**Week 2:**

**Question 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))

#import<iostream>

using namespace std;

int binary(int arr[],int l,int r, int x)

{

if(r>=l){

int mid=l+(r-l)/2;

if(arr[mid]==x)

return mid;

if(arr[mid]>x)

return binary(arr,l,mid-1,x);

return binary(arr,mid+1,r,x);

}

return -1;

}

int main()

{

int T;

cout<<"enter the number of test cases"<<endl;

cin>>T;

while(T--)

{

int a[100],n,k;

int c=0;

cout<<"enter the number of elements"<<endl;

cin>>n;

cout<<"enter the elements"<<endl;

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

cin>>a[i];

cout<<"enter the element to be searched"<<endl;

cin>>k;

int f=binary(a,0,n-1,k);

if(f!=-1)

{

int j=f+1;

while(a[f--]==k)

{

c++;

}

while(a[j++]==k)

{

c++;

}

cout<<k<<"-"<<c<<endl;

}

else

cout<<"Key not found"<<endl;

}

return 0;

}

**Output:**

enter the number of test cases

3

enter the number of elements

10

enter the elements

235 235 278 278 763 764 790 853 981 981

enter the element to be searched

981

981-2

enter the number of elements

15

enter the elements

1 2 2 3 3 5 5 5 25 75 75 75 97 97 97

enter the element to be searched

75

75-3

enter the number of elements

5

enter the elements

1 2 2 3 4

enter the element to be searched

5

Key not found

Process returned 0 (0x0) execution time : 110.122 s

Press any key to continue.

**Question 2:** Given a sorted array of positive integers, design an algorithm and implement it using a program to find three indices i, j, k such that arr[i] + arr[j] = arr[k].

#import<iostream>

using namespace std;

int main()

{

int T;

cout<<"enter the number of test cases"<<endl;

cin>>T;

while(T--)

{

int a[100],n,i,j,k;

cout<<"enter the number of elements"<<endl;

cin>>n;

cout<<"enter the elements"<<endl;

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

cin>>a[i];

int flag=0;

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

{

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

{

int sum=a[i]+a[j];

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

{

if(sum==a[k])

{

flag=1;

break;

}

}

if(flag==1)

break;

}

if(flag==1)

break;

}

if(flag==1)

cout<<"value of i ,j and k: "<<i+1<<","<<j+1<<","<<k+1<<endl;

else

cout<<"No sequence found."<<endl;

}

return 0;

}

**Output:**

enter the number of test cases

3

enter the number of elements

5

enter the elements

1 5 84 209 341

No sequence found.

enter the number of elements

10

enter the elements

24 28 48 71 86 89 92 120 194 201

value of i ,j and k: 2,7,8

enter the number of elements

15

enter the elements

64 69 82 95 99 107 113 141 171 350 369 400 511 590 666

value of i ,j and k: 1,6,9

Process returned 0 (0x0) execution time : 97.041 s

Press any key to continue.

**Question 3:** 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.

#import<iostream>

#import<cstdlib>

using namespace std;

int main()

{

int T;

cout<<"enter the number of test cases"<<endl;

cin>>T;

while(T--)

{

int a[100],n,i,j;

int key;

cout<<"enter the number of elements"<<endl;

cin>>n;

cout<<"enter the elements"<<endl;

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

cin>>a[i];

cout<<"enter the key"<<endl;

cin>>key;

int c=0;

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

{

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

{

int diff=abs(a[i]-a[j]);

if(diff==key)

c++;

}

}

cout<<"number of such pairs:"<<c<<endl;

}

return 0;

}

**Output:**

enter the number of test cases

3

enter the number of elements

5

enter the elements

1 51 84 21 31

enter the key

20

number of such pairs:2

enter the number of elements

10

enter the elements

24 71 16 92 12 28 48 14 20 22

enter the key

4

number of such pairs:4

enter the number of elements

5

enter the elements

1 2 3 4 5

enter the key

5

number of such pairs:0

Process returned 0 (0x0) execution time : 65.493 s

Press any key to continue.