Started on	Monday, 19 August 2024, 10:20 AM
State	Finished
Completed on	Monday, 19 August 2024, 10:41 AM
Time taken	21 mins 3 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 2 3 4	1

Answer: (penalty regime: 0 %)

```
#include <stdio.h>

int main () {
    int n,a[50];
    scanf("%d", &n);
    for (int i=0; i<n; i++) {
        scanf("%d", &a[i]);
    }
    for (int j=0; j<n; j++) {
        for (int k=j+1; k<n; k++) {
            if (a[j]==a[k]) {
                printf("%d", a[k]);
            }
        }
    }
}
```

	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.

◀ 4-DP-Longest non-decreasing Subsequence

Jump to...

2-Finding Duplicates-O(n) Time Complexity,O(1) Space Complex

Started on	Monday, 19 August 2024, 10:27 AM
State	Finished
Completed on	Wednesday, 28 August 2024, 9:06 PM
Time taken	9 days 10 hours
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 2 3 4	1

Answer: (penalty regime: 0 %)

```
#include<stdio.h>

int main() {
    int n,j=0;
    scanf("%d",&n);
    int a[n];
    for(int i=0;i<n;i++)
        scanf("%d",&a[i]);
    for(int i=0;i<n;i++) {
       j+=a[i];
    }
    printf("%d",j-(n*(n-1))/2);
}
```

	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.

◀ 1-Finding Duplicates-O(n^2) Time Complexity,O(1) Space Complexity

Jump to...

3-Print Intersection of 2 sorted arrays-O(m*n)Time Complexity,O(1) Space Complex

Started on	Monday, 18 November 2024, 9:07 PM
State	Finished
Completed on	Monday, 18 November 2024, 9:20 PM
Time taken	12 mins 50 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:

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

Answer: (penalty regime: 0 %)

```
#include <stdio.h>

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

    while (t--) {
        int n1;
        scanf("%d", &n1);
        int a1[n1];
        for (int i = 0; i < n1; i++) {
            scanf("%d", &a1[i]);
        }

        int n2;
        scanf("%d", &n2);
```

```
        int a2[n2];
        for (int i = 0; i < n2; i++) {
            scanf("%d", &a2[i]);
        }

        int i = 0, j = 0;
        while (i < n1 && j < n2) {
            if (a1[i] == a2[j]) {
                printf("%d ", a1[i]);
                i++;
                j++;
            } else if (a1[i] < a2[j]) {
                i++;
            } else {
                j++;
            }
        }
        printf("\n");
    }

    return 0;
}
```

	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.

◄ 2-Finding Duplicates-O(n) Time Complexity,O(1) Space Complexity

Jump to...

4-Print Intersection of 2 sorted arrays-O(m+n)Time Complexity,O(1) Space Complex

Started on	Monday, 18 November 2024, 9:09 PM
State	Finished
Completed on	Monday, 18 November 2024, 9:19 PM
Time taken	10 mins 42 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,

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 3 5 4	1

Answer: (penalty regime: 0 %)

```c
#include <stdio.h>

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

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

    scanf("%d", &k);

    for (int i = 0; i < n; i++) {
        for (int j = i + 1; j < n; j++) {
            if (a[j] - a[i] == k) {
                printf("1\n");
                return 0;
            }
        }
    }

    printf("0\n");
    return 0;
}
```

	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.

◀ 4-Print Intersection of 2 sorted arrays-O(m+n)Time Complexity,O(1) Space Complexity

Jump to...

6-Pair with Difference -O(n) Time Complexity,O(1) Space Complex

Started on	Monday, 18 November 2024, 9:10 PM
State	Finished
Completed on	Monday, 18 November 2024, 9:17 PM
Time taken	7 mins 52 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,

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 3 5 4	1

Answer: (penalty regime: 0 %)

```
#include <stdio.h>

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

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

    scanf("%d", &k);

    int i = 0, j = 1;
    while (j < n) {
        if (a[j] - a[i] == k && i != j) {
            printf("1\n");
            return 0;
        } else if (a[j] - a[i] < k) {
            j++;
        } else {
            i++;
            if (i == j) {
                j++;
            }
        }
    }

    printf("0\n");
    return 0;
}

```

	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.

◄ 5-Pair with Difference-O(n^2)Time Complexity,O(1) Space Complexity

Jump to...