

PRACTICAL FILE

Student Name	SANJIV GUPTA
UID	21BCS3478
Section & Group	602-B
Department	Computer Science & Engineering
Session	July-Dec 2023
Course Name	Design and Analysis of Algorithms with Lab
Course Code	21ITH-311/21CSH-311
Semester	5 TH

Department of Computer Science & Engineering
Chandigarh University, Mohali



Course Name: DAA Lab

Course Code: 21ITH-311/21CSH-311

INDEX

S. No.	Experiment	Date	Conduct (12)	Viva (10)	Worksheet (8)	Total (30)	Remarks
1	Experiment 1.1: Analyze if stack Isempty, Isfull and if elements are present then return top element in stacks using templates and also perform push and pop operation in stack.						
2	Experiment 1.2: Develop a program for implementation of power function and determine that complexity should be $O(\log n)$.						
3	Experiment 1.3: Evaluate the complexity of the developed program to find frequency of elements in a given array.						
4	Experiment 1.4: i. Apply the concept of Linked list and write code to Insert and Delete an element at the beginning and end of Singly Linked List. ii. Apply the concept of Linked list and write code to Insert and Delete an element at the beginning and at end in Doubly and Circular Linked List.						
5	Experiment 2.1: Sort a given set of elements using the Quick sort method and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted. The elements can be read from a file or can be generated using the random number generator.						
6	Experiment 2.2 Develop a program and analyze complexity to implement subset-sum problem using Dynamic Programming.						

7	Experiment 2.3: Develop a program and analyze complexity to implement 0-1 Knapsack using Dynamic Programming.						
8	Experiment 3.1: Develop a program and analyze complexity to do a depth-first search (DFS) on an undirected graph. Implementing an application of DFS such as (i) to find the topological sort of a directed acyclic graph, OR (ii) to find a path from source to goal in a maze						
9	Experiment 3.2: Develop a program and analyze complexity to find shortest paths in a graph with positive edge weights using Dijkstra's algorithm.						
10	Experiment 3.3: Develop a program and analyze complexity to find all occurrences of a pattern P in a given string S.						