

**Data Structures and Algorithms (CSC-154)**  
**Tribhuvan University**  
**Soch College of Information Technology**  
**Bachelor of Science in Computer Science and Information Technology**

**Course Title:** Data Structures and Algorithms

**Course no:** CSC-154 ----- Full Marks: 60+20+20

**Credit hours:** 3 ----- Pass Marks: 24+8+8

**Nature of course:** Theory (3 Hrs.) + Lab (3 Hrs.)

**Course Synopsis:** Study of basic data structure vocabulary, the concept of an algorithm.

**Goal:** To provide the concept of data structure and its implementation using programming techniques.

**Course Contents:**

**Unit 1:** ----- 14 Hrs.

1.1 Introduction to Data Structures: Information and its meaning, Array in C: The array as an ADT, Using one dimensional array, Two dimensional array, Multi dimensional array, Structure , Union, Classes in C++.

1.2 The Stack: Introduction, definition, primitive operation, the stack as an abstract data type, implementing the POP operation, testing for exceptional condition, implementing the PUSH operation.

1.3 The Infix, Postfix & Prefix: Introduction, evaluating the postfix operation, program to evaluate the postfix operation, limitation of program, converting from one to another.

1.4 Recursion: Introduction, factorial functions, multiplication of natural numbers, Fibonacci sequence, binary search, the tower of Hanoi problem, translation from prefix to postfix using recursion.

**Unit 2:** ----- 31 Hrs.

1.1 Queues: Introduction, the queue and its sequential representation: The queue as an abstract data type, implementation of queue, inserts operation, priority queue.

1.2 Linked Lists: Introduction, inserting and deleting the nodes from a list, linked implementation of stack, getnode and freenode operation, linked implementation of queue, Linked list as a data structure, circular lists, stack as a circular list, queue as a circular list.

1.3 Tree: Introduction, Binary Trees: operation on Binary Trees, application of Binary Trees. Binary Tree Representation: node representation of binary tree, internal and external nodes, implicit array representation of binary tree, binary tree traversal, threaded binary tree, heterogonous binary tree. The Huffman algorithm. Representing lists as binary trees. Trees and their application.

1.4 Sorting: Introduction, O notation, efficiency of sorting, exchange sort: bubble sort, quick sort. 1.5 Selection and Tree Sorting: Introduction, straight selection sort, binary tree sort, heapsort, insertion sort, merge and radix sort.

1.6 Searching: Introduction, sequential searching, binary search, interpolation search, tree search,

general search tree, hashing.

1.7 Graphs: Introduction, linked representation of graphs. 1.8 Algorithm: Introduction, design of algorithm, algorithm validation, analysis of algorithm, algorithm testing. subalgorithm

**Laboratory works:**

Write a code to multiply two matrixes and get the transpose of the third one.

Write a code to implement the stack, that should check overflow and underflow also.

Write a code to convert any prefix number to postfix.

Write a code to convert any infix number to postfix.

Write a code to convert any post fix number to prefix.

Implement tower of Hanoi.

Write a code to implement different sorting techniques.

Write a code to demonstrate the binary search.

Write a code to implement the queue.

Write a code to convert stack operation to queue operation.

**Text books:**

Data Structure Using C & C++, Langsam Yedidyah, Augenstein Moshe J., Tennenbaum Aaron M., PHI

**Reference:**

The Design and Analysis of Algorithm, Nitin Upadhyay, SK Kataria & Sons.

**Homework**

**Assignment:** Assignment should be given from the above units in throughout the semester.

**Computer usage:** No specific

**Prerequisite:** C, C++