

**ANNAMACHARYA INSTITUTE OF TECHNOLOGY & SCIENCES::KADAPA
(AUTONOMOUS)**



UTUKUR (P), C. K. DINNE (V&M), KADAPA, YSR DIST.
Accredited by NAAC with 'A' Grade & NBA (B. Tech-CSE, ECE & EEE)

COMPUTER SCIENCE AND ENGINEERING- MINORS

S.No.	Code	Course Name	Contact Hours per week			Credits
			L	T	P	
1	23HMD0501	Data Structures And Algorithms	3	-	0	3
2	23HMD0502	Software Engineering	3	-	0	3
3	23HMD0503	Computer Organization	3	-	0	3
4	23HMD0504	Web Technologies	3	-	0	3
5	23HMD0505	Computer Networks	3	-	0	3
6	23HMD0506	Data Structures and Algorithms Lab	0	0	3	1.5
7	23HMD0507	Web Technologies Lab	0	0	3	1.5

23HMD0501	DATA STRUCTURES & ALGORITHMS	L	T	P	C
		3	0	0	3

Course Objectives:

The main objectives of the course is to

- provide knowledge on advance data structures frequently used in Computer Science domain
- Develop skills in algorithm design techniques popularly used
- Understand the use of various data structures in the algorithm design

Course Outcomes:

After completion of the course, students will be able to

1. Illustrate the working of the advanced tree data structures and their applications (L2)
2. Understand the Graph data structure, traversals and apply them in various contexts. (L2)
3. Use various data structures in the design of algorithms (L3)
4. Recommend appropriate data structures based on the problem being solved (L5)
5. Analyze algorithms with respect to space and time complexities (L4)

UNIT – I:

Introduction to Algorithm Analysis, Space and Time Complexity analysis, Asymptotic Notations.

AVL Trees – Creation, Insertion, Deletion operations and Applications

B-Trees – Creation, Insertion, Deletion operations and Applications

UNIT – II:

Heap Trees (Priority Queues) – Min and Max Heaps, Operations and Applications

Graphs – Terminology, Representations, Basic Search and Traversals, Connected Components and Biconnected Components, applications

Divide and Conquer: The General Method, Quick Sort, Merge Sort, Strassen’s matrix multiplication, Convex Hull

UNIT – III:

Greedy Method: General Method, Job Sequencing with deadlines, Knapsack Problem, Minimum cost spanning trees, Single Source Shortest Paths

Dynamic Programming: General Method, All pairs shortest paths, Single Source Shortest Paths– General Weights (Bellman Ford Algorithm), Optimal Binary Search Trees, 0/1 Knapsack, String Editing, Travelling Salesperson problem

UNIT – IV:

Backtracking: General Method, 8-Queens Problem, Sum of Subsets problem, Graph Coloring, 0/1 Knapsack Problem

Branch and Bound: The General Method, 0/1 Knapsack Problem, Travelling Salesperson problem

UNIT – V:

NP Hard and NP Complete Problems: Basic Concepts, Cook’s theorem

NP Hard Graph Problems: Clique Decision Problem (CDP), Chromatic Number Decision Problem (CNDP), Traveling Salesperson Decision Problem (TSP)

NP Hard Scheduling Problems: Scheduling Identical Processors, Job Shop Scheduling.

Textbooks:

1. Fundamentals of Data Structures in C++, Horowitz, Ellis; Sahni, Sartaj; Mehta, Dinesh, 2nd Edition Universities Press
2. Computer Algorithms in C++, Ellis Horowitz, SartajSahni, SanguthevarRajasekaran, 2nd Edition University Press

Reference Books:

1. Data Structures and program design in C, Robert Kruse, Pearson Education Asia
2. An introduction to Data Structures with applications, Trembley & Sorenson, McGraw Hill
3. The Art of Computer Programming, Vol.1: Fundamental Algorithms, Donald E Knuth, Addison-Wesley, 1997.
4. Data Structures using C & C++: Langsam, Augenstein&Tanenbaum, Pearson, 1995
5. Algorithms + Data Structures &Programs: N. Wirth, PHI
6. Fundamentals of Data Structures in C++: Horowitz Sahni& Mehta, Galgottia Pub.
7. Data structures in Java: Thomas Standish, Pearson Education Asia

Online Learning Resources:

1. https://www.tutorialspoint.com/advanced_data_structures/index.asp
2. <http://peterindia.net/Algorithms.html>
3. https://www.youtube.com/playlist?list=PLDN4rrl48XKpZkf03iYF1-O29szjTrs_O

23HMD0502	SOFTWARE ENGINEERING	L	T	P	C
		3	0	0	3

Course Objectives:

The objectives of this course are to introduce

- Software life cycle models, Software requirements and SRS document.
- Project Planning, quality control and ensuring good quality software.
- Software Testing strategies, use of CASE tools, Implementation issues, validation & verification procedures.

Course Outcomes:

After completion of the course, students will be able to

1. Perform various life cycle activities like Analysis, Design, Implementation, Testing and Maintenance (L3)
2. Analyse various software engineering models and apply methods for design and development of software projects. (L4)
3. Develop system designs using appropriate techniques. (L3)
4. Understand various testing techniques for a software project. (L2)
5. Apply standards, CASE tools and techniques for engineering software projects (L3)

UNIT I:

Introduction: Evolution, Software development projects, Exploratory style of software developments, Emergence of software engineering, Notable changes in software development practices, Computer system engineering.

Software Life Cycle Models: Basic concepts, Waterfall model and its extensions, Rapid application development, Agile development model, Spiral model.

UNIT II:

Software Project Management: Software project management complexities, Responsibilities of a software project manager, Metrics for project size estimation, Project estimation techniques, Empirical Estimation techniques, COCOMO, Halstead's software science, risk management.

Requirements Analysis and Specification: Requirements gathering and analysis, Software Requirements Specification (SRS), Formal system specification, Axiomatic specification, Algebraic specification, Executable specification and 4GL.

UNIT III:

Software Design: Overview of the design process, how to characterize a good software design? Layered arrangement of modules, Cohesion and Coupling. approaches to software design.

Agility: Agility and the Cost of Change, Agile Process, Extreme Programming (XP), Other Agile Process Models, Tool Set for the Agile Process (Text Book 2)

Function-Oriented Software Design: Overview of SA/SD methodology, Structured analysis, Developing the DFD model of a system, Structured design, Detailed design, and Design Review.

User Interface Design: Characteristics of a good user interface, Basic concepts, Types of user interfaces, Fundamentals of component-based GUI development, and user interface design methodology.

UNIT IV:

Coding And Testing: Coding, Code review, Software documentation, Testing, Black-box testing, White-Box testing, Debugging, Program analysis tools, Integration testing, testing object-oriented programs, Smoke testing, and some general issues associated with testing.

Software Reliability and Quality Management: Software reliability. Statistical testing, Software quality, Software quality management system, ISO 9000. SEI Capability maturity model. Few other important quality standards, and Six Sigma.

UNIT V:

Computer-Aided Software Engineering (Case): CASE and its scope, CASE environment, CASE support in the software life cycle, other characteristics of CASE tools, Towards second generation CASE Tool, and Architecture of a CASE Environment.

Software Maintenance: Characteristics of software maintenance, Software reverse engineering, Software maintenance process models and Estimation of maintenance cost.

Software Reuse: reuse- definition, introduction, reason behind no reuse so far, Basic issues in any reuse program, A reuse approach, and Reuse at organization level.

Text Books:

1. Fundamentals of Software Engineering, Rajib Mall, 5th Edition, PHI.
2. Software Engineering a Practitioner's Approach, Roger S. Pressman, 9th Edition, Mc-Graw Hill International Edition.

Reference Books:

1. Software Engineering, Ian Sommerville, 10th Edition, Pearson.
2. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.

e-Resources:

- 1) <https://nptel.ac.in/courses/106/105/106105182/>
- 2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01260589506387148827_s_hared/overview
- 3) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013382690411003904735_shared/overview

23HMD0503	COMPUTER ORGANIZATION	L	T	P	C
		3	0	0	3

Course Objectives:

- To learn the fundamentals of computer organization and its relevance to classical and modern problems of computer design
- To understand the structure and behavior of various functional modules of a computer.
- To learn the techniques that computers use to communicate with I/O devices
- To acquire the concept of pipelining and exploitation of processing speed.
- To learn the basic characteristics of multiprocessors

Course Outcomes (CO):

After completion of the course, students will be able to

- Understand computer architecture concepts related to the design of modern processors, memories and I/Os
- Identify the hardware requirements for cache memory and virtual memory
- Design algorithms to exploit pipelining and multiprocessors
- Understand the importance and trade-offs of different types of memories.
- Identify pipeline hazards and possible solutions to those hazards

UNIT - I

Basic Structure of Computer: Computer Types, Functional Units, Basic operational Concepts, Bus Structure, Software, Performance, Multiprocessors and Multicomputer.

Machine Instructions and Programs: Numbers, Arithmetic Operations and Programs, Instructions and Instruction Sequencing, Addressing Modes, Basic Input/output Operations, Stacks and Queues, Subroutines, Additional Instructions.

UNIT - II

Arithmetic: Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed-operand Multiplication, Fast Multiplication, Integer Division, Floating-Point Numbers and Operations.

Basic Processing Unit: Fundamental Concepts, Execution of a Complete Instruction, Multiple-Bus Organization, Hardwired Control, and Multi programmed Control.

UNIT - III

The Memory System: Basic Concepts, Semiconductor RAM Memories, Read-Only Memories, Speed, Size and Cost, Cache Memories, Performance Considerations, Virtual Memories, Memory Management Requirements, Secondary Storage.

UNIT - IV

Input/Output Organization: Accessing I/O Devices, Interrupts, Processor Examples, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces.

UNIT - V

Pipelining: Basic Concepts, Data Hazards, Instruction Hazards, Influence on Instruction Sets.

Large Computer Systems: Forms of Parallel Processing, Array Processors, The Structure of General-Purpose multiprocessors, Interconnection Networks.

Textbooks:

1. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, "Computer Organization", 5th Edition, McGraw

Hill Education, 2013.

Reference Books:

1. M. Morris Mano, “Computer System Architecture”, 3rd Edition, Pearson Education.
2. Themes and Variations, Alan Clements, “Computer Organization and Architecture”, CENGAGE Learning.
3. SmrutiRanjanSarangi, “Computer Organization and Architecture”, McGraw Hill Education.
4. John P. Hayes, “Computer Architecture and Organization”, McGraw Hill Education

Online Learning Resources:

<https://nptel.ac.in/courses/106/103/106103068/>

23HMD0504	WEB TECHNOLOGIES	L	T	P	C
		3	0	0	3

Course Objectives:

The course is designed to Introduce the key technologies that have been developed as part of the birth and maturation of the World Wide Web.

Course Outcomes:

- Understand the Web essentials.
- Develop web pages using XHTML
- Apply style to web pages using CSS
- Write scripts for client side
- Develop and transform XML documents.

UNIT I

Web Essentials: Clients, Servers, and Communication, The Internet, Basic Internet protocols, WWW, HTTP request message, HTTP response message, Web clients, Web Servers, Case study.

UNIT II

Markup Languages: XHTML 1.0, An introduction to HTML, Basic XHTML syntax and semantics, fundamental HTML elements, Relative URLs, Lists, Tables, Frames, Forms, Defining XHTML's abstract syntax, Creating HTML documents.

UNIT III

Cascading Style Sheets: Introduction, features, core syntax, style sheets and HTML, style rule cascading and inheritance, text properties, Box model, normal flow box layout, beyond the normal flow, lists, tables, cursor styles.

UNIT IV

Client-side programming - JavaScript: Basic syntax, variables and data types, statements, operators, literals, functions, objects, Arrays, built-in objects, JavaScript debuggers.

UNIT V

Representing Web Data-XML: Documents and vocabularies, Versions and declaration, Namespaces, Ajax, DOM and SAX parsers, transforming XML documents, XPath, XSLT, Displaying XML documents in Web browsers.

Textbooks:

1. J.C. Jackson, Web technologies: A computer science perspective, Pearson.
- 2.

Reference Books:

1. Sebesta, Programming world wide web, Pearson.
2. Dietel and Nieto, Internet and World Wide Web –How to program, Pearson Education
3. Chris Bates, Web Programming, building internet applications, 2nd edition, WILEY, Dreamtech

Online Learning Resources:

<http://getbootstrap.com/>
<https://www.w3schools.com/whatis/>
<https://nptel.ac.in/courses/106105084>

23HMD0505	COMPUTER NETWORKS	L	T	P	C
		3	0	0	3

Course Objectives:

The course is designed to

- Understand the basic concepts of Computer Networks.
- Introduce the layered approach for design of computer networks
- Expose the network protocols used in Internet environment
- Explain the format of headers of IP, TCP and UDP
- Familiarize with the applications of Internet
- Elucidate the design issues for a computer network

Course Outcomes:

After completion of the course, students will be able to

- Identify the software and hardware components of a computer network
- Design software for a computer network
- Develop new routing, and congestion control algorithms
- Assess critically the existing routing protocols
- Explain the functionality of each layer of a computer network
- Choose the appropriate transport protocol based on the application requirements

UNIT I

Computer Networks and the Internet : What Is the Internet? The Network Edge, The Network Core, Delay, Loss, and Throughput in Packet Switched Networks (Textbook 2), Reference Models, Example Networks, Guided Transmission Media, Wireless Transmission (Textbook 1)

UNIT II

The Data Link Layer, Access Networks, and LANs: Data Link Layer Design Issues, Error Detection and Correction, Elementary Data Link Protocols, Sliding Window Protocols, Introduction to the Link Layer, Error-Detection and -Correction Techniques, Multiple Access Links and Protocols, Switched Local Area Networks

Link Virtualization: A Network as a Link Layer, Data Center Networking, Retrospective: A Day in the Life of a Web Page Request (Textbook 2)

UNIT III

The Network Layer: Routing Algorithms, Internetworking, The Network Layer in The Internet (Textbook 1)

UNIT IV

The Transport Layer: Connectionless Transport: UDP (Textbook 2), The Internet Transport Protocols: TCP, Congestion Control (Textbook 1)

UNIT V

Principles of Network Applications: Principles of Network Applications, The Web and HTTP, Electronic Mail in the Internet, DNS—The Internet’s Directory Service, Peer-to-Peer Applications Video Streaming and Content Distribution Networks (Textbook 2)

Textbooks:

1. Andrew S. Tanenbaum, David j. wetherall, Computer Networks, 5th Edition, PEARSON.
2. James F. Kurose, Keith W. Ross, “Computer Networking: A Top-Down Approach”, 6th edition, Pearson, 2019.

Reference Books:

1. Forouzan, Datacommunications and Networking, 5th Edition, McGraw Hill Publication.
2. Youlu Zheng, ShakilAkthar, “Networks for Computer Scientists and Engineers”, Oxford Publishers, 2016.

Online Learning Resources:

1. <https://nptel.ac.in/courses/106105183/25>
2. <http://www.nptelvideos.in/2012/11/computer-networks.html>
3. <https://nptel.ac.in/courses/106105183/3>

23HMD0506	DATA STRUCTURES & ALGORITHMS LAB	L	T	P	C
		0	0	3	1.5

Course Objectives:

The objectives of the course is to

- acquire practical skills in constructing and managing Data structures
- apply the popular algorithm design methods in problem-solving scenarios

Course Outcomes:

After completion of the course, students will be able to

- Design and develop programs to solve real world problems with the popular algorithm design methods. (L5)
- Demonstrate an understanding of Non-Linear data structures by developing implementing the operations on AVL Trees, B-Trees, Heaps and Graphs. (L2)
- Critically assess the design choices and implementation strategies of algorithms and data structures in complex applications. (L5)
- Utilize appropriate data structures and algorithms to optimize solutions for specific computational problems. (L3)
- Compare the performance of different of algorithm design strategies (L4)

Experiments covering the Topics:

- Operations on AVL trees, B-Trees, Heap Trees
- Graph Traversals
- Sorting techniques
- Finding Biconnected components in a graph
- Shortest path algorithms using greedy Method
- 0/1 Knapsack Problem using Dynamic Programming and Backtracking
- Travelling Salesperson problem using Branch and Bound
- N-Queens Problem using Backtracking
- Job Sequencing using Branch and Bound

Sample Programs:

1. Construct an AVL tree for a given set of elements which are stored in a file. And implement insert and delete operation on the constructed tree. Write contents of tree into a new file using in-order.
2. Construct B-Tree an order of 5 with a set of 100 random elements stored in array. Implement searching, insertion and deletion operations.
3. Construct Min and Max Heap using arrays, delete any element and display the content of the Heap.
4. Implement BFT and DFT for given graph, when graph is represented by
 - a) Adjacency Matrix
 - b) Adjacency Lists
5. Write a program for finding the biconnected components in a given graph.
6. Implement Quick sort and Merge sort and observe the execution time for various input sizes (Average, Worst and Best cases).
7. Compare the performance of Single Source Shortest Paths using Greedy method when the graph is represented by adjacency matrix and adjacency lists.
8. Implement Job Sequencing with deadlines using Greedy strategy.
9. Write a program to solve 0/1 Knapsack problem Using Dynamic Programming.
10. Implement N-Queens Problem Using Backtracking.
11. Use Backtracking strategy to solve 0/1 Knapsack problem.
12. Implement Travelling Sales Person problem using Branch and Bound approach.

Reference Books:

1. Fundamentals of Data Structures in C++, Horowitz Ellis, SahniSartaj, Mehta, Dinesh, 2nd Edition, Universities Press
2. Computer Algorithms/C++ Ellis Horowitz, SartajSahni, SanguthevarRajasekaran, 2nd Edition, University Press
3. Data Structures and program design in C, Robert Kruse, Pearson Education Asia
4. An introduction to Data Structures with applications, Trembley & Sorenson, McGraw Hill

Online Learning Resources:

1. <http://cse01-iiith.vlabs.ac.in/>
2. <http://peterindia.net/Algorithms.html>

23HMD0507	WEB TECHNOLOGIES LAB	L	T	P	C
		0	0	3	1.5

Course Objectives:

- Learn website development using HTML, CSS, JavaScript.
- Understand the concepts of responsive web development using the bootstrap framework
- Make use of the JQueryjavascript library to provide interactiveness to the websites.
- Discover how to use Google Charts to provide a better way to visualize data on a website
- Learn Content Management Systems to speed the development process

Course Outcomes (CO):

After completion of the course, students will be able to

- Construct web sites with valid HTML, CSS, JavaScript
- Create responsive Web designs that work on phones, tablets, or traditional laptops and widescreen monitors.
- Develop websites using jQuery to provide interactivity and engaging user experiences
- Embed Google chart tools in a website for better visualization of data.
- Design and develop web applications using Content Management Systems like WordPress

Sample Programs:

1. Create a Basic HTML document
2. Create your Profile Page
3. Create a Class Timetable (to merge rows/columns, use rowspan/colspan)
4. Create a Student Hostel Application Form
5. Make the Hostel Application Form designed in Module -4 beautiful using CSS (add colors, backgrounds, change font properties, borders, etc.)
6. Style the Hostel Application Form designed in Module-5still more beautiful using Bootstrap
7. Analyse various HTTP requests (initiators, timing diagrams, responses) and identify problems if any.
8. Design a simple calculator using JavaScript to perform sum, product, difference, and quotient operations:
9. Design& develop a Shopping Cart Application with features including Add Products, Update Quantity, Display Price (Sub-Total & Total), Remove items/products from the cart.
10. Validate all Fields and Submit the Hostel Application Form designed in Module-6 using JQuery
11. Develop an HTML document to illustrate each chart with real-time examples.
12. Develop an E-learning website using any CMS (for example WordPress)

References:

1. Deitel and Deitel and Nieto, —Internet and World Wide Web - How to Program, PrenticeHall, 5th Edition, 2011.
2. Web Technologies, Uttam K. Roy, Oxford Higher Education., 1st edition, 10th impression, 2015.
3. Stephen Wynkoop and John Burke —Running a Perfect Websitel, QUE, 2nd Edition,1999.
4. Jeffrey C and Jackson, —Web Technologies A Computer Science PerspectivePearsonEducation, 2011.
5. Gopalan N.P. and Akilandeswari J., —Web Technology, Prentice Hall of India, 2011.

Online Learning Resources/Virtual Labs:

- a.HTML: <https://html.spec.whatwg.org/multipage/>
- b.HTML: <https://developer.mozilla.org/en-US/docs/Glossary/HTML5>
- c.CSS: <https://www.w3.org/Style/CSS/>
- d.Bootstrap - CSS Framework: <https://getbootstrap.com/>
- e.BrowserDeveloperTools:https://developer.mozilla.org/enUS/docs/Learn/Common_questions/What_are_browser_developer_tools
- f. Javascript: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- g. JQuery: <https://jquery.com>
- h. Google Charts: <https://developers.google.com/chart>
- i. Wordpress: <https://wordpress.com>

