DAA MOODLE PROGRAMS COMPETITIVE PROGRAMS

230701221

M.Nithyashree

CSE-D

1.

AIM-

Find Duplicate in Array.

Given a read only array of n integers between 1 and n, find one number that repeats.

CODE-

```
#include<stdio.h>
int main()
          int n;
scanf("%d",&n);
int a[n];
for(int i=0;i<n;i++)</pre>
 4
 5
6
                scanf("%d",&a[i]);
10
11
           for(int i=0;i<n;i++)
12 v
                for(int j=i+1;j<n;j++)</pre>
13
14 v
15
                     if(a[i]==a[j])
16 v
17
                           printf("%d",a[i]);
18
19
20
21
22
23
24
```

INPUT-

First Line - Number of elements

n Lines - n Elements

OUTPUT-

Element x - That is repeated



AIM-

Find Duplicate in Array.

Given a read only array of n integers between 1 and n, find one number that repeats.

CODE-

```
1 #include<stdio.h>
 3 v int main() {
         int n;
scanf("%d",&n);
 5
         int a[n];
for(int i=0;i<n;i++) {
    scanf("%d",&a[i]);
 6
 7 *
 8
9
          int index=0;
10
           for(int i=0;i<n;i++) {</pre>
11 v
               index = a[i] % n;
a[index] += n;
12
13
14
           for(int i=0;i<n;i++) {
    if(a[i]/n >= 2) {
        printf("%d\n", i);
    }
}
15 v
16 •
17
18
19
           return 0;
20
21
22
```

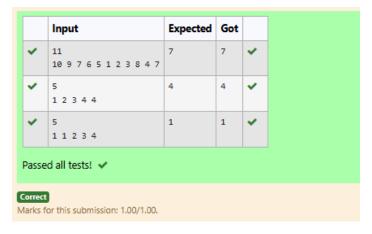
INPUT-

First Line - Number of elements

n Lines - n Elements

OUTPUT-

Element x - That is repeated



AIM-

Find the intersection of two sorted arrays.

OR in other words,

Given 2 sorted arrays, find all the elements which occur in both the arrays.

CODE-

```
int intersection(int arr1[],int n1,int arr2[],int n2)
{
          int i=0,j=0;
while(i<n1&&j<n2)</pre>
               if(arr1[i]==arr2[j])
{
                    printf("%d ",arr1[i]);
i++;
j++;
lse if(arr1[i]<arr2[j])
               els.
{
i++;
                    j++;
               }
          }
return 0;
    int main()
{
          int t;
scanf("%d",&t);
while(t--)
{
              int n1;
scanf("%d",&n1);
int arr1[n1];
for(int i=0;i<n1;i++)</pre>
                {
scanf("%d",&arr1[i]);
               {
{
scanf("%d",&arr2[i]);
               }
intersection(arr1,n1,arr2,n2);
          }
return 0;
```

INPUT-

The first line contains T, the number of test cases. Following T lines contain:

- 1. Line 1 contains N1, followed by N1 integers of the first array
- 2. Line 2 contains N2, followed by N2 integers of the second array

OUTPUT-

The intersection of the arrays in a single line

	Input	Expected	Got		
~	1 3 10 17 57 6 2 7 10 15 57 246	10 57	10 57	~	
~	1 6 1 2 3 4 5 6 2 1 6	1 6	16	~	
Passe	d all tests! 🗸				
Correct Varks for this submission: 1.00/1.00.					

AIM-

Find the intersection of two sorted arrays.

OR in other words, Given 2 sorted arrays, find all the elements which occur in both the arrays.

CODE-

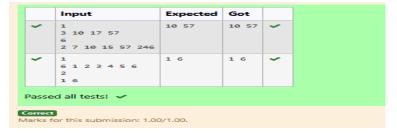
```
#include <stdio.h>
        void findintersection(int arr1[], int n1, int arr2[], int n2)
  4₩ {
               int i = 0, j = 0;
while (i < n1 && j < n2)
{</pre>
                      if (arr1[i] == arr2[j])
                             printf("%d ", arr1[i]);
i++;
j++;
12
13
14
15
16
                        lse if (arr1[i] < arr2[j])
                             i++;
17
18
19
20
21
22
23
24
25
26
27
28
               printf("\n");
       }
        int main()
              int t;
scanf("%d", &t);
while (t--) {
   int n1, n2;
   scanf("%d", &n1);
   int arr1[n1];
   for (int i = 0; i < n1; i++)
   {
        --f("%d", &arr1[i]);
}</pre>
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
                      }
scanf("%d", &n2);
int arr2[n2];
for (int i = 0; i < n2; i++)</pre>
                              scanf("%d", &arr2[i]);
                      findintersection(arr1, n1, arr2, n2);
```

INPUT-

- The first line contains T, the number of test cases. Following T lines contain:
- 1. Line 1 contains N1, followed by N1 integers of the first array
- 2. Line 2 contains N2, followed by N2 integers of the second array

OUTPUT-

The intersection of the arrays in a single line



AIM-

Given an array A of sorted integers and another non negative integer k, find if there exists 2 indices i and j such that A[j] - A[i] = k, i ! = j.

CODE-

```
#include<stdio.h>
int main()
 2
3 v
          int n;
scanf("%d",&n);
 4
 5
          int arr[n];
for(int i=0;i<n;i++)</pre>
 6
 7
 8 ,
                scanf("%d",&arr[i]);
 9
10
          int k;
scanf("%d",&k);
for(int i=0;i<n;i++)</pre>
11
12
13
14 🔻
                for(int j=i+1;j<n;j++)</pre>
15
16 •
                     if(arr[j]-arr[i]==k)
17
18 🔻
                          printf("1\n");
19
20
                          return 0;
21
22
23
          printf("0\n");
24
25
```

INPUT-

First Line n - Number of elements in an array

Next n Lines - N elements in the array

k - Non - Negative Integer

OUTPUT-

- 1 If pair exists
- 0 If no pair exists

	Input	Expected	Got		
~	3 1 3 5 4	1	1	~	
~	10 1 4 6 8 12 14 15 20 21 25 1	1	1	~	
~	10 1 2 3 5 11 14 16 24 28 29 0	0	0	~	
~	10 0 2 3 7 13 14 15 20 24 25 10	1	1	~	
Passed all tests! 🗸					
orrect arks for this submission: 1.00/1.00.					

AIM-

Given an array A of sorted integers and another non negative integer k, find if there exists 2 indices i and j such that A[j] - A[i] = k, i! = j.

CODE-

INPUT-

First Line n - Number of elements in an array

Next n Lines - N elements in the array

k - Non - Negative Integer

OUTPUT-

- 1 If pair exists
- 0 If no pair exists

