//Linear search using function

#include <stdio.h>

#include<string.h>

int linearsearch(int arr1[],int size,int search);

int linearsearch(int arr1[],int size,int search)

{

int i;

for(i=0;i<size;i++)

{

if(arr1[i]==search)

{

return i;

}

}

return -1;

}

void main()

{

int arr[100];

int n,i,search;

printf("Enter array size:");

scanf("%d",&n);

printf("\nEnter array elements:");

for(i=0;i<n;i++)

{

scanf("%d",&arr[i]);

}

printf("Enter element you ant to search");

scanf("%d",&search);

int res=linearsearch(arr,n,search);

if(res==-1)

{

printf("Element not found");

}

else

{

printf("Element found at %d index", res);

}

}

///Linear search using recursion

#include <stdio.h>

#include<string.h>

int linearsearch(int arr1[],int size,int search);

int linearsearch(int arr1[],int size,int search)

{

int i;

if(size==0)

{

return -1;

}

if(arr1[size-1]==search)

{

return size-1;

}

return linearsearch(arr1,size-1,search);

}

void main()

{

int arr[100];

int n,i,search;

printf("Enter array size:");

scanf("%d",&n);

printf("\nEnter array elements:");

for(i=0;i<n;i++)

{

scanf("%d",&arr[i]);

}

printf("Enter element you ant to search");

scanf("%d",&search);

int res=linearsearch(arr,n,search);

if(res==-1)

{

printf("Element not found");

}

else

{

printf("Element found at %d index", res);

}

}

//Binary Search using function

#include <stdio.h>

#include<string.h>

int binary\_search(int\* arr,int start,int end,int search)

{

int mid,i;

while(start<=end)

{

mid=(start+end)/2;

if(search==arr[mid])

{

return mid;

}

if(search<arr[mid])

{

end=mid-1;

}

else

{

start=mid+1;

}

}

return -1;

}

int main()

{

int arr[100];

int n,i,j,search,result,temp;

printf("Enter array size:");

scanf("%d",&n);

printf("\nEnter array elements in sorted order:");

for(i=0;i<n;i++)

{

scanf("%d",&arr[i]);

}

printf("Enter element which you want to search:");

scanf("%d",&search);

// for(i=0;i<n;i++)

// {

// for(j=i+1;j<n;j++)

// {

// if(arr[i]>arr[j])

// {

// temp=arr[i];

// arr[i]=arr[j];

// arr[j]=temp;

// }

// }

// }

result=binary\_search(arr,0,n-1,search);

if(result==-1)

{

printf("Element not found");

}

else

{

printf("Element found at %d",result);

}

}

//Binary using recursion

#include <stdio.h>

#include<string.h>

int binary\_search(int\* arr,int start,int end,int search)

{

int mid,i;

while(start<=end)

{

mid=(start+end)/2;

if(search==arr[mid])

{

return mid;

}

if(search<arr[mid])

{

return binary\_search(arr,start,mid-1,search);

}

else

{

return binary\_search(arr,mid+1,end,search);

}

}

return -1;

}

int main()

{

int arr[100];

int n,i,j,search,result,temp;

printf("Enter array size:");

scanf("%d",&n);

printf("\nEnter array elements in sorted order:");

for(i=0;i<n;i++)

{

scanf("%d",&arr[i]);

}

printf("Enter element which you want to search:");

scanf("%d",&search);

// for(i=0;i<n;i++)

// {

// for(j=i+1;j<n;j++)

// {

// if(arr[i]>arr[j])

// {

// temp=arr[i];

// arr[i]=arr[j];

// arr[j]=temp;

// }

// }

// }

result=binary\_search(arr,0,n-1,search);

if(result==-1)

{

printf("Element not found");

}

else

{

printf("Element found at %d",result);

}

}

// Display the word at given location in sentence

#include <stdio.h>

#include<string.h>

void main()

{

char str[50];

int i,j,k=0,l=0,len,pos,count=0;

printf("Enter the string:");

fgets(str,sizeof(str),stdin);

len=strlen(str);

printf("Enter position:");

scanf("%d",&pos);

for(i=0;i<len-1;i++)

{

if(str[i]==' ')

{

count++;

if(count==pos-1){

k=i;

break;

}

}

}

for(j=k+1;str[j]!=' ';j++)

{

l=j;

}

for(i=k;i<=l;i++)

{

printf("%c",str[i]);

}

}

//Bubble sort

#include <stdio.h>

#include<string.h>

void main()

{

int arr[100],i,j,n,temp,swap=0;

printf("Enter the no of elements:");

scanf("%d",&n);

printf("\nEnter the elements:");

for(i=1;i<n;i++)

{

scanf("%d",&arr[i]);

}

for(i=1;i<n;i++)

{

for(j=0;j<n-i;j++)

{

if(arr[j]>arr[j+1])

{

temp=arr[j];

arr[j]=arr[j+1];

arr[j+1]=temp;

swap=1;

}

}

if(swap==0)

{

break;

}

}

printf("\nAfter sorting array is:");

for(i=0;i<n;i++)

{

printf("%d\t",arr[i]);

}

}

//Bubble sort using DMA and function

#include <stdio.h>

#include<stdlib.h>

void bubblesort(int[],int);

void bubblesort(int arr[],int n)

{

int i,j,temp,swap=0;

for(i=1;i<n;i++)

{

for(j=0;j<n-i;j++)

{

if(arr[j]>arr[j+1])

{

temp=arr[j];

arr[j]=arr[j+1];

arr[j+1]=temp;

swap=1;

}

}

if(swap==0)

{

break;

}

}

}

void main()

{

int arr[100],i,n,\*ptr;

printf("Enter the no of elements:");

scanf("%d",&n);

ptr=(int\*) malloc(n\*sizeof(int));

printf("\nEnter the elements:");

for(i=1;i<n;i++)

{

scanf("%d",(ptr+i));

}

bubblesort(ptr,n);

printf("\nAfter sorting array is:");

for(i=0;i<n;i++)

{

printf("%d\t",\*(ptr+i));

}

}

//Selection sort

#include <stdio.h>

#include<stdlib.h>

void selectionsort(int[],int);

void swap(int\* a,int\* b)

{

int temp=\*a;

\*a=\*b;

\*b=temp;

}

void selectionsort(int arr[],int n)

{

int i,j;

for(i=0;i<n-1;i++)

{

int min=i;

for(j=i+1;j<n;j++)

{

if(arr[j]<arr[min])

{

min=j;

}

}

if(min!=i)

{

swap(&arr[i],&arr[min]);

}

}

}

void main()

{

int arr[100],i,n,\*ptr;

printf("Enter the no of elements:");

scanf("%d",&n);

ptr=(int\*) malloc(n\*sizeof(int));

printf("\nEnter the elements:");

for(i=1;i<n;i++)

{

scanf("%d",(ptr+i));

}

selectionsort(ptr,n);

printf("\nAfter sorting array is:");

for(i=0;i<n;i++)

{

printf("%d\t",\*(ptr+i));

}

}