**LAB # 06**

**Searching in a Linear Array**

**OBJECTIVE:** To find an element in linear array using Linear Search and Binary Search.

A linear search works by looking at each element in a list of data until it either finds the target or reaches the end.

A binary search comes with the prerequisite that the data must be sorted. Search a sorted array by repeatedly dividing the search interval in half. Low keep tracks of the beginning of half and high keeps track of ending of half.

**Algorithm for Linear Search**

1. Set k = 1 & loc = 0

2. Repeat step 3 & 4 while loc = 0 &k < = n

3. If (item = data[k])

loc = k

Else

K = k + 1

4. If loc = 0 ,then

Print “no. not found” Else

Print “loc is the location of item”

5. Exit

**Algorithm for Binary Search**

1. low = 1,high = n, Loc=NULL

2. Repeat step 3 to 5 while low <= high

3. mid = (low + high)/2

4. If a[mid] = x

Loc= mid

5. If a[mid] < x

low = mid + 1

Else

High = mid – 1

7. Return Loc

**Lab Task**

1. Declare an array of size 10 to store account balances. Initialize with values 0 to 1000000. Check all array if any value is less than 10000. Show message:

Account No. Low Balance

Account No. Low Balance

1. Write a program to search in array using Array built-in class.
2. Given an unsorted array arr of integers, find the smallest positive integer that is **missing** from the array. You need to implement this using **binary search**. The array can contain both negative numbers and positive numbers, and you can assume that the array does not have duplicates.
3. You are given a sorted array arr[] and a target element target. Your task is to find the **first occurrence** of the target in the array using binary search. If the target is not found, return -1. You are given a sorted array arr[] and a target element target. Your task is to find the **first occurrence** of the target in the array using binary search. If the target is not found, return -1.

**Home Task**

1. Write a program initializing array of size 20 and search an element using binary search.
2. Write a function called occurrences that, given an array of numbers A, prints all the distinct values in A each followed by its number of occurrences.

For example, if A = (28, 1, 0, 1, 0, 3, 4, 0, 0, 3), the function should output the following five lines (here separated by a semicolon) “28 1; 1 2; 0 4; 3 2; 4 1”.

1. Assume a bank's system needs to identify accounts with critically low balances and alert the user. Test the function with various balance values to ensure it correctly identifies all accounts below the threshold.