of the course, the students should be able to:

- Expose the students to the concepts of Data warehousing Architecture and Implementation.
- Understand Data mining principles and pre-process techniques.
- Expose the students to the concepts of Association Rule Mining Techniques.
- Study the overview of classification and predictions in data mining.
- Identify the major categorization of clustering techniques in Data mining.

COURSE OUTCOMES
On completion of the course, the student will be able to

- Able to construct the multidimensional data modeling.
- Evolve the importance of data preprocessing.
- Discover the knowledge imbibed in the high dimensional system.
- Experiment Ensemble Methods of various classification algorithms.
- Evaluate various mining techniques on complex data objects.


UNIT II DATA MINING & DATA PREPROCESSING: Introduction to KDD process - Knowledge Discovery from Databases - Need for Data Preprocessing - Data Cleaning - Data Integration and Transformation - Data Reduction - Data Discretization and Concept Hierarchy Generation.

UNIT III ASSOCIATION RULE MINING: Introduction - Data Mining Functionalities - Association Rule Mining - Mining Frequent Itemsets with and without Candidate Generation - Mining Various Kinds of Association Rules - Constraint-Based Association Mining.

UNIT IV CLASSIFICATION & PREDICTION: Classification vs Prediction - Data preparation for Classification and Prediction - Classification by Decision Tree Introduction - Bayesian Classification - Rule Based Classification - Classification by Back propagation - Support Vector Machines - Associative Classification - Lazy Learners - Other Classification Methods - Prediction - Accuracy and Error Measures - Evaluating the Accuracy of a Classifier or Predictor - Ensemble Methods - Model Section.

UNIT V CLUSTERING: Cluster Analysis: - Types of Data in Cluster Analysis - A Categorization of Major Clustering Methods - Partitioning Methods - Hierarchical methods - Density-Based Methods – Grid-Based Methods – Model-Based Clustering Methods – Clustering HighDimensional Data - Constraint-Based Cluster Analysis - Outlier Analysis.

REFERENCE BOOKS:

2. K.P. Soman, Shyam Diwakar and V. Ajay “Insight into Data mining Theory and Practice”, Easter

COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

- To learn the characteristics of mobile applications.
- Understand the intricacies of UI required by mobile applications.
- To study about the design aspects of mobile application.
- To learn development and programming of mobile application.
- To learn the latest tools used to develop mobile app.

COURSE OUTCOMES

On completion of the course, the student will be able to

- To design and implement the user interfaces of mobile applications.
- To design the mobile applications that is aware of the resource constraints of the mobile devices.
- To develop advanced mobile applications that accesses the databases and the web.
- To develop useful mobile applications in the current scenario using Google Android and Eclipse simulator.


UNIT II USER INTERFACE - Generic UI Development - VUIs and Mobile Applications - Text to Speech techniques - Designing the right UI - Multimodal and Multichannel UI - Gesture based UIs - Screen Elements and Layouts - Voice XML - Java API.


UNIT IV APPLICATION DEVELOPMENT - Intents and Services - Storing and Retrieving data - Communication via the Web - Notification and Alarms – Graphics and Multimedia – Telephony - Location based services - Packaging and Deployment - Security and Hacking.


REFERENCE BOOKS:

COURSE OBJECTIVES
Upon Completion of the course, the students should be able to:

- Understand the basics of object oriented analysis and design.
- Learn UML models and tools.
- Learn to analysis the business objects.
- To apply design process to various applications.
- Learn basic testing strategies applied on it.

COURSE OUTCOMES
On completion of the course, the student will be able to

- To know understand object behavior and its methods.
- Familiarize with the topics of object oriented System analysis.
- Implement use case models.
- Apply UML to design various applications.
- Explore test plan and derive test cases.


UNIT III OBJECT ORIENTED ANALYSIS: Identifying Use case - Business object analysis - Use case driven object oriented analysis - Use case model - Documentation - Classification - Identifying object, relationships, attributes, methods - Super-sub class - A part of relationships Identifying attributes and methods - Object responsibility.

UNIT IV OBJECT ORIENTED DESIGN: Design process - Axioms - Colollaries - Designing classes - Class visibility - Refining attributes - Methods and protocols - Object storage and object interoperability - Databases - Object relational systems - Designing interface objects - Macro and Micro level processes - The purpose of a view layer interface.

UNIT V SOFTWARE QUALITY: Quality assurance - Testing strategies - Object orientation testing - Test cases - Test Plan - Debugging principles - Usability - Satisfaction - Usability testing - Satisfaction testing.

REFERENCE BOOKS:


COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

- Understand project planning
- Able to Analysis and Design Application
- Able to Implement and Test Software.

COURSE OUTCOMES

On completion of the course, the student will be able to

- Able to Draw DFD and UML Diagrams
- Able to Write programs for Design
- Generate and Execute Test cases.

EXPERIMENTS IN THE FOLLOWING TOPICS

1. Apply the following to typical application problems:
2. Project Planning
3. Software Requirement Analysis
4. Software Estimation
5. Software Design
6. Data Modelling & Implementation
7. Software Testing
8. Software Debugging

A possible set of applications may be the following:

a. Library System
b. Student Marks Analyzing System
c. Text Editor.
d. Create a dictionary.
e. Telephone dictionary.
f. Simulator Software for Parallel Processing Operation.
g. Inventory System.
COURSE OBJECTIVES
Upon completion of the course, the students should be able to:

- To know about various platforms and tools available for developing mobile applications.
- To realize the differences between developing conventional applications and mobile applications.
- To learn programming skills in J2ME and Android SDK.
- To study about micro browser based applications to access the Internet using Sun Java Toolkit.

COURSE OUTCOMES
On completion of the course, the student will be able to

- Develop useful mobile applications for the current scenario in mobile computing and pervasive computing.

EXPERIMENTS IN THE FOLLOWING TOPICS:
2. Form design for mobile applications.
3. Applications using controls.
4. Graphical and Multimedia applications.
5. Data retrieval applications.

Gaming application:
(Perform the experiments from 2 to 7 in J2ME and Android SDK framework)

Micro browser based applications using WAP, WML and WML scripts
(Perform experiments in 8 using Sun Java Wireless toolkit).
ELECTIVES

DCS8001 OPEN SOURCE SYSTEMS CREDIT: 3

COURSE OBJECTIVES
Upon Completion of the course, the students should be able to:

- To understand open-source licenses and learn the implications for users, developers and the software community.
- To become familiar with and become adapt using the tools of open source development.
- To learn GNU and practice open-source programming techniques.

COURSE OUTCOMES
On completion of the course, the student will be able to

- Understands the importance of open source and how it can be used in efficient manure.
- Gain knowledge on Linux.
- Configure Hardware using open source tools
- Get experience with GNU Libraries.
- Understand various system software tools.


UNIT II Linux OS Installation and Hardware Configuration - Configure disk partitions & file systems and install a GNU/Linux distribution - Basic shell commands - Logging in, Listing files, editing files, copying/moving files, viewing file contents, changing file modes and permissions, process management User and group management - File ownerships and permissions.

UNIT III Configuring additional hardware - Sound cards - Displays & display cards - Network cards - Modems - USB drives - CD writers - The OS boot up process - Performing everyday tasks using GNU / Linux - Accessing the Internet - Playing music - Editing documents and spreadsheets - Sending and receiving email - Copy files from disks and over the network - Playing games - Writing CDs - X Window system configuration and utilities - Configure X windows - Detect display devices - Installing software - From source code as well as using binary packages - Setting up email servers Using postfix -(SMTP services) - Courier (IMAP & POP3 services) - Squirrel mail (web mail services).

UNIT IV GNU compiler tools - The C compiler (gcc) and the C++ compiler (g++) - Linking against object archives (.a libraries) and dynamic shared object libraries (.so libraries) - Generating statically linked binaries and libraries - Generating dynamically linked libraries - Using the GNU debugging tools - Gdb to debug programs - Graphical debuggers like ddd - Memory debugging/profiling libraries mpatrol and valgrind.

UNIT V Application Programming - Basics of the X Windows server architecture - Qt programming - Gtk+ programming - Execution Environment - Programming GUI applications with
localisation support, Open Source Equivalent of existing commercial software.
REFERENCE BOOKS:


COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

- 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.

- Provide the mathematical background for carrying out the optimization associated with neural network learning.

- 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.

COURSE OUTCOMES

On completion of the course, the student will be able to

- Describe human intelligence and AI

- 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.

UNIT I

Introduction to soft computing - brief description of separate theories, Introduction to biological and artificial neural network, Classification algorithms - Decision Trees, Bayesian classifier - Neural Networks and Probabilistic Reasoning.

UNIT II


UNIT III

Fundamentals of fuzzy sets and fuzzy logic theory, fuzzy inference principle, Examples of use of fuzzy logic in control of real-world systems.

UNIT IV


UNIT V

REFERENCE BOOKS:


COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

- To understand basics of Cryptography and Network Security.
- To be able to secure a message over in secure channel by various means.
- To learn about how to maintain the Confidentiality, Integrity and Availability of a data.
- To understand various protocols for network security to protect against the threats in the networks.
- To Understand Intruders and detection Techniques.

COURSE OUTCOMES

On completion of the course, the student will be able to

- Provide security of the data over the network.
- Do research in the emerging areas of cryptography and network security.
- Implement various networking protocols.
- Protect any network from the threats in the world.
- To protect from intruders and Virus Threats.

UNIT I

UNIT II

UNIT III

UNIT IV

UNIT V

REFERENCE BOOKS:


COURSE OBJECTIVES
Upon Completion of the course, the students should be able to:

- To understand the basic concepts of Distributed systems.
- To learn about the current trend and basics of Cloud computing.
- To be familiar with various Cloud concepts.
- To expose with the Server, Network and storage virtualization.
- To be aware of Microservices and DevOps.

COURSE OUTCOMES
On completion of the course, the student will be able to

- Upon completion of the course, the students will be able to.
- Use Distributed systems in Cloud Environment.
- Articulate the main concepts, key technologies, strengths and limitations of Cloud computing.
- Identify the Architecture, Infrastructure and delivery models of Cloud computing.
- Install, choose and use the appropriate current technology for the implementation of Cloud.
- Adopt Microservices and DevOps in Cloud environment.

UNIT I DISTRIBUTED SYSTEMS
Introduction to Distributed Systems - Characterization of Distributed Systems - Distributed Architectural Models

UNIT II INTRODUCTION TO CLOUD COMPUTING

UNIT III CLOUD INFRASTRUCTURE

UNIT IV CLOUD ENABLING TECHNOLOGIES

UNIT V MICROSERVICES
Defining Micro services - Emergence of Micro service Architecture - Design patterns of Micro services - The Mini web service architecture - Micro service dependency tree - Challenges with Micro services - SOA
vs Micro service - Micro service and API - Deploying and maintaining Micro services.
REFERENCE BOOKS:


COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

- To understand the hacking techniques of computer forensics.
- To learn about data recovery methods.
- To identity the threats in computer forensics.

COURSE OUTCOMES

On completion of the course, the student will be able to

- To apply the principles of computer forensics for security.
- To implement the data recovery methods.
- To manage threats and the tactics.

UNIT I  Foundation for Ethical Hacking-Ethical Hacking in Motion-Hacking Network Hosts-Hacking Operating Systems-Hacking Applications.


UNIT III  Data Recovery - Evidence Collection and Data Seizure - Duplication and Preservation of Digital Evidence - Computer Image Verification and Authentication.

UNIT IV  Discover of Electronic Evidence - Identification of Data - Reconstructing Past Events - Networks.

UNIT V  Fighting against Macro Threats - Information Warfare Arsenal - Tactics of Military - Tactics of Terrorist and Rogues - Tactics of Private Companies.

REFERENCE BOOKS:

COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

- To know the behavior of the testing techniques and to design test cases to detect the errors in the software.
- To get insight into software testing methodologies
- To understand standard emerging areas in testing
- To learn about the software quality models.
- To understand the models and metrics of software quality and reliability.

COURSE OUTCOMES

On completion of the course, the student will be able to

- choose the software testing techniques to cater to the need of the project.
- identify the components of software quality assurance systems.
- apply various software testing strategies.
- design and develop software quality models.
- make use of statistical methods in software quality.


UNIT V Role of Statistical Methods in Software Quality - Transforming Requirements into Test Cases - Deming’s Quality Principles - Continuous Improvement through Plan Do Check Act (PDCA).

REFERENCE BOOKS:


COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

- To gain knowledge about social networks, its structure and their data sources.
- To study about the knowledge representation technologies for social network analysis.
- To analyse the data left behind in social networks.
- To gain knowledge about the community maintained social media resources.
- To learn about the visualization of social networks.

COURSE OUTCOMES

On completion of the course, the student will be able to

- Explain the basic principles behind network analysis algorithms.
- Model and represent knowledge for social semantic Web.
- Use extraction and mining tools for analyzing Social networks.
- Discuss about community maintained social media resources.
- Develop personalized visualization for Social networks.

UNIT I
The development of Semantic Web - Emergence of the Social Web - The Development of Social Network Analysis - Basic Graph Theoretical Concepts of Social Network Analysis - Electronic Sources for Network Analysis - Electronic Discussion Networks, Blogs and Online Communities, Web-based Networks.

UNIT II
Ontology-based knowledge Representation - Ontology languages for the Semantic Web: RDF and OWL- Modeling Social Network Data – Network Data Representation, Ontological Representation of Social Individuals and Relationships -Aggregating and Reasoning with Social Network Data.

UNIT III
Detecting Communities in Social Network - Evaluating Communities -Methods for Community Detection - Applications of Community Mining Algorithms - Tools for detecting communities - Application: Mining Facebook - Exploring Facebook’s social Graph API - Analyzing social graph connections.

UNIT IV
Community Maintained Resources - Supporting technologies for community maintained resources- User motivations-Location based social interaction - location technology-mobile location sharing - Social Information Sharing and social filtering - Automated recommendersystem.

UNIT V

REFERENCE BOOKS:


COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

- To know the fundamental concepts of data science and analytics.
- To learn fundamental data analysis using R.
- To understand various data modeling techniques.
- To learn the basic and advanced features of open source big data tools and frameworks.
- To study various analytics on stream data.

COURSE OUTCOMES

On completion of the course, the student will be able to:

- On completion of the course, the students will be able to:
  - Convert real world problems to hypothesis and perform statistical testing.
  - Perform data analysis using R.
  - Design efficient modeling of very large data and work with big data platforms.
  - Implement suitable data analysis for stream data.
  - Write efficient Map Reduce programs for small problem solving methods.

UNIT I


UNIT II


UNIT III


UNIT IV


UNIT V

REFERENCE BOOKS:

COURSE OBJECTIVES
Upon Completion of the course, the students should be able to:

- To know the fundamental concepts of big data and analytics.
- To explore tools and practices for working with big data.
- To learn about stream computing.
- To know about the research that requires the integration of large amounts of data.

COURSE OUTCOMES
On completion of the course, the student will be able to:

- Work with big data tools and its analysis techniques.
- Analyze data by utilizing clustering and classification algorithms.
- Learn and apply different mining algorithms and recommendation systems for large volumes of data.
- Perform analytics on data streams.
- Learn NoSQL databases and management.

UNIT I Introduction to Big Data:
Types of Digital Data: Classification of Digital Data, Introduction to Big Data: Characteristics of data-Evolution of Big data-Challenges of Big data-Other Characteristics of Data Which are not Definitional Traits of Big Data-Why Big Data?-Are we Just an Information Consumer or Do we also produce Information?-Traditional Business Intelligence (BI) versus Big Data – A Typical Data Warehouse Environment – A Typical Hadoop Environment – What is New Today? – What is changing in the Realms of Big Data?

UNIT II Analytics Basics:

UNIT III Big Data Technologies:

UNIT IV Introduction to MAPREDUCE Programming:
Function (UDF).

UNIT V Analytical Algorithms:

REFERENCE BOOKS:


COURSE OBJECTIVES

Upon Completion of the course, the students should be able to:

- To know the underlying structure behind intelligence mathematically.
- To know the logical implications in computational intelligence.
- To know the automated learning techniques.
- To study the techniques of Knowledge Representation.
- To explore Artificial Intelligence techniques in real-time scenarios.

COURSE OUTCOMES

On completion of the course, the student will be able to

- Understand the search techniques.
- Apply the search techniques to real-time problems.
- Apply the reasoning techniques to real-world problems.
- Understand the representation of knowledge.
- Understand the learning techniques.
- Apply AI techniques in developing real-world applications.


UNIT III  REASONING WITH LOWER ORDER LOGICS: Logical Agent - Proposition Logic - Syntax and Semantics - Theorem Proving - Model Checking - Inference in First Order Logic.

UNIT IV  ARTIFICIAL INTELLIGENCE PLANNING: Classical Planning - Partial Order Planning - Graph Plan and SAT Plan - Hierarchical Planning - Planning and Acting in Nondeterministic Domains – Multiagent Planning.

UNIT V  LEARNING TECHNIQUES: Logical Formulation of Learning - Knowledge in Learning - Explanation-Based Learning - Learning using Relevance Information - Inductive Logic Programming - Statistical Learning - Learning with Complete Data - Learning with Hidden Data – Applications.

REFERENCE BOOKS:


COURSE OBJECTIVES
Upon Completion of the course, the students should be able to:

- Able to Understand
- the TCP/IP networks
- Electronic payment systems
- E-Security principles.

COURSE OUTCOMES
On completion of the course, the student will be able to

- XML and Web Databases
- Implement Electronic payment systems
- Implement E-Security systems.


REFERENCE BOOKS:
COURSE OBJECTIVES
Upon Completion of the course, the students should be able to:

- To understand the concepts of Machine Learning.
- To appreciate supervised learning and their applications.
- To appreciate the concepts and algorithms of unsupervised learning.
- To understand the theoretical and practical aspects of Probabilistic Graphical Models.
- To appreciate the concepts and algorithms of advanced learning.

COURSE OUTCOMES
On completion of the course, the student will be able to

- Design a learning model appropriate to the application.
- Design a Neural Network for an application of your choice.
- Implement Probabilistic Discriminative and Generative algorithms for an application of your choice and analyze the results.
- Use a tool to implement typical Clustering algorithms for different types of applications.
- Design and implement an HMM for a Sequence Model type of application.
- Identify applications suitable for different types of Machine Learning with suitable justification.


REFERENCE BOOKS:
