KAKATIYA UNIVERSITY

Under Graduate Courses (Under CBCS 2020–2021onwords)

B.Sc. Computer Science II Year
SEMESTER – III

DATA STRUCTURES USING C++

Theory:	**4 Hours/Week;**	**Credits: 4**	**Marks: 100 (Internal: 20; External: 80)**
Practical:	**3 Hours/Week**	**Credits: 1**	**Marks: 25**

Unit - I

Basic data Structure: Introduction to Data Structures, Types of Data Structures, and Introduction to Algorithms, Pseudo code, and Relationship among data, data structures, and algorithms, Implementation of data structures, Analysis of Algorithms.

Stacks: Concept of Stacks and Queues, Stacks, Stack Abstract Data Type, Representation of Stacks Using Sequential Organization (Arrays), Multiple Stacks, Applications of Stack, Expression Evaluation and Conversion, Polish notation and expression conversion, Processing of Function Calls, Reversing a String with a Stack, Recursion.

Unit - II

Recursion: Introduction, Recurrence, Use of Stack in Recursion, Variants of Recursion, Recursive Functions, Iteration versus Recursion.

Queues: Concept of Queues, Queue as Abstract Data Type, Realization of Queues Using Arrays, Circular Queue, Multi-queues, Dequeue, Priority Queue, Applications of Queues,

Linked Lists: Introduction, Linked List, Linked List Abstract Data Type, Linked List Variants, Doubly Linked List, Circular Linked List, Representation of Sparse Matrix Using Linked List, Linked Stack, Linked Queue.

Unit - III

Trees: Introduction, Types of Trees, Binary Tree, Binary Tree Abstract Data Type, Realization of a Binary Tree, Insertion of a Node in Binary Tree, Binary Tree Traversal, Other Tree Operations, Binary Search Tree, Threaded Binary Tree, Applications of Binary Trees.

Searching and Sorting: Search Techniques-Linear Search, Binary Search, Sorting Techniques- Selection Sort, Bubble Sort, Insertion Sort, Merge Sort, Quick Sort, Comparison of All Sorting Methods, Search Trees: Symbol Table, Optimal Binary Search Tree, AVL Tree (Height-balanced Tree).

Unit - IV

Graphs: Introduction, Representation of Graphs, Graph Traversal – Depth First Search, Breadth First Search, Spanning Tree, Prim’s Algorithm, Kruskal’s Algorithm.

Hashing: Introduction, Key Terms and Issues, Hash Functions, Collision Resolution Strategies, Hash Table Overflow, Extendible Hashing

Heaps: Basic Concepts, Implementation of Heap, Heap as Abstract Data Type, Heap Sort, Heap Applications.

Text books:

1. Varsha H. Patil “Data structures using C++” Oxford University press, 2012
2. M.T. Goodrich, R. Tamassia and D. Mount, Data Structures and Algorithms in C++, John Wiley and Sons, Inc., 2011.

References:

1. Adam Drozdek “Data structures and algorithm in C++” Second edition, 2001
2. T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein, Introduction to Algorithms,2nd Ed., Prentice-Hall of India, 2006.
3. Robert L. Kruse and A.J. Ryba, Data Structures and Program Design in C++, PrenticeHall, Inc., NJ, 1998.
4. B. Stroupstrup, The C++ Programming Language, Addison Wesley, 2004
5. D.E. Knuth, Fundamental Algorithms (Vol. I), Addison Wesley, 1997

Dr. B.Rama Chairperson Board of Studies, Department of Computer Science, KU

KAKATIYA UNIVERSITY

Under Graduate Courses (Under CBCS 2020–2021 onwards)

B.Sc. Computer Science II Year

SEMESTER – III

DATA STRUCTURES USING C++ LAB

Practical 3 Hours/Week 1 Credit Marks: 25

Note:

- Programs of all the Concepts from Text Book including exercises must be practice and execute.
- In the external lab examination student has to execute two programs with compilation and deployment steps are necessary.
- External Vice-Voce is compulsory.

1. Write C++ programs to implement the following using an array

   a) Stack ADT b) Queue ADT

2. Write a C++ program to implement Circular queue using array.
3. Write C++ programs to implement the following using a single linked list.

   a) Stack ADT b) Queue ADT

4. Write a C++ program to implement Circular queue using Single linked list.
5. Write a C++ program to implement the double ended queue ADT using double linked list.
6. Write a C++ program to solve tower of Hanoi problem recursively
7. Write C++ program to perform the following operations:

   a) Insert an element into a binary search tree.

   b) Delete an element from binary search tree.

   c) Search for a key in a binary search tree.

8. Write C++ programs for the implementation tree traversal technique BFS.
9. Write a C++ program that uses recursive functions to traverse a binary search tree.

   a) Pre-order b) In-order c) Post-order

10. Write a C++ program to find height of a tree.

11 Write a C++ program to find MIN and MAX element of a BST.

12 Write a C++ program to find Inorder Successor of a given node.

13. Write C++ programs to perform the following operations on B-Trees and AVL Trees.

    a) Insertion b) Deletion

14 Write C++ programs for sorting a given list of elements in ascending order using the following sorting methods.

   a) Quick sort b) Merge sort

15. Write a C++ program to find optimal ordering of matrix multiplication.
16. Write a C++ program that uses dynamic programming algorithm to solve the optimal binary search tree problem
17. Write a C++ program to implement Hash Table
18. Write C++ programs to perform the following on Heap

    a) Build Heap b) Insertion c) Deletion

19. Write C++ programs to perform following operations on Skip List

    a) Insertion b) Deletion

20. Write a C++ Program to Create a Graph using Adjacency Matrix Representation.
21. Write a C++ program to implement graph traversal techniques

    a) BFS b) DFS

22. Write a C++ program to Heap sort using tree structure.

Dr. B.Rama Chairperson Board of Studies, Department of Computer Science, KU