

**Question 1** | Correct Mark 1.00 out of 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 int findDuplicate(int* nums, int n) {
3     int slow = nums[0];
4     int fast = nums[0];
5
6     do {
7         slow = nums[slow];
8         fast = nums[nums[fast]];
9     } while (slow != fast);
10    slow = nums[0];
11    while (slow != fast) {
12        slow = nums[slow];
13        fast = nums[fast];
14    }
15
16    return slow;
17 }
18
19 int main() {
20     int n;
21     scanf("%d", &n);
22
23     int arr[n];
24     for (int i = 0; i < n; i++) {
25         scanf("%d", &arr[i]);
26     }
27     int duplicate = findDuplicate(arr, n);
28     printf("%d\n", duplicate);
29
30     return 0;
31 }
32

```

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

	Input	Expected	Got	
✓	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.

[Back to Course](#)

**Question 1** | Correct Mark 1.00 out of 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 int findDuplicate(int* nums, int n) {
3     int slow = nums[0];
4     int fast = nums[0];
5
6     do {
7         slow = nums[slow];
8         fast = nums[nums[fast]];
9     } while (slow != fast);
10
11    slow = nums[0];
12    while (slow != fast) {
13        slow = nums[slow];
14        fast = nums[fast];
15    }
16    return slow;
17 }
18 int main() {
19     int n;
20     scanf("%d", &n);
21
22     int arr[n];
23     for (int i = 0; i < n; i++) {
24         scanf("%d", &arr[i]);
25     }
26     int duplicate = findDuplicate(arr, n);
27     printf("%d\n", duplicate);
28     return 0;
29 }
```

	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	✓

	Input	Expected	Got	
✓	5 1 1 2 3 4	1	1	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)

**Question 1** | Correct Mark 1.00 out of 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 void printIntersection(int* arr1, int n1, int* arr2, int n2) {
3     int i = 0, j = 0;
4     while (i < n1 && j < n2) {
5         if (arr1[i] < arr2[j]) {
6             i++;
7         } else if (arr1[i] > arr2[j]) {
8             j++;
9         } else {
10            printf("%d ", arr1[i]);
11            i++;
12            j++;
13        }
14    }
... ... ...
```

```

15     printf("\n");
16 }
17 int main() {
18     int T;
19     scanf("%d", &T);
20
21     while (T--) {
22         int n1;
23         scanf("%d", &n1);
24         int arr1[n1];
25         for (int i = 0; i < n1; i++) {
26             scanf("%d", &arr1[i]);
27         }
28
29         int n2;
30         scanf("%d", &n2);
31         int arr2[n2];
32         for (int i = 0; i < n2; i++) {
33             scanf("%d", &arr2[i]);
34         }
35
36         printIntersection(arr1, n1, arr2, n2);
37     }
38     return 0;
39 }
40

```

	<b>Input</b>	<b>Expected</b>	<b>Got</b>	
✓	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](#)

**Question 1** | Correct Mark 1.00 out of 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 void printIntersection(int* arr1, int n1, int* arr2, int n2) {
3     int i = 0, j = 0;
4     int firstPrinted = 0;
5     while (i < n1 && j < n2) {
6         if (arr1[i] < arr2[j]) {
7             i++;
8         } else if (arr1[i] > arr2[j]) {
9             j++;
10        } else {
11            if (firstPrinted) printf(" ");
12            printf("%d", arr1[i]);
13            firstPrinted = 1;
14        }
15    }
16 }
```

```

15         j++;
16     }
17 }
18 printf("\n");
19 }
20 int main() {
21     int T;
22     scanf("%d", &T);
23
24 while (T--) {
25     int n1;
26     scanf("%d", &n1);
27     int arr1[n1];
28 for (int i = 0; i < n1; i++) {
29     scanf("%d", &arr1[i]);
30 }
31     int n2;
32     scanf("%d", &n2);
33     int arr2[n2];
34 for (int i = 0; i < n2; i++) {
35     scanf("%d", &arr2[i]);
36 }
37
38     printIntersection(arr1, n1, arr2, n2);
39 }
40 return 0;
41 }
42

```

	<b>Input</b>	<b>Expected</b>	<b>Got</b>	
✓	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](#)

**Question 1** | Correct Mark 1.00 out of 1.00

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     int arr[n];
6     for (int i = 0; i < n; i++) {
7         scanf("%d", &arr[i]);
8     }
9     int k;
10    scanf("%d", &k);
11    int i = 0, j = 1;
12    while (j < n && i < n) {
13        int diff = arr[j] - arr[i];
14        if (diff == k && i != j) {
15            printf("1\n");
16            return 0;
17        } else if (diff < k) {
18            j++;
19        } else {
20            i++;
21            if (i == j) j++;
22        }
23    }
24    printf("0\n");
25    return 0;
26 }
27

```

	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](#)

**Question 1** | Correct Mark 1.00 out of 1.00

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     int arr[n];
6     for (int i = 0; i < n; i++) {
7         scanf("%d", &arr[i]);
8     }
9     int k;
10    scanf("%d", &k);
11    int i = 0, j = 1;
12    while (j < n && i < n) {
13        int diff = arr[j] - arr[i];
14
15        if (diff == k && i != j) {
16            printf("1\n");
17            return 0;
18        } else if (diff < k) {
19            j++;
20        } else {
21            i++;
22            if (i == j) j++;
23        }
24    }
25    printf("0\n");
26    return 0;
27 }
28 }
```

	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](#)