Folder SEM 3\Exp7

2 printable files

(file list disabled)

SEM 3\Exp7\Bin_Search.c

```c
#include <stdio.h>

// Function to perform binary search
int binarySearch(int arr[], int size, int key)
{
    int left = 0;
    int right = size - 1;

    while (left ⩽ right)
    {
        int mid = left + (right - left) / 2;

        if (arr[mid] == key)
        {
            return mid; // Return the index of the found element
        }
        if (arr[mid] < key)
        {
            left = mid + 1; // Search in the right half
        }
        else
        {
            right = mid - 1; // Search in the left half
        }
    }
    return -1; // Return -1 if the element is not found
}

int main()
{
    int arr[] = {1, 2, 5, 5, 6, 9}; // Note: Array must be sorted for binary search
    int size = sizeof(arr) / sizeof(arr[0]);
    int key;

    printf("Enter the element to search for (Binary Search): ");
    scanf("%d", &key);

    int index = binarySearch(arr, size, key);
    if (index ≠ -1)
    {
        printf("Element %d found at index %d.\n", key, index);
    }
    else
    {
        printf("Element %d not found in the array.\n", key);
```

```
    }

    return 0;
}

```

SEM 3\Exp7\Linear_Search.c

```c
#include <stdio.h>

// Function to perform linear search
int linearSearch(int arr[], int size, int key)
{
    for (int i = 0; i < size; i++)
    {
        if (arr[i] == key)
        {
            return i; // Return the index of the found element
        }
    }
    return -1; // Return -1 if the element is not found
}

int main()
{
    int arr[] = {5, 2, 9, 1, 5, 6};
    int size = sizeof(arr) / sizeof(arr[0]);
    int key;

    printf("Enter the element to search for (Linear Search): ");
    scanf("%d", &key);

    int index = linearSearch(arr, size, key);
    if (index != -1)
    {
        printf("Element %d found at index %d.\n", key, index);
    }
    else
    {
        printf("Element %d not found in the array.\n", key);
    }

    return 0;
}

```