11 - Searching - Bin & linear

Aim

Show Searching using Linear & Binary Search

Algorithm

I. void showarr (int arr[])

1. START
2. for (int i=0 to len (arr))
   1. printf ("%d \t", arr[i])
3. STOP

II. void linearSearch (int arr[], int key)

1. START
2. ~~printf ("\n~~ for (int i=0 to len (arr))
   1. if (arr[i] == key)
      1. break;
   2. End If
3. End for.
4. if (key == arr[i])
   1. print ("key found");

5. Else
   1. print ("Key not found")
6. End If
7. STOP

III int compfunc (const void *a, const void *b)

1. START
2. return (*(int *) a > *(int *) b);
3. STOP

IV void binSearch (int arr[], int key)

1. START
2. qsort (arr, key(arr), sizeof(int, cmpfunc)
3. showarr(arr),
4. int low = 0, high = n-1 , mid = 0;
5. while (key != arr[mid] && low < high) then
   1. if (key < arr[mid])
      1. high = mid - 1
   2. Else
      1. low = mid + 1
   3. mid = (low + high)/2;
6. End while

7. if (key == arr[mid])
    1. print ("key found");
8. Else
    1. print ("key not found");
9. End if.
10. STOP

V int main()

1. Start
2. Input an array values from user
3. Input the value to search for
4. ~~call~~ linearSearch (arr, key)
5. binSearch (arr, key);
6. STOP

Output

Output obtained & verified.

```
#include <stdio.h>
#include <stdlib.h>

#define n 4
void showarr(int arr[]){
    printf("\n");
    for(int i=0 ;i<n ;i++){
        printf("%d\t",arr[i]);
    }
}
void linearSearch(int arr[], int key){
    printf("\n********Linear Search*********");
    showarr(arr);
        int i;
        for(i=0; i<n; i++){
                if(arr[i]==key)
                        break;
        }
        if(key==arr[i])
                printf("\n-- key is found at a[%d] in unsorted array--\n", i);
        else
                printf("\n-- key not found --\n");
}

int cmpfunc (const void *a, const void *b) {
   return (*(int*)a > *(int*)b);
}
/*void sort(int arr[]){
        int tmp;
        for(int i=0 ;i<n-1 ;i++){
                for(int j=0 ;j<n-i-1 ;j++){
                        if(arr[j]>arr[j+1]){
                                tmp=arr[j];
                                arr[j]=arr[j+1];
                                arr[j+1]=tmp;
                        }
                }
        }
}*/
void binSerach(int arr[], int key){
    printf("\n********Binary Search*********");
        qsort(arr,n,sizeof(int),cmpfunc);
        //sort(arr);
        showarr(arr);

        int low=0 ,high=n-1 ,mid=0 ;
        while(key!=arr[mid] && low<high){
                if(key<arr[mid])
                        high=mid-1;
                else
```

```
                        low=mid+1;
                mid=(low+high)/2;
        }
        if(key==arr[mid])
                printf("\n-- key is found at a[%d] in sorted array--", mid);
        else
                printf("\n-- key not found --");
}

int main(){


        int arr[n], s;
        printf("Enter the array elements -->");
        for(int i=0 ;i<n ;i++){
                scanf("%d",&arr[i]);
        }

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

        linearSearch(arr,s);
        binSerach(arr,s);
}
```

```
Enter the array elements -->6
4
8
2
Enter the element to search for:2

********Linear Search*********
6    4    8    2
-- key is found at a[3] in unsorted array--

********Binary Search*********
2    4    6    8
-- key is found at a[0] in sorted array-->
```