/\* Given an array of nonnegative integers, design a linear algorithm and implement it using a program to find whether given key element is present in the array or not. Also, find total number of comparisons for each input case. (Time Complexity = O(n), where n is the size of input)\*/

#include <stdio.h>

int main() {

int i,n,a[100],k,f=0,test,idx;

//Input number of times.

scanf("%d",&test);

//Loop for same operations

for(int i=1;i<=test;i++)

{

idx=0;

//Enter Array Size

scanf("%d",&n);

//Enter Array Elements

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

{

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

}

//Enter Element to find

scanf("%d",&k);

// Loop for comparison

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

{

idx++;

if(a[i]==k)

{

f=1;

break;

}

}

// Output

if(f==1)

{

printf("Present %d\n",idx);

}

else

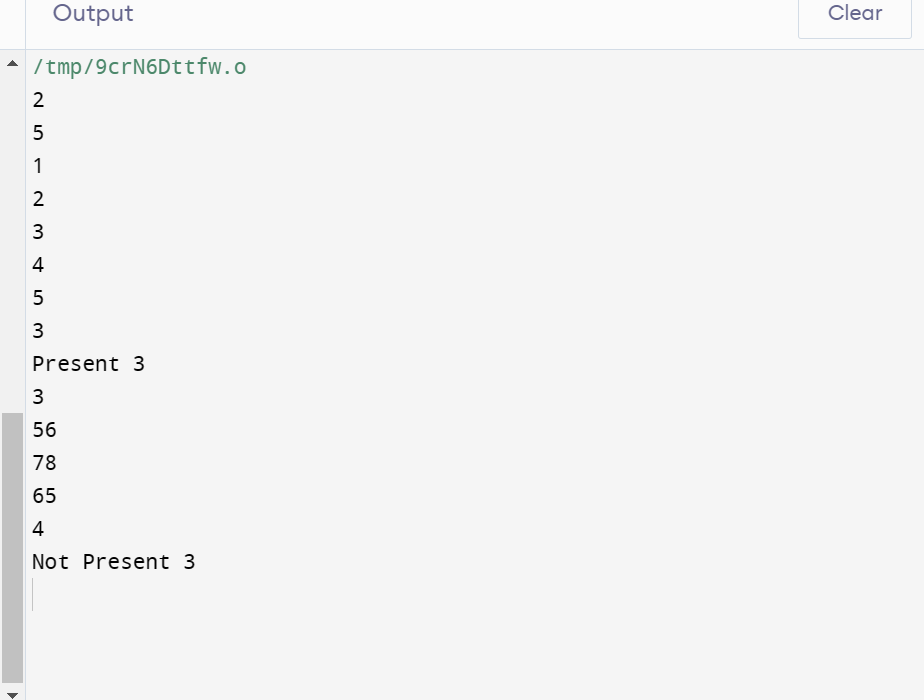
{

printf("Not Present %d\n",idx);

}

}

return 0;

}

Name -Shalvi jain Sec – A Roll\_No – 56

/\* Given an already sorted array of positive integers, design an algorithm and implement it using a program to find whether given key element is present in the array or not. Also, find total number of comparisons for each input case. (Time Complexity = O(nlogn), where n is the size of input). \*/

#include <stdio.h>

int main() {

int i,n,a[100],k,f=0,test,idx=0;

//Input number of times.

scanf("%d",&test);

//Loop for same operations

for(int i=1;i<=test;i++)

{

f=0;

idx=0;

//Enter Array Size

scanf("%d",&n);

//Enter Array Elements

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

{

scanf("%d",&a[j]);

}

//Enter Element to find

scanf("%d",&k);

// Binary Search for comparison

int low=0,high=n-1,mid;

do

{

idx++;

mid = (low+high)/2;

if(k==a[mid])

{

f = 1;

break;

}

(k > a[mid]) ? (low = mid + 1) : (high = mid - 1);

}while(low<=high);

if(f==1)

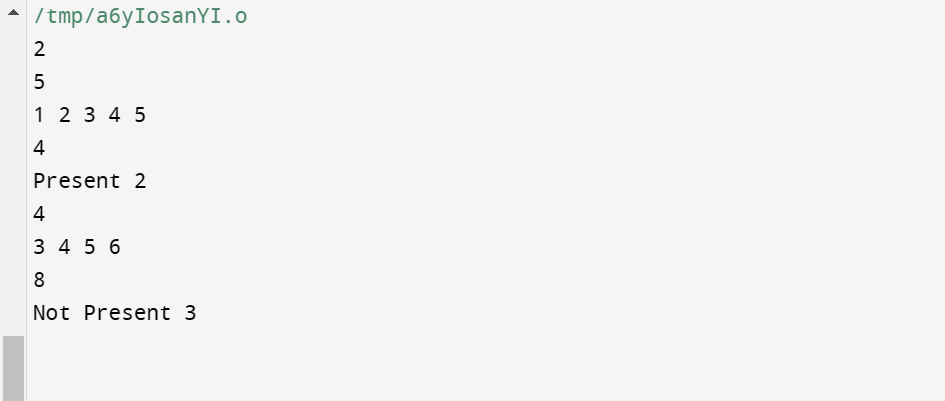
printf("Present %d\n",idx);

else

printf("Not Present %d\n",idx);

}

return 0;

}~~~~

Name -Shalvi jain Sec – A Roll\_No – 56

/\*Given an already sorted array of positive integers, design an algorithm and implement it using a program to find whether a given key element is present in the sorted array or not. For an array a[n], search at the indexes a[0], a[2], a[4],.....,a[2k] and so on. Once the interval (a[2k] < key < a[ 2k+1] ) is found, perform a linear search operation from the index 2k to find the element key. (Complexity < O(n), where n is the number of elements need to be scanned for searching): \*/

#include<stdio.h>

#include<math.h>

int main()

{

int i,n,a[100],k,f,test,idx;

//Input number of times.

scanf("%d",&test);

//Loop for same operations

while(test)

{

f=0;

idx=0;

//Enter Array Size

scanf("%d",&n);

//Enter Array Elements

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

{

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

}

//Enter Element to find

scanf("%d",&k);

// Jump Search

int jump=sqrt(n),low = 0, high = jump;

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

{

idx++;

if(a[i]==k)

{

f=1;

break;

}

if(a[i]<k)

{

low = i;

}

if(a[i]>k)

{

break;

}

}

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

{

if(a[i]==k)

f=1;

}

// Output

if(f==1)

{

printf("Present %d\n",idx);

}

else

{

printf("Not Present %d\n",idx);

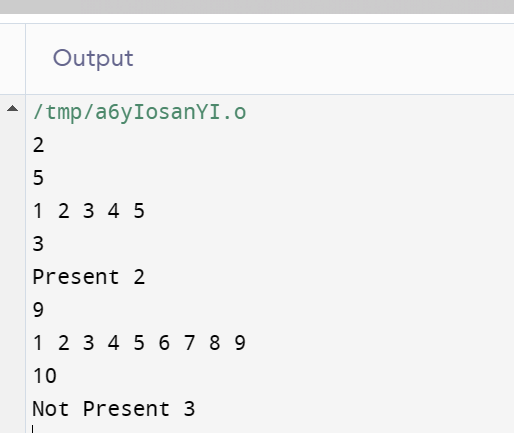
}

test--;

}

return 0;

}



Name -Shalvi jain Sec – A Roll\_No – 56