



## ANDHRA PRADESH STATE COUNCIL OF HIGHER EDUCATION

(A Statutory body of the Government of Andhra Pradesh)

3<sup>rd</sup>, 4<sup>th</sup> and 5<sup>th</sup> floors, Neeladri Towers, Sri Ram Nagar, 6<sup>th</sup> Battalion Road,  
Atmakur(V), Mangalagiri(M), Guntur-522 503, Andhra Pradesh  
Web: [www.apsche.org](http://www.apsche.org) Email: [acapsche@gmail.com](mailto:acapsche@gmail.com)

REVISED SYLLABUS OF B.Sc. (COMPUTER SCIENCE/ INFORMATION  
TECHNOLOGY) UNDER CBCS FRAMEWORK WITH EFFECT FROM 2020-2021

PROGRAMME: THREE-YEAR B.Sc.

(B.Sc. Computer Science/ Information Technology (IT))

# COMPUTER SCIENCE SYLLABUS SEM – 2

(With Learning Outcomes, Unit-wise Syllabus, References, Co-curricular Activities &  
Model Q.P.)

For Fifteen Courses of 1, 2, 3 & 4 Semesters)  
(To be Implemented from 2020-21 Academic Year)

Structure of Computer Science /Information Technology (IT)

Programme: B.Sc. with Computer Science as one of the Core Subjects.

Discipline: Computer Science

| Year        | Semester | Paper Code | Subject                                    | Hrs. per Week | Credits | IA | ES | Total |
|-------------|----------|------------|--------------------------------------------|---------------|---------|----|----|-------|
| First Year  | I        | C1         | Problem Solving in C                       | 4             | 3       | 25 | 75 | 100   |
|             | I        | C1-P       | Problem Solving in C Lab                   | 2             | 2       |    | 50 | 50    |
|             | II       | C2         | Data Structures using C                    | 4             | 3       | 25 | 75 | 100   |
|             | II       | C2-P       | Data Structures using C Lab                | 2             | 2       |    | 50 | 50    |
| Second Year | III      | C3         | Database Management System                 | 4             | 3       | 25 | 75 | 100   |
|             | III      | C3-P       | Database Management System Lab             | 2             | 2       |    | 50 | 50    |
|             | IV       | C4         | Object Oriented Programming using Java     | 4             | 3       | 25 | 75 | 100   |
|             | IV       | C4-P       | Object Oriented Programming using Java Lab | 2             | 2       |    | 50 | 50    |
|             | IV       | C5         | Operating Systems                          | 4             | 3       | 25 | 75 | 100   |
|             | IV       | C5-P       | Operating Systems Lab using C/Java         | 2             | 2       |    | 50 | 50    |

## DATA STRUCTURES USING C

| Semester | Course Code | Course Title            | Hours | Credits |
|----------|-------------|-------------------------|-------|---------|
| II       | C2          | DATA STRUCTURES USING C | 60    | 3       |

### Course Objectives

To introduce the fundamental concept of data structures and to emphasize the importance of various data structures in developing and implementing efficient algorithms.

### UNIT - I:

Introduction to Data Structures: Introduction to the Theory of Data Structures, Data Representation, Abstract Data Types, Data Types, Primitive Data Types, Data Structure and Structured Type, Atomic Type, Difference between Abstract Data Types, Data Types, and Data Structures, Refinement Stages

Principles of Programming and Analysis of Algorithms: Software Engineering, Program Design, Algorithms, Different Approaches to Designing an Algorithm, Complexity, Big 'O' Notation, Algorithm Analysis, Structured Approach to Programming, Recursion, Tips and Techniques for Writing Programs in 'C'

## UNIT - II:

Arrays: Introduction to Linear and Non- Linear Data Structures, One- Dimensional Arrays, Array Operations, Two- Dimensional arrays, Multidimensional Arrays, Pointers and Arrays, an Overview of Pointers

Linked Lists: Introduction to Lists and Linked Lists, Dynamic Memory Allocation, Basic Linked List Operations, Doubly Linked List, Circular Linked List, Atomic Linked List, Linked List in Arrays, Linked List versus Arrays

## UNIT - III:

Stacks: Introduction to Stacks, Stack as an Abstract Data Type, Representation of Stacks through Arrays, Representation of Stacks through Linked Lists, Applications of Stacks, Stacks and Recursion

Queues: Introduction, Queue as an Abstract data Type, Representation of Queues, Circular Queues, Double Ended Queues- Deques, Priority Queues, Application of Queues

## UNIT - IV:

Binary Trees: Introduction to Non- Linear Data Structures, Introduction Binary Trees, Types of Trees, Basic Definition of Binary Trees, Properties of Binary Trees, Representation of Binary Trees, Operations on a Binary Search Tree, Binary Tree Traversal, Counting Number of Binary Trees, Applications of Binary Tree

## UNIT - V:

Searching and sorting: Sorting - An Introduction, Bubble Sort, Insertion Sort, Merge Sort, Searching - An Introduction, Linear or Sequential Search, Binary Search, Indexed Sequential Search

Graphs: Introduction to Graphs, Terms Associated with Graphs, Sequential Representation of Graphs, Linked Representation of Graphs, Traversal of Graphs, Spanning Trees, Shortest Path, Application of Graphs.

## BOOKS:

1. "Data Structures using C", ISRD group Second Edition, TMH
2. "Data Structures through C", Yashavant Kanetkar, BPB Publications
3. "Data Structures Using C" Balagurusamy E. TMH

## RECOMMENDED CO-CURRICULAR ACTIVITIES:

(Co-curricular activities shall not promote copying from textbook or from others work and shall encourage self/independent and group learning)

### A. Measurable

1. Assignments (in writing and doing forms on the aspects of syllabus content and outside the syllabus content. Shall be individual and challenging)
2. Student seminars (on topics of the syllabus and related aspects (individual activity))
3. Quiz (on topics where the content can be compiled by smaller aspects and data (Individuals or groups as teams))
4. Study projects (by very small groups of students on selected local real-time problems pertaining to syllabus or related areas. The individual participation and contribution of students shall be ensured (team activity))

### B. General

1. Group Discussion
2. Others

## RECOMMENDED CONTINUOUS ASSESSMENT METHODS:

Some of the following suggested assessment methodologies could be adopted;

1. The oral and written examinations (Scheduled and surprise tests),
2. Closed-book and open-book tests,
3. Programming exercises,
4. Practical assignments and laboratory reports,
5. Observation of practical skills,
6. Individual and group project reports.
7. Efficient delivery using seminar presentations,
8. Viva voce interviews.
9. Computerized adaptive testing, literature surveys and evaluations,
10. Peers and self-assessment, outputs form individual and collaborative work

| Semester | Course Code | Course Title            | Hours | Credits |
|----------|-------------|-------------------------|-------|---------|
| II       | C2-P        | DATA STRUCTURES USING C | 30    | 2       |
|          |             | LAB                     |       |         |

1. Write a program to read 'N' numbers of elements into an array and also perform the following operation on an array
  - a. Add an element at the beginning of an array
  - b. Insert an element at given index of array
  - c. Update an element using a value and index
  - d. Delete an existing element
2. Write a program using stacks to convert a given
  - a. postfix expression to prefix
  - b. prefix expression to postfix
  - c. infix expression to postfix
3. Write Programs to implement the Stack operations using an array
4. Write Programs to implement the Stack operations using Linked List.
5. Write Programs to implement the Queue operations using an array.
6. Write Programs to implement the Queue operations using Linked List.
7. Write a program for arithmetic expression evaluation.
8. Write a program for Binary Search Tree Traversals
9. Write a program to implement dequeue using a doubly linked list.
10. Write a program to search an item in a given list using the following Searching Algorithms
  - a. Linear Search
  - b. Binary Search.
11. Write a program for implementation of the following Sorting Algorithms
  - a. Bubble Sort
  - b. Insertion Sort
  - c. Quick Sort
12. Write a program for polynomial addition using single linked list
13. Write a program to find out shortest path between given Source Node and Destination Node in a given graph using Dijkstra's algorithm.
14. Write a program to implement Depth First Search graph traversals algorithm
15. Write a program to implement Breadth First Search graph traversals algorithm.