**Practical File**

of

**Data Structure and Algorithms Lab**

**(PCC-CS-303)**

submitted in partial fulfillment of the requirement for the award of degree of

**Bachelor of Technology (B.Tech)**

in

**Computer Engineering**

by

**Name**

**(Roll No.)**

Under the guidance of

**Mr. Piyush Gupta**

**Assistant Professor**



**Department of Computer Engineering**

**J. C. BOSE UNIVERSITY OF SCIENCE & TECHNOLOGY, YMCA**

**SECTOR-6 FARIDABAD**

**HARYANA-121006**

**INDEX**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No.** | **Problem** | **List of Programs** | **Page** |
|  | Searching | Given an array of integers *nums* and an integer *target*, write a function to search *target* in *nums*. If *target* exists, then return its index. Otherwise, return *-1*. |  |
|  | Given an array of integers *nums* which is sorted in ascending order, and an integer *target*, write a function to search target in *nums*. If target exists, then return its index. Otherwise, return *-1*. |  |
|  | Given a sorted array of *n* elements, possibly with duplicates, find the number of occurrences of the *target* element. |  |
|  | Given a **0-indexed** integer array *nums*, find a **peak element**, and return its index. If the array contains multiple peaks, return the index to **any of the peaks**.  \*A peak element is an element that is strictly greater than its neighbors |  |
|  | There is an integer array *nums* sorted in ascending order (with distinct values). After the possible **rotation** of the given array, find an integer *target*, return the index of *target* if it is in *nums*, or -1 if it is not in *nums*. |  |
|  | Given an array *arr* of positive integers sorted in a strictly increasing order, and an integer *k*.  Write a function to return the *kth* positive integer that is missing from this array. |  |
|  | Stack | Write a program to implement stack using array (Show all the operations like insertion, deletion and display) |  |
|  | Write a program to convert Infix expression into Postfix expression and also analyze its Complexity. |  |
|  | Write a program to evaluate the Postfix expression. |  |
|  | Queue | Write a program to implement Simple Queue using arrays (Show all the operations like insertion, deletion and display) |  |
|  | Write a program to implement Circular Queue using arrays (Show all the operations like insertion, deletion and display) |  |
|  | Write a program to implement Priority Queue using both ordered and unordered arrays (Show all the operations like insertion, deletion and display) |  |
|  |  |  |  |

**Searching**

**Program No. 1:** Given an array of integers *nums* and an integer *target*, write a function to search *target* in *nums*. If *target* exists, then return its index. Otherwise, return *-1*.

**Code:**

**Output:**

**Complexity:**