Program: Linear and Binary Search

LINEAR

```
#include <stdio.h>

void linearSearch(int values[], int target, int length);

int main(){
    int arr[5],length, target;
    printf("Enter The Number of Values: ");
    scanf("%d", &length);
    printf("Enter The %d Values: \n", length);
    for (int i = 0; i < length; i++){
        scanf("%d", &arr[i]);
    }
    printf("Enter The Value to be Searched: ");
    scanf("%d", &target);
    linearSearch(arr, target, length);
}

void linearSearch(int values[], int target, int length){
    for (int i = 0; i < length; i++){
        if (values[i] == target){
            printf("Element Found at index %d  ", i);
            return;
        }
    }
    printf("Elememt not Found");
}
```

OUTPUT:

```
root@kali:Data-Structures/expt6 - linear and binary # ./a.out
Enter The Number of Values: 5
Enter The 5 Values:
10 15 100 2 40
Enter The Value to be Searched: 2
Element Found at index 3
```

BINARY

```
#include <stdio.h>
void binarySearch(int values[], int target, int length);
int main()
{
    int arr[5];
    int length, target;
    printf("Enter The Number of Values: ");
    scanf("%d", &length);

    printf("Enter The %d Values: \n", length);
    for (int i = 0; i < length; i++)
    {
        scanf("%d", &arr[i]);
    }

    printf("Enter The Value to be Searched: ");
    scanf("%d", &target);

    binarySearch(arr, target, length);
    return 0;
}

void binarySearch(int values[], int target, int length)
{
    int low, high, mid;
    low = 0;
    high = length - 1;
    mid = (low + high) / 2;

    while (high >= low)
    {
        if (values[mid] < target)
        {
```

```
            low = mid + 1;
        }
        else if (target == values[mid])
        {
            printf("Element Found at index %d  ", mid);
            return;
        }
        else
        {
            high = mid - 1;
        }
        mid = (high + low) / 2;
    }
    if (low > high)
    {
        printf("Elememt not Found");
    }
}
```

OUTPUT:

```
root@kali:Data-Structures/expt6 - linear and binary # ./a.out
Enter The Number of Values: 5
Enter The 5 Values:
10 15 20 25 30
Enter The Value to be Searched: 30
Element Found at index 4
```