**WEEK II**

1. **Given a sorted array of positive integers containing few duplicate elements, design an algorithm and implement it using a program to find whether the given key element is present in the array or not. If present, then also find the number of copies of given key. (Time Complexity = O(log n))**

**ANSWER I. PROGRAM**

|  |
| --- |
| #include<iostream> |
|  | using namespace std; |
|  | int Binary\_search\_Firstoccurence(int arr[],int n,int key) |
|  | { |
|  | int res=0; |
|  | int low=0,high=n-1; |
|  |  |
|  | while (low<=high) |
|  | { |
|  | int mid=low+(high-low)/2; |
|  | if (key==arr[mid]) |
|  | { |
|  | res=mid; |
|  | high=mid-1; |
|  | } |
|  | else if (key>arr[mid]) |
|  | low=mid+1; |
|  | else |
|  | high=mid-1; |
|  | } |
|  | return res; |
|  | } |
|  | int Binary\_search\_Lastoccurence(int arr[],int n,int key) |
|  | { |
|  | int flag=0; |
|  | int low=0,high=n-1; |
|  | int res=0; |
|  |  |
|  | while (low<=high) |
|  | { |
|  | int mid=low+(high-low)/2; |
|  | if (key==arr[mid]) |
|  | { |
|  | flag=1; |
|  | res=mid; |
|  | low=mid+1; |
|  | } |
|  | else if (key>arr[mid]) |
|  | low=mid+1; |
|  | else |
|  | high=mid-1; |
|  | } |
|  | if (flag==0) |
|  | return -1; |
|  | return res; |
|  | } |
|  | int main() |
|  | { |
|  | int t; |
|  | cin>>t; |
|  | while (t--) |
|  | { |
|  | int n; |
|  | cin>>n; |
|  | int arr[n]; |
|  | for (int i=0;i<n;i++) |
|  | cin>>arr[i]; |
|  | int key; |
|  | cin>>key; |
|  |  |
|  | int first=Binary\_search\_Firstoccurence(arr,n,key); |
|  | int last=Binary\_search\_Lastoccurence(arr,n,key); |
|  | if (last!=-1) |
|  | cout<<key<<"-"<<last-first+1<<endl; |
|  | else |
|  | cout<<"Key not present"<<endl; |
|  |  |
|  | } |
|  |  |
|  | } |

**OUTPUT**

**1**

**5**

**1 2 2 4 5**

**2**

**2-2**

1. **Given an array of nonnegative integers, design an algorithm and a program to count the number of pairs of integers such that their difference is equal to a given key, K.**

**ANSWER II. PROGRAM**

#include<stdio.h>

#define MAX 100

int main()

{

int flag=0;

int t ,A[MAX] ,n;

printf("Enter the test cases :");

scanf("%d",&t);

while(t)

{

int i=0 ,j=0 ,k=0;

printf("\nEnter the size of the array :");

scanf("%d",&n);

printf("Enter the elements in the array :");

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

{

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

}

for(i=0 ;i<n-2 ;i++)

{

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

{

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

{

if(A[i]+A[j]==A[k])

{

printf("\nIndices :%d\t%d\t%d",i ,j ,k);

flag=1;

break;

}

}

}

}

t=t-1;

}

if(flag==0)

{

printf("No sequence found");

}

return 0;

}

**OUTPUT**

**Enter the test cases :1**

**Enter the size of the array :5**

**Enter the elements in the array :1 2 3 4 5**

**Indices :0 1 2**

**Indices :0 2 3**

**Indices :0 3 4**

**Indices :1 2 4**

1. **Given an array of nonnegative integers, design an algorithm and a program to count the number of pairs of integers such that their difference is equal to a given key, K.**

**ANSWER III. PROGRAM**

#include<stdio.h>

#include<time.h>

#include<stdlib.h>

#define MAX 100

void sort(int A[],int n)

{

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

{

for(int j=0 ;j<n ;j++)

{

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

{

int temp=A[j];

A[j]=A[j+1];

A[j+1]=temp;

}

}

}

}

int count\_difference(int A[],int key,int n)

{

int count=0;

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

{

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

{

if(A[j]-A[i]==key)

{

count++;

}

}

}

return count;

}

int main()

{

int t;

int key;

int count=0;

int A[MAX],n;

printf("Enter the test case :");

scanf("%d",&t);

while(t--)

{

printf("Enter the size :");

scanf("%d",&n);

printf("Enter the elements in the array :");

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

{

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

}

printf("Enter the key value :");

scanf("%d",&key);

sort(A,n);

count=count\_difference(A,key ,n);

printf("\nTotal count %d",count);

}

if(!count)

{

printf("No pair found");

}

return 0;

}

**OUTPUT**

**Enter the test case :1**

**Enter the size :5**

**Enter the elements in the array :12 21 67 90 23**

**Enter the key value :23**

**Total count 1**