

Find Duplicate in Array.

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

Input Format:

First Line - Number of elements

n Lines - n Elements

Output Format:

Element x - That is repeated

For example:

Input	Result
5	1
1 1 2 3 4	

Answer: (penalty regime: 0 %)

```

1 #include<stdio.h>
2 int main(){
3     int n,repeat;
4     scanf("%d",&n);
5     int a[n];
6     for(int i=0;i<n;i++){
7         scanf("%d",&a[i]);
8     }
9     for(int i=0;i<n;i++){
10        for(int j=i+1;j<n;j++){
11            if(a[i]==a[j]){
12                repeat=a[i];
13            }
14        }
15    }
16    printf("%d",repeat);
17 }
```

	Input	Expected	Got	
✓	11 10 9 7 6 5 1 2 3 8 4 7	7	7	✓
✓	5 1 2 3 4 4	4	4	✓
✓	5 1 1 2 3 4	1	1	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Find Duplicate in Array.

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

Input Format:

First Line - Number of elements

n Lines - n Elements

Output Format:

Element x - That is repeated

For example:

Input	Result
5	1
1 1 2 3 4	

Answer: (penalty regime: 0 %)

```

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

	Input	Expected	Got	
✓	11 10 9 7 6 5 1 2 3 8 4 7	7	7	✓
✓	5 1 2 3 4 4	4	4	✓
✓	5 1 1 2 3 4	1	1	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

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.

Input Format

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

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

Output Format

The intersection of the arrays in a single line

Example

Input:

```
1
3 10 17 57
6 2 7 10 15 57 246
```

Output:

```
10 57
```

Input:

```
1
6 1 2 3 4 5 6
2 1 6
```

Output:

```
1 6
```

For example:

Input	Result
1	10 57
3 10 17 57	
6	
2 7 10 15 57 246	

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
2 int main(){
3     int t;
4     scanf("%d",&t);
5     for(int s=0;s<t;s++){
6         int n,m;
7         scanf("%d",&n);
8         int a[n];
9         for(int i=0;i<n;i++){
10             scanf("%d",&a[i]);
11         }
12         scanf("%d",&m);
13         int b[m];
14         for(int i=0;i<m;i++){
15             scanf("%d",&b[i]);
16         }
17         for(int i=0;i<n;i++){
18             for(int j=0;j<m;j++){
19                 if(a[i]==b[j]){
20                     printf("%d ", a[i]);
21                 }
22             }
23         }
24     }
25 }
```

```
20     printf("%d ",a[i]);
21 }
22 }
23 }
24 }
25 }
```

	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	1 6 ✓	

Passed all tests! ✓



Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)

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.

Input Format

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

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

Output Format

The intersection of the arrays in a single line

Example

Input:

```
1
3 10 17 57
6 2 7 10 15 57 246
```

Output:

```
10 57
```

Input:

```
1
6 1 2 3 4 5 6
2 1 6
```

Output:

```
1 6
```

For example:

Input	Result
1	10 57
3 10 17 57	
6	
2 7 10 15 57 246	

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>
2
3 int main() {
4     int T;
5     scanf("%d", &T);
6
7     while (T--) {
8         int n1, n2;
9         scanf("%d", &n1);
10        int a[n1];
11        for (int i = 0; i < n1; i++)
12            scanf("%d", &a[i]);
13
14        scanf("%d", &n2);
15        int b[n2];
16        for (int i = 0; i < n2; i++)
17            scanf("%d", &b[i]);
18
19        int i = 0, j = 0;
```

```

20
21
22 v     int found = 0;
23 v
24 v         while (i < n1 && j < n2) {
25 v             if (a[i] == b[j]) {
26 v                 printf("%d ", a[i]);
27 v                 found = 1;
28 v                 i++;
29 v                 j++;
30 v             }
31 v             else if (a[i] < b[j])
32 v                 i++;
33 v             else
34 v                 j++;
35 v         }
36 v
37 v         if (!found)
38 v             printf("No Common Elements");
39 v
40 v         printf("\n");
41 v
42 v     }
43 v

```

	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	1 6	✓

Passed all tests! ✓



Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)

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

Input Format:

First Line n - Number of elements in an array

Next n Lines - N elements in the array

k - Non - Negative Integer

Output Format:

1 - If pair exists

0 - If no pair exists

Explanation for the given Sample Testcase:

YES as $5 - 1 = 4$

So Return 1.

For example:

Input	Result
3	1
1 3 5	
4	

Answer: (penalty regime: 0 %)

```

1 #include<stdio.h>
2 int main(){
3     int n,flag=0;
4     scanf("%d",&n);
5     int a[n];
6     for(int i=0;i<n;i++){
7         scanf("%d",&a[i]);
8     }
9     int k,d;
10    scanf("%d",&k);
11    for(int i=0;i<n;i++){
12        for(int j=i+1;j<n;j++){
13            d=a[j]-a[i];
14            if(d==k){
15                flag=1;
16            }
17        }
18    }
19    if(flag==1){
20        printf("1");
21    }
22    else{
23        printf("0");
24    }
25}
26 }
```

	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! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)

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

Input Format:

First Line n - Number of elements in an array

Next n Lines - N elements in the array

k - Non - Negative Integer

Output Format:

1 - If pair exists

0 - If no pair exists

Explanation for the given Sample Testcase:

YES as $5 - 1 = 4$

So Return 1.

For example:

Input	Result
3	1
1 3 5	
4	

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2 int main() {
3     int n;
4     scanf("%d", &n);
5
6     int arr[n];
7     for (int i = 0; i < n; i++)
8         scanf("%d", &arr[i]);
9
10    int k;
11    scanf("%d", &k);
12
13    int i = 0, j = 1;
14    int found = 0;
15
16    while (i < n && j < n) {
17        if (i != j) {
18            int diff = arr[j] - arr[i];
19
20            if (diff == k) {
21                found = 1;
22                break;
23            }
24            else if (diff < k)
25                j++;
26            else
27                i++;
28        } else {
29            j++;
30        }
31    }
32
33    printf("%d\n", found);
34    return 0;
35 }
```

	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! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)