

(1)Question 1

Correct

Mark 1.00 out of 1.00

Flag question

Question text

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:

```
#include <stdio.h>
```

```
int main() {
```

```
    int n;
```

```
    scanf("%d", &n);
```

```
int arr[n];  
for(int i = 0; i < n; i++) {  
    scanf("%d", &arr[i]);  
}  
  
int duplicate = -1;  
for(int i = 0; i < n; i++) {  
    for(int j = i + 1; j < n; j++) {  
        if(arr[i] == arr[j]) {  
            duplicate = arr[i];  
            break;  
        }  
    }  
    if(duplicate != -1)  
        break;  
}  
  
printf("%d", duplicate);  
return 0;  
}
```

CODE:

```

1  #include <stdio.h>
2
3  int main() {
4      int n;
5      scanf("%d", &n);
6
7      int arr[n];
8      for(int i = 0; i < n; i++) {
9          scanf("%d", &arr[i]);
10     }
11
12     int duplicate = -1;
13     for(int i = 0; i < n; i++) {
14         for(int j = i + 1; j < n; j++) {
15             if(arr[i] == arr[j]) {
16                 duplicate = arr[i];
17                 break;
18             }
19         }
20         if(duplicate != -1)
21             break;
22     }
23
24     printf("%d", duplicate);
25     return 0;
26 }
27

```

	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! ✓

(2) Question 1

Not complete

Marked out of 1.00

Flag question

Question text

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:

```
#include <stdio.h>
```

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

```
int duplicate = -1;
for(int i = 0; i < n; i++) {
    for(int j = i + 1; j < n; j++) {
        if(arr[i] == arr[j]) {
            duplicate = arr[i];
            break;
        }
    }
    if(duplicate != -1)
        break;
}

printf("%d", duplicate);
return 0;
}
```

CODE:

Answer: (penalty regime: 0 %)

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

	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! ✓

(3) Question 1

Not complete

Marked out of 1.00

Flag question

Question text

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

Answer:

```
#include <stdio.h>
```

```
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++) {
```

```
            scanf("%d", &arr1[i]);
```

```
        }
```



```

scanf("%d", &n2);
int arr2[n2];
for (int i = 0; i < n2; i++) {
    scanf("%d", &arr2[i]);
}

int i = 0, j = 0;
while (i < n1 && j < n2) {
    if (arr1[i] == arr2[j]) {
        printf("%d ", arr1[i]);
        i++;
        j++;
    } else if (arr1[i] < arr2[j]) {
        i++;
    } else {
        j++;
    }
}
printf("\n");

return 0;
}

```

CODE:

```
1 #include <stdio.h>
2 int main() {
3     int T;
4     scanf("%d", &T);
5
6     while (T--) {
7         int n1, n2;
8         scanf("%d", &n1);
9         int arr1[n1];
10        for (int i = 0; i < n1; i++) {
11            scanf("%d", &arr1[i]);
12        }
13
14        scanf("%d", &n2);
15        int arr2[n2];
16        for (int i = 0; i < n2; i++) {
17            scanf("%d", &arr2[i]);
18        }
19
20        int i = 0, j = 0;
21        while (i < n1 && j < n2) {
22            if (arr1[i] == arr2[j]) {
23                printf("%d ", arr1[i]);
24                i++;
25                j++;
26            } else if (arr1[i] < arr2[j]) {
27                i++;
28            } else {
29                j++;
30            }
31        }
32        printf("\n");
33    }
34    return 0;
35 }
```

	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! ✓

(4) Question 1

Not complete

Marked out of 1.00

Flag question

Question text

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

Answer:

```
#include <stdio.h>
```

```
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++)
    scanf("%d", &arr1[i]);

scanf("%d", &n2);
int arr2[n2];
for (int i = 0; i < n2; i++)
    scanf("%d", &arr2[i]);

int i = 0, j = 0;

while (i < n1 && j < n2) {
    if (arr1[i] == arr2[j]) {
        printf("%d ", arr1[i]);
        i++;
        j++;
    } else if (arr1[i] < arr2[j]) {
        i++;
    } else {
        j++;
    }
}

printf("\n");
}

return 0;
```

}

CODE:

```
1  #include <stdio.h>
2  int main() {
3      int T;
4      scanf("%d", &T);
5
6      while (T--) {
7          int n1, n2;
8          scanf("%d", &n1);
9          int arr1[n1];
10         for (int i = 0; i < n1; i++)
11             scanf("%d", &arr1[i]);
12
13         scanf("%d", &n2);
14         int arr2[n2];
15         for (int i = 0; i < n2; i++)
16             scanf("%d", &arr2[i]);
17
18         int i = 0, j = 0;
19
20         while (i < n1 && j < n2) {
21             if (arr1[i] == arr2[j]) {
22                 printf("%d ", arr1[i]);
23                 i++;
24                 j++;
25             } else if (arr1[i] < arr2[j]) {
26                 i++;
27             } else {
28                 j++;
29             }
30         }
31         printf("\n");
32     }
33
34     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! ✓

(5) Question 1

Correct

Mark 1.00 out of 1.00

Flag question

Question text

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 3 5 4	1

Answer:

```
#include <stdio.h>
```

```
int main() {
```

```
    int n;
```

```
    scanf("%d", &n);
```

```
    int arr[n];
```

```
    for (int i = 0; i < n; i++) {
```

```
        scanf("%d", &arr[i]);
```

```
    }
```

```
    int k;
```

```
    scanf("%d", &k);
```

```
    int i = 0, j = 1;
```

```
    int found = 0;
```

```
    while (i < n && j < n) {
```

```
        int diff = arr[j] - arr[i];
```

```
        if (i != j && diff == k) {
```

```
            found = 1;
```

```
            break;
```

```
        } else if (diff < k) {
```

```
            j++;
```



```

        } else {
            i++;
        }
    }

    printf("%d", found);

    return 0;
}

```

CODE:

```

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

(6) Question 1

Not complete

Marked out of 1.00

Flag question

Question text

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:

```
#include <stdio.h>
```

```
int main() {
```

```
    int n;
```

```
    scanf("%d", &n);
```

```
    int arr[n];
```

```
    for (int i = 0; i < n; i++) {
```

```
        scanf("%d", &arr[i]);
```

```
    }
```

```
    int k;
```

```
    scanf("%d", &k);
```

```
int i = 0, j = 1;
int found = 0;

while (i < n && j < n) {
    int diff = arr[j] - arr[i];

    if (i != j && diff == k) {
        found = 1;
        break;
    }
    else if (diff < k) {
        j++;
    }
    else {
        i++;
    }
}

printf("%d", found);
return 0;
}
```

CODE:

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