THIRD SEMESTER
(Detailed Syllabus)

'CYBER FORENSICS
&
INFORMATION SECURITY'

DATA STRUCTURE USING ‘C’

Course Code:	433003
Course Title	Data Structure Using ‘C’
No. of Credits	6 (TH:4,T:0,P:4)

COURSE OUTCOMES : At the end of the course, the student will be able to:

1. Understand the fundamental concepts of data structures and their importance in programming.
2. Gain proficiency in implementing and using various data structures such as arrays, linked lists, stacks, queues, trees, and graphs using the C programming language.
3. Develop problem-solving skills by applying appropriate data structures and algorithms to solve real-world programming challenges.
4. Acquire knowledge of different sorting and searching techniques and their implementation using C.
5. Learn the concept of recursion and its applications in solving problems related to data structures.
6. Develop proficiency in using dynamic memory allocation and pointers to efficiently manage and manipulate data structures.

COURSE CONTENTS

1. Fundamental Notations

1.1 Problem solving concept, top down and bottom up design, structured programming

1.2 Concept of data types, variables and constants

1.3 Concept of pointer variables and constants

1.4 Introduction to data Structure (Linear, Non Linear, Primitive, Non Primitive)

1.5 Concepts of Data Structure(Array, Linked List, Stack, Queue, Trees, graphs)

2. Arrays

2.1 Concept of Arrays

2.2 Single dimensional array

2.3 Two dimensional array: Representation of Two dimensional Array (Base Address, LB, UB)

2.4 Operations on arrays with Algorithms (searching, traversing, inserting, deleting)

3. Linked

3.1 Introduction to linked list and double linked list

3.2 Representation of linked lists in Memory, Comparison between Linked List and Array

3.3 Traversing a linked list

3.4 Searching linked list

3.5 Insertion and deletion into linked list (At first Node, Specified Position, Last node

3.6 Application of linked lists

4. Stacks, Queues and Recursion

4.1 Introduction to stacks

4.2 Representation of stacks with array and Linked List

4.3 Implementation of stacks

4.4 Application of stacks

4.5 Recursion: Concept and Comparison between recursion and Iteration

4.6 Introduction to queues

4.7 Implementation of queues

5. Trees: Introduction

5.1 Concept of binary trees (complete, extended binary tree)

5.2 Concept of representation of Binary Tree

5.3 Concept of balanced Binary Tree

5.4 Traversing binary trees (pre order, post order & in order)

5.5 Searching, inserting and deleting in binary search trees

6. Sorting and Searching

6.1 Introduction

6.2 Search algorithm (Linear and Binary)

6.3 Concept of sorting

6.4 Sorting algorithms (Overview only)

PRACTICAL OUTCOMES : At the end of the course, the student will be able to:

1. Gain hands - on experience in implementing various data structures using the C programming language.
2. Develop proficiency in using C language constructs and functions to create and manipulate data structures such as arrays, linked lists, stacks, queues, trees, and graphs.
3. Acquire practical knowledge of implementing algorithms for sorting, searching, and other operations on different data structures.
4. Enhance problem-solving skills by applying appropriate data structures and algorithms to solve programming challenges in a laboratory setting.
5. Develop the ability to analyze and evaluate the efficiency and performance of different data structures and algorithms through practical experiments and benchmarking.
6. Improve teamwork and collaboration skills by working on group projects and assignments that involve implementing and testing data structures using C programming.

List of Practicals :-

Write Programs in 'C' language to implement:-

1. The addition of two matrices using functions.
2. Inserting and deleting elements in array.
3. Push and pop operation in stack.
4. Conversion from in-fix notation.
5. The factorial of a given number using recursion.
6. Insertion and Deletion of elements in queue using pointers.
7. Insertion and Deletion of elements in circular queue using pointers.
8. Insertion and Deletion of elements in linked list.
9. Insertion and Deletion of elements in doubly linked list.
10. The linear search procedures to search an element in given list.
11. The binary search procedures to search an element in a given list.

References /Suggested Learning Resources:

1. Digital principles & Applications Albert Paul Malvino & Donald P. Leach McGraw Hill Education; Eighth edition ISBN: 978-9339203405

2. Digital Electronics Roger L. Tokheim Macmillan McGraw-Hill Education (ISE Editions); International 2 Revised edition ISBN: 978-0071167963

3. Digital Electronics – an introduction to theory and practice William H. Gothmann Prentice Hall India Learning Private Limited; 2 edition ISBN: 978-8120303485

4. Fundamentals of Logic Design Charles H. Roth Jr. Jaicoh Publishing House; First edition ISBN: 978-8172247744

5. Digital Electronics R. Anand Khanna Publications, New Delhi (Edition 2018) ISBN: 978-93-82609445

*****

DATA COMMUNICATION & NETWORKING

Course Code:	493001
Course Title	Data Communication & Networking
No. of Credits	6 (TH:4,T:0,P:4)

COURSE OUTCOMES: At the end of the course, the student will be able to:

1. Understand the fundamental concepts, principles, and protocols of data communication and networking.
2. Analyse and evaluate different network architectures and technologies for efficient data transmission and connectivity.
3. Demonstrate proficiency in configuring and trouble-hooting network devices, protocols, and services.
4. Design and implement secure and reliable network infrastructures to meet organizational requirements.
5. Apply knowledge of network management and administration practices for efficient network operation.
6. Collaborate effectively in multidisciplinary teams, demonstrating strong communication and problem-solving skills in network-related scenarios.

COURSE CONTENTS

1. Introduction to Data Communication & Networking

- Basic concept of communication system.
- Introduction to network components, devices, and their functionalities.

2. Network Architecture and Protocols:

- TCP/IP protocol.
- Local area networks (LANs) and wide area networks (WANs).

3. Network Infrastructure and Devices:

- Network devices, such as routers, switches & firewalls.
- Network media & cabling types for data transmission.

4. Network Services and Applications:

- Introduction to network services.
- Network application layer protocols.

5. Network Security and Management:

- Network security principles, authentication, encryption, and firewall technologies.
- Network management tools and techniques.

6. Network Design and Implementation:

- Designing network infrastructures.

Practical Outcomes: The student will be able to understand:-

1. Concepts of networking and its components
2. Concepts of OSI and TCP/IP model
3. Detailed concepts and practices of Layer-2 and Layer-3 networking systems
4. Case study of a practical network setup of a model office.

List of Practicals:

1. Study different types of network cables and Connectors. Then practically make the Cross-over Cable and Straight cable using RJ45 connectors, Crimping tools and LAN Tester.
2. Learn IP Addressing scheme, Subnet masking and Subnetting. Then implement static IP setting, subnet mask, Gateway and DNS configuration on Linux/Windows machine in a network system.
3. Install and run the following applications in a network system and get knowledge:

   (i) FTP (ii) SSH (iii) PUTTY (iv) Remote Desktop
4. a) Locate the MAC address of computer.

   b) Use step by step method for File sharing , Printer sharing and Internet sharing in a network system

   c) Familiarization with loop back testing

   d) Familiarization with the concept of socket and write a socket program
5. Basic TCP/IP utilities and commands. (e.g.: ping, ipconfig, tracert, arp, tcpdump, whois, host, netstat, nslookup)
6. Configure a router (Ethernet & Serial Interface) using router commands.

Reference Books:

1. "Computer Networking: A Top-Down Approach" by James F. Kurose and Keith W. Ross
2. "Data Communications and Networking" by Behrouz A. Forouzan
3. "TCP/IP Illustrated, Volume 1: The Protocols" by W. Richard Stevens
4. "Network Warrior: Everything You Need to Know That Wasn't on the CCNA Exam" by Gary A. Donahue
5. "CCNA Routing and Switching Complete Study Guide: Exam 100-105, Exam 200-105, Exam 200-125" by Todd Lammle
6. "Network Security Essentials: Applications and Standards" by William Stallings.

*****

Introduction to Cyber Security

Course Code:	493002
Course Title	Introduction to Cyber Security
No. of Credits	6 (TH:4,T:0,P:4)

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

1. Understand, appreciate, employ, design and implement appropriate security technologies and policies to protect computers and digital information.

2. Identify & Evaluate Information Security threats and vulnerabilities in Information Systems and apply security measures to real time scenarios

3. Identify common trade-offs and compromises that are made in the design and development process of Information Systems

3. Analyze and resolve security issues in networks and computer systems to secure an IT infrastructure.

COURSE CONTENTS

Unit-I : Cyber Security Concepts

Essential Terminologies: CIA, Risks, Breaches, Threats, Vulnerability, Attacks, Exploits, Information Gathering (Social Engineering, Foot Printing and Scanning), Enumeration and System Hacking Open Source Tools.

Unit-2 : Infrastructure & Network Security (Overview Only)

Introduction to System Security, Server OS Security, Physical Security, Network packet Sniffing, DOS/DDOS attacks, Man-in-the middle attack (Spoofing), Vulnerability Assessment, Penetration Testing, Asset Management, Intrusion detection and Prevention, Security Information Management, Open Source Tools: DOS Attacks, DDOS attacks, Wireshark, Cain &Abel, Linux/Windows Firewall.

Unit-3: Cyber Security Vulnerabilities and Safe Guards

Internet Security, Concepts of Cloud Computing, Social Network sites security, Cyber Security Vulnerabilities-Overview.

Unit-4: Malware

Definition of Malware, Types of Malware, OS Hardening, Malware Analysis, Open Source Tools: Antivirus Protection, Anti Spywares, System tuning tools, Anti-Phishing.

Practical Outcomes: The student will be able to understand:-

1. The Concepts of gathering information from a network using specialized tools.
2. The various Concepts of Cryptography.
3. The idea of Network Security using Windows firewall etc.
4. The Detailed concepts and practices of Network attacking tools.

List of Practical:

1. Implementation to gather information from any PC's connected to the LAN using whois, port scanners, network scanning, Angry IP scanners etc.
2. Implementation of Symmetric and Asymmetric cryptography.
3. Implementation of LSB Steganography
4. Implementation of MITM- attack using wireshark / network sniffers, Netcat, Nmap or Popular Open source tools.
5. Implementation of Windows security using firewall and other tools.
6. Implementation to identify web vulnerabilities using popular open source tools.

Text Books:

1. Nina Godbole, “Information System Security”, Wiley
2. Gupta Sarika, “Information and Cyber Security”, Khanna Publishing House, Delhi.
3. William Stallings, “Cryptography and Network Security”, Pearson Education/PHI
4. V.K. Jain, “Cryptography and Network Security”, Khanna Publishing House.
5. Atul Kahate, “Cryptography and Network Security”, McGraw Hill.
6. V.K. Pachghare, “Cryptography and Information Security”, PHI Learning
7. Bothra Harsh, “Hacking”, Khanna Publishing House, Delhi.

*****

WEB TECHNOLOGIES

Course Code:	433005
Course Title	Web Technologies
No. of Credits	6 (TH:4,T:0,P:4)

COURSE OUTCOMES: At the end of the course, the student will be able to:

1. Understand the fundamental concepts and principles of web technologies, including client-server architecture, HTTP, and web standards.
2. Design and develop interactive and user-friendly web pages using HTML, CSS, and JavaScript.
3. Implement responsive web design techniques to create websites that adapt to different screen sizes and devices.
4. Utilize server-side scripting languages (such as PHP or ASP.NET) to create dynamic web pages and handle form submissions.
5. Integrate databases into web applications and perform basic database operations using SQL.
6. Apply web security principles and best practices to protect against common vulnerabilities and threats.

COURSE CONTENTS

UNIT - 1: Introduction to WWW

1.1 Protocols and programs

  1.1.1 Secure connections,

  1.1.2 Application and development tools

1.2 The web browser

1.3 What is server

1.4 Setting up UNIX and LINUX web servers

1.5 Logging users

1.6.Dynamic IP Web Design

  1.6.1 Web site design principles

  1.6.2 Planning the site and navigation

UNIT - 2 : Web Systems Architecture (Overview Only)

2.1 Architecture of Web based systems

2.2 Building blocks of fast and scalable data access Concepts

2.3 Web Application architecture (WAA)

UNIT - 3: Javascript

3.1 Client-side scripting

3.2 Basic Concept of JavaScript

3.3 Variables, Functions, conditions, Loops and repetition

UNIT 4: Advance Scripting

4.1.JavaScript and objects

  4.1.1 JavaScript own objects

  4.1.2 DOM and web browser environments, forms and validations

4.2 DHTML

4.3 Introduction to Ajax

4.4 Introduction to XML

4.5 Introduction to Web Services

UNIT - 5 : PHP

5.1 Server- side scripting

- 5.1.1 Arrays
- 5.1.2 Function and forms
- 5.1.3 Advance php

5.2 Databases

- 5.2.1 Basic command with php examples
- 5.2.2 Connection to server, creating database
- 5.2.3 Selecting and listing database
- 5.2.4 Listing table- names creating a table
- 5.2.5 Inserting data

PRACTICAL OUTCOMES: By the end of the web technology lab, the student will be able to:

1. Demonstrate proficiency in developing server-client programs using various programming languages & protocols.
2. Create dynamic web pages using HTML and enhance interactivity using JavaScript.
3. Apply advanced techniques of CSS to design visually appealing and user-friendly web applications.
4. Develop PHP programs by applying the basics of PHP programming language.
5. Build web applications using PHP, including form handling, session management, & integration with databases.
6. Connect PHP with MySQL database and develop secure tiered web applications.
7. Design and develop a fully functional web service application by integrating all the technologies learned throughout the course.

List of Practicals:

1. Coding Server Client Programs
    - Practice developing server-client programs using various programming languages and protocols.
2. Developing Web Applications using HTML and JavaScript
    - Learn and practice creating dynamic web pages using HTML for structure and JavaScript for interactivity.
3. Developing Advanced Web Application Programs using CSS

- Explore advanced techniques of Cascading Style Sheets (CSS) to design and enhance web applications.

4. Practicing PHP: Basics
   - Gain hands-on experience in PHP programming language, covering its syntax, variables, data types, and control structures.
5. Practicing PHP: Web Application Development
   - Develop web applications using PHP, focusing on form handling, session management, and integrating databases.
6. Practicing PHP: MySQL - Tiered Applications
   - Learn how to connect PHP with MySQL database and build tiered web applications with secure data handling.
7. Developing a Fully Functional Web Service Application
   - Apply all the technologies learned throughout the course to develop a comprehensive and functional web service application.

References:

1. "Web Technologies--A Computer Science Perspective", Jeffrey C. Jackson,
2. "Internet & World Wide Web How To Program", Deitel, Deitel, Goldberg, Pearson Education
3. "Web programming- Building Internet Application", Chris Bales
4. "Web Applications: Concepts and Real World Design", Knuckles

*****

DISCRETE MATHEMATICS

Course Code:	433001
Course Title	Discrete Mathematics
No. of Credits	4 (TH:4,T:0,P:0)

Course Outcomes: At the end of the course, the student will be able to:

1. Understand fundamental concepts of discrete mathematics.
2. Apply discrete mathematics principles to solve practical problems in Artificial Intelligence and Machine Learning.
3. Analyse basic algorithms and data structures used in relevant field.
4. Identify graph theory applications in Artificial Intelligence and Machine Learning analysis.
5. Demonstrate basic knowledge of set theory, logic, and combinatorics relevant to Artificial Intelligence and Machine Learning.

COURSE CONTENTS

Unit 1: Sets and Logic

- Introduction to sets and set operations
- Logic and truth tables
- Propositional logic and logical operators

Unit 2: Combinatorics

- Counting principles: Permutations and combinations
- Basic probability concepts and applications

Unit 3: Graph Theory

- Introduction to graphs and their components
- Graph representations and basic graph algorithms
- Applications of graph theory in AI&ML (over view only)

Unit 4: Number Theory

- Divisibility and prime numbers
- Modular arithmetic and its applications in cryptography (over view only)

Unit 5: Relations and Functions

- Introduction to relations and properties
- Functions and their properties
- Applications of relations and functions in AI&ML (over view only)

Suggested Books:

1. "Discrete Mathematics and its Applications" by Kenneth H. Rosen
2. "Discrete Mathematics: Elementary and Beyond" by L. Lovász, J. Pelikán, and K. Vesztergombi
3. "A Path to Combinatorics for Undergraduates" by Titu Andreescu and Zuming Feng
4. "Introduction to Graph Theory" by Richard J. Trudeau
5. "Introduction to the Theory of Computation" by Michael Sipser

*****

COMPUTER SYSTEM ORGANIZATION

Course Code:	433002
Course Title	Computer System Organization
No. of Credits	6 (TH:4,T:0,P:4)

COURSE OUTCOMES: After completion of this course, student will be able to:

1. Have a good understanding of functioning of computer system as such and its various sub- components. Student will be able to understand computing requirement for a specific purpose, analyze performance bottlenecks of the computing device and choose appropriate computing device for a given use case.
2. Understand a computer system that has hardware and software components, which controls and makes them useful.
3. Understand the fixed and floating point number representation in computer.
4. Understand how arithmetic operation will be performed in computer system.
5. Acquire knowledge on Cache and virtual memory.
6. Understand Interrupt and DMA access.
7. Acquire knowledge on RISC and CISC architecture.
8. Understand how pipelining and parallel processing improves the performance of computer system.

COURSE CONTENTS

UNIT - 1: Basics of Computer system

1.1 Structure of Computers: Computer Functional units, Von-Neumann architecture and its bottleneck, Bus structures, Basic Operational Concepts, Data representation (Fixed and Floating point), Error detecting codes.

1.2 Register Transfer and Micro Operations: Register transfer, Bus and memory transfers, Arithmetic micro-operations, Logic micro-operations, Shift micro-operations, and Arithmetic logic shift unit.

UNIT - 2 : Introduction to Microprocessor Architecture

2.1 Difference between Computer Architecture and Organization, Instruction Set Architecture design principles from programmer's perspective. One example of microprocessor (8085).

UNIT - 3 : Assembly Language Programming

3.1 Simple programs, Assembly language programs involving logical, branch and call instructions, sorting, evaluation of arithmetic expressions, string manipulation, assembler directives, procedures and macros.

UNIT - 4 : Memory and Digital Interfacing

4.1 Addressing and address decoding, interfacing RAM, ROM, EPROM, programmable peripheral interface, various modes of operation and interfacing to processor, interfacing keyboard, displays, etc.

4.2 Memory and I/O: Memory Hierarchy model and comparison on cost, speed and size; Cache memory, Locality of Reference, Concept of virtual memory technique, Overview of different methods of IO access mechanism; Programmed IO or Status check IO, Interrupt Classification, DMA data transfer, IO processor; Overview of DMA transfer, Programmable Interrupt Controller, Programmable DMA Controller.

UNIT - 5 : Control Unit and Parallel Processing

5.1 Micro Programmed Control: Control memory, Address sequencing, and design of control unit.

5.2 Overview of Basic Computer Arithmetic operations.

Practical Outcomes : At the end of the course, the student will be able to:

1. To understand the 8085 trainer & execute simple program.
2. Develop Different program in Assembly Language Programming and execute them and remove faults.

List of Practicals:

1. 1.Engage in hands-on exercises involving binary addition and subtraction to gain familiarity with fundamental computer arithmetic operations.
2. Write and execute simple assembly language programs to add numbers, providing insights into low-level computer programming.
3. Explore memory sizes by comparing capacities such as kilobytes, megabytes, and gigabytes, enhancing your understanding of data storage measurements.
4. Disassemble a computer system with guidance to identify memory components like RAM, ROM & cache, gaining practical insight.
5. Participate in a simulation activity to understand how computers handle interrupts & respond to external events.
6. Create and run assembly programs that interact with users, reading input and displaying output, providing hands-on data handling experience.
7. Practice converting numbers between decimal and binary, developing a strong grasp of different number systems.
8. Construct basic logic gates using electronic components and breadboards, gaining insight into the foundations of digital circuits.
9. Design a paper-based communication protocol to comprehend data buses & their role in information exchange.
10. Study microprocessor architecture through block diagrams to identify and understand its core components.

Text Books:

1. Computer System Architecture, M. Moris Mano, Pearson/PHI, India.
2. Computer Organization, Carl Hamacher, Zvonks Vranesic, SafeaZaky, McGraw-Hill
3. Computer Organization and Architecture, Stallings, Pearson

Reference Books:

1. Microprocessors Interface, Douglas V.Hall, Tata McGraw-Hill.
2. Advanced Microprocessors and Peripherals- Architecture, Programming and interfacing, A.K.Ray, K. M. Bhurchandi, Tata McGraw-Hill, New Delhi, India.
3. Computer Organization and Design: A Hardwar/ Software Interface (MIPS Edition) by Patterson and Hennessy

*****

SUMMER INTERNSHIP-I

Course Code:	AS301
Course Title	Summer Internship - I
No. of Credits	2 (TH:0,T:0,P:0)

Summer Internship provides an invaluable opportunity for students pursuing their Diploma in Engineering to gain real-world experience and exposure to various industrial production units and commercial activities related to their field of study. This program aims to bridge the gap between theoretical knowledge and practical application, equipping students with the necessary skills and expertise to thrive in the branch related industry.

At the end of the **Second semester**, students will undertake a minimum 3 to 4-week Summer Internship, scheduled during the semester break following the Second Semester examinations. The respective Heads of Departments (HoDs) and experienced faculty members will guide and assist students in securing suitable training opportunities that align with their specialization. Each student will have a personalized training schedule developed in collaboration with the training providers, ensuring a comprehensive and enriching learning experience.

Before starting their training, students will receive a comprehensive briefing about the organizational setup, product range, manufacturing processes, and significant machinery and materials used in the training organization. This preliminary understanding will enhance their engagement and productivity during the internship.

To ensure a fruitful learning experience, faculty members will supervise students during their training in the industry or field organization. Each faculty member will mentor a small group of 4-5 students, providing personalized attention and guidance. Students will be encouraged to maintain daily reports in their diaries, which will assist them in composing their final training report and presentation.

The evaluation process for the Summer Internship will include both internal and external assessments, as per the study and evaluation scheme of the **Third Semester**. During the viva - voce / presentation examination, students' understanding of materials, industrial processes, practices in the industry, and problem-solving abilities will be assessed. The evaluation will also focus on their application of knowledge and skills in real-life situations.

The components of evaluation will comprise:

(a) Punctuality and regularity: 15%

(b) Initiative in learning new things: 15%

(c) Relationship with peers and colleagues: 10%

(d) Summer Internship report: 25%

(e) Viva-Voce: 35%

The purpose of summer Internship program is to create a transformative experience for students, empowering them to excel in their future careers and make meaningful contributions to the Engineering industry. The collaborative efforts of experienced faculty members and industry partners will ensure that students may gain valuable insights and practical skills during this immersive learning journey.

*****