

LOGA PRABHA G B 2024-CSE ▾**L2****Started on** Wednesday, 8 October 2025, 8:41 AM**State** Finished**Completed on** Wednesday, 8 October 2025, 8:42 AM**Time taken** 56 secs**Marks** 1.00/1.00**Grade** 4.00 out of 4.00 (100%)

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 %)

```
#include <stdio.h>

#define MAX 10000

int main() {
    int n;
    scanf("%d", &n);

    int arr[MAX];
    int freq[MAX] = {0};
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
        freq[arr[i]]++;
    }

    if (freq[arr[i]] > 1) {
        printf("%d\n", arr[i]);
        return 0;
    }
}
```

	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.

LOGA PRABHA G B 2024-CSE ▾**L2****Started on** Wednesday, 8 October 2025, 8:43 AM**State** Finished**Completed on** Wednesday, 8 October 2025, 8:44 AM**Time taken** 1 min**Marks** 1.00/1.00**Grade** 4.00 out of 4.00 (100%)

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
3 int findDuplicate(int arr[], int n) {
4     int slow = arr[0];
5     int fast = arr[0];
6
7     do {
8         slow = arr[slow];
9         fast = arr[arr[fast]];
10    } while (slow != fast);
11
12    slow = arr[0];
13    while (slow != fast) {
14        slow = arr[slow];
15        fast = arr[fast];
16    }
17
18    return slow;
19}
20
21 int main() {
22     int n;
23     scanf("%d", &n);
24
25     int arr[n];
26     for (int i = 0; i < n; i++)
27         scanf("%d", &arr[i]);
28
29     int duplicate = findDuplicate(arr, n);
30     printf("%d\n", duplicate);
31
32     return 0;
33}
34

```

	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.

[Back to Course](#)



L2

Started on Wednesday, 8 October 2025, 8:44 AM

State Finished

Completed on Wednesday, 8 October 2025, 8:48 AM

Time taken 4 mins 35 secs

Marks 1.00/1.00

Grade 30.00 out of 30.00 (100%)

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
3 #define MAX 10000
4
5 void findIntersection(int a[], int n1, int b[], int n2) {
6     int i = 0, j = 0;
7
8     while (i < n1 && j < n2) {
9         if (a[i] < b[j]) {
10             i++;
11         } else if (a[i] > b[j]) {
12             j++;
13         } else {
14             printf("%d ", a[i]);
15             int val = a[i];
16             while (i < n1 && a[i] == val) i++;
17             while (j < n2 && b[j] == val) j++;
18         }
19     }
20     printf("\n");
21 }
```

```

23 int main() {
24     int T;
25     scanf("%d", &T);
26
27     while (T--) {
28         int n1, n2;
29         int a[MAX], b[MAX];
30
31         scanf("%d", &n1);
32         for (int i = 0; i < n1; i++)
33             scanf("%d", &a[i]);
34
35         scanf("%d", &n2);
36         for (int i = 0; i < n2; i++)
37             scanf("%d", &b[i]);
38
39         findIntersection(a, n1, b, n2);
40     }
41
42     return 0;
43 }
44

```

	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](#)

 LOGA PRABHA G B 2024-CSE ▾

L2

Started on Wednesday, 8 October 2025, 8:48 AM**State** Finished**Completed on** Wednesday, 8 October 2025, 8:50 AM**Time taken** 1 min 14 secs**Marks** 1.00/1.00**Grade** 30.00 out of 30.00 (100%)

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
3 #define MAX 10000
4
5 void findIntersection(int a[], int n1, int b[], int n2) {
6     int i = 0, j = 0;
7
8     while (i < n1 && j < n2) {
9         if (a[i] < b[j]) {
10             i++;
11         } else if (a[i] > b[j]) {
12             j++;
13         } else {
14             printf("%d ", a[i]);
15             i++;
16             j++;
17         }
18     }
19     printf("\n");
20 }
21
22 int main() {
```

```

23     int T;
24     scanf("%d", &T);
25
26     while (T--) {
27         int n1, n2;
28         int a[MAX], b[MAX];
29
30         scanf("%d", &n1);
31         for (int i = 0; i < n1; i++)
32             scanf("%d", &a[i]);
33
34         scanf("%d", &n2);
35         for (int i = 0; i < n2; i++)
36             scanf("%d", &b[i]);
37
38         findIntersection(a, n1, b, n2);
39     }
40
41     return 0;
42 }
43

```

	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](#)

LOGA PRABHA G B 2024-CSE ▾**L2****Started on** Wednesday, 8 October 2025, 8:50 AM**State** Finished**Completed on** Wednesday, 8 October 2025, 8:51 AM**Time taken** 56 secs**Marks** 1.00/1.00**Grade** 4.00 out of 4.00 (100%)

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
3 #define MAX 10000
4
5 int main() {
6     int n, k;
7     int A[MAX];
8
9     scanf("%d", &n);
10
11    for (int i = 0; i < n; i++)
12        scanf("%d", &A[i]);
13
14    scanf("%d", &k);
15
16    int i = 0, j = 1;
17    while (i < n && j < n) {
18        int diff = A[j] - A[i];
19
20        if (diff == k && i != j) {
21            printf("1\n");
22            return 0;
23        } else if (diff < k) {
24            j++;
25        } else {
26            i++;
27            if (i == j) j++;
28        }
29    }
30
31    printf("0\n");
32    return 0;
33}
34
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](#)

LOGA PRABHA G B 2024-CSE ▾**L2****Started on** Wednesday, 8 October 2025, 8:51 AM**State** Finished**Completed on** Wednesday, 8 October 2025, 8:52 AM**Time taken** 1 min 1 sec**Marks** 1.00/1.00**Grade** 4.00 out of 4.00 (100%)

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
3 #define MAX 10000
4
5 int main() {
6     int n, k;
7     int A[MAX];
8
9     scanf("%d", &n);
10
11    for (int i = 0; i < n; i++)
12        scanf("%d", &A[i]);
13
14    scanf("%d", &k);
15
16    int i = 0, j = 1;
17    while (i < n && j < n) {
18        int diff = A[j] - A[i];
19
20        if (diff == k && i != j) {
21            printf("1\n");
22            return 0;
23        } else if (diff < k) {
24            j++;
25        } else {
26            i++;
27            if (i == j) j++;
28        }
29    }
30
31    printf("0\n");
32    return 0;
33}
34

```

	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](#)