

**Name: Farhan Ahmad**

**SAP ID: 56193**

**Lab Task : 07**

**DSA LAB**

### **Question 1**

```
#include<iostream>
```

```
Using namespace std;
```

```
Int binarySearch(int arr[], int size, int target){
```

```
    Int lb = 0;
```

```
    Int ub = size-1;
```

```
    Cout << “\nActive elements at beginning : \n”;
```

```
    For(int l = 0 ; i<size ; i++){
```

```
        Cout << arr[i] << “\t”;
```

```
    }
```

```
    While(lb<=ub){
```

```
        Int mid = (lb+ub)/2;
```

```
        If(target == arr[mid]){
```

```
            Return mid;
```

```
        }
```

```
        Else if(target < arr[mid]){
```

```
            Ub = mid-1;
```

```

        Cout << "\nActive elements : \n";

        For(int l = lb; i<=ub; i++){

            Cout << arr[i] << "\t";

        }

    }

Else if(target > arr[mid]){

    Lb= mid+1;

    Cout << "\nActive elements : \n";

    For(int l = lb; i<=ub; i++){

        Cout << arr[i] << "\t";

    }

}

}

Return -1;

}

Int main(){

    Int size;

    Cout << "Enter size of array : ";

    Cin>> size;

```

```
Int *arr = new int(size);
```

```
For(int i=0; i<size; i++){
```

```
    Cout<< "Enter element for index " << i << " : ";
```

```
    Cin>> arr[i];
```

```
}
```

```
Int target;
```

```
Cout<< "Enter element you want to search : ";
```

```
Cin>> target;
```

```
Int result = binarySearch(arr, size, target);
```

```
If(result != -1){
```

```
    Cout << "Number if found at index " << result << endl ;
```

```
}
```

```
Else
```

```
    Cout << "Number is not found.\n";
```

```
}
```

## **Question 2**

```
#include<iostream>
```

```
Using namespace std;
```

```
Int binarySearch(int arr[], int size, int target){
```

```
    Int lb = 0;
```

```
    Int ub = size-1;
```

```
    Int result = -1;
```

```
    While(lb<=ub){
```

```
        Int mid = (lb+ub)/2;
```

```
        If(target == arr[mid]){
```

```
            Result = mid;
```

```
            Ub = mid-1;
```

```
        }
```

```
        Else if(target < arr[mid]){
```

```
            Ub = mid-1;
```

```
        }
```

```
        Else if(target > arr[mid]){
```

```
            Lb= mid+1;
```

```
        }
```

```
    }
```

```
    Return result;
```

```
}
```

```

Int main(){

    Int size;

    Cout << "Enter size of array : ";
    Cin >> size;

    Int *arr = new int(size);

    Cout << "Enter elements in ascending order.\n";
    For(int i=0; i<size; i++){
        Cout << "Enter element for index " << i << " : ";
        Cin >> arr[i];
    }

    Int target;

    Cout << "Enter element you want to search : ";
    Cin >> target;

    Int result = binarySearch(arr, size, target);

    If(result != -1){
        Cout << "Number if found at index " << result << endl ;
    }
}

```

```
Else
Cout << "Number is not found.\n";
}
```

### **Question 3**

```
#include<iostream>
```

```
Using namespace std;
```

```
Int binarySearch(int arr[], int size, int target){
```

```
    Int lb = 0;
```

```
    Int ub = size-1;
```

```
    Int result = -1;
```

```
    While(lb<=ub){
```

```
        Int mid = (lb+ub)/2;
```

```
        If(target == arr[mid]){
```

```
            Result = mid;
```

```
            Lb= mid+1;
```

```
        }
```

```
        Else if(target < arr[mid]){
```

```
            Ub = mid-1;
```

```
        }
```

```
        Else if(target > arr[mid]){
```

```
        Lb= mid+1;
    }
}
```

```
Return result;
```

```
}
```

```
Int main(){
```

```
    Int size;
```

```
    Cout << "Enter size of array : ";
```

```
    Cin >> size;
```

```
    Int *arr = new int(size);
```

```
    Cout << "Enter elements in ascending order.\n";
```

```
    For(int i=0; i<size; i++){
```

```
        Cout << "Enter element for index " << i << " : ";
```

```
        Cin >> arr[i];
```

```
    }
```

```
    Int target;
```

```
    Cout << "Enter element you want to search : ";
```

```

Cin>> target;

Int result = binarySearch(arr, size, target);

If(result != -1){
    Cout << "Number if found at index " << result << endl ;
}
Else
Cout << "Number is not found.\n";
}

```

#### **Question 4**

```

#include <iostream>

Using namespace std;

Int binarySearch(int arr[], int size, int target) {
    Int lb = 0;
    Int ub = size - 1;

    Int count = 0;

    While (lb <= ub) {
        Int mid = (lb + ub) / 2;

```



```

If (target == arr[mid]) {
    Count++;

    Int left = mid - 1;
    While (left >= lb && arr[left] == target) {
        Count++;
        Left--;
    }

    Int right = mid + 1;
    While (right <= ub && arr[right] == target) {
        Count++;
        Right++;
    }
Return count;
}

Else if (target < arr[mid]) {
    Ub = mid - 1; // Search left
}

Else {
    Lb = mid + 1; // Search right
}
}

```

```
    Return count;
}
```

```
Int main() {
```

```
    Int size;
```

```
    Cout << "Enter size of array: ";
```

```
    Cin >> size;
```

```
    Int *arr = new int[size]; // Allocate memory for an array
```

```
    Cout << "Enter elements in ascending order.\n";
```

```
    For (int I = 0; I < size; i++) {
```

```
        Cout << "Enter element for index " << I << ": ";
```

```
        Cin >> arr[i];
```

```
    }
```

```
    Int target;
```

```
    Cout << "Enter element you want to search: ";
```

```
    Cin >> target;
```

```
    Int count = binarySearch(arr, size, target); // Pass count by reference
```

```

If (count > 0) {
    Cout << "Number of occurrences = " << count << endl;
}
Else {
    Cout << "Number is not found.\n";
}

Delete[] arr; // Free the allocated memory

Return 0;
}

```

## **Sorting**

/ First sort the unsorted array

```
#include <iostream>
```

```
Using namespace std;
```

```

Void sort(int arr[], int size) {
    For (int i = 0; i < size-1; i++) {
        For (int j = 0; j < size-i-1; j++) {
            If (arr[j] > arr[j + 1]) {
                Int temp = arr[j];
                Arr[j] = arr[j + 1];

```

```

        Arr[j + 1] = temp;
    }
}
}
}

```

```

Void printArray(int arr[], int size) {
    For (int i = 0; i < size; i++) {
        Cout << arr[i] << " ";
    }
    Cout << endl;
}

```

```

Int main() {

```

```

    Int arr[] = {64, 25, 12, 22, 11, 12};

    Int size = sizeof(arr) / sizeof(arr[0]);

    Cout << size << endl;

    Cout << sizeof(arr) << endl;

    Cout << sizeof(arr[0]) << endl;

```

```
Cout << "Unsorted array: ";
```

```
printArray(arr, size);
```

```
sort(arr, size);
```

```
cout << "Sorted array: ";
```

```
printArray(arr, size);
```

```
return 0;
```

```
}
```