



REC-CIS

GE23131-Programming Using C-2024

Quiz navigation



Show one page at a time

Finish review

Status Finished

Started Monday, 23 December 2024, 5:33 PM

Completed Tuesday, 17 December 2024, 8:40 AM

Duration 6 days 8 hours

Question 1

Correct

Marked out of 3.00

Flag question

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

Input Format

1. First line is number of test cases T. Following T lines contain:

2. N, followed by N integers of the array

3. The non-negative integer k

Output format

Print 1 if such a pair exists and 0 if it doesn't.

Example

Input:

```
1
3 1 3 5
4
```

Output:

```
1
```

Input:

```
1
3 1 3 5
99
```

Output:

```
0
```

Answer: (penalty regime: 0 %)

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

Answer: (penalty regime: 0 %)

```

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

```

	Input	Expected	Got	
✓	1 3 1 3 5 4	1	1	✓
✓	1 3 1 3 5 99	0	0	✓

Passed all tests! ✓

Question 2

Correct

Marked out of 5.00

Flag question

Sam loves chocolates and starts buying them on the 1st day of the year. Each day of the year, x , is numbered from 1 to Y . On days when x is odd, Sam will buy x chocolates; on days when x is even, Sam will not purchase any chocolates.

Complete the code in the editor so that for each day N_i (where $1 \leq x \leq N \leq Y$) in array arr , the number of chocolates Sam purchased (during days 1 through N) is printed on a new line. This is a function-only challenge, so input is handled for you by the locked stub code in the editor.

Input Format

The program takes an array of integers as a parameter.

The locked code in the editor handles reading the following input from `stdin`, assembling it into an array of integers (`arr`), and calling `calculate(arr)`.

The first line of input contains an integer, T (the number of test cases). Each line i of the T subsequent lines describes the i th test case as an integer, N_i (the number of days).

Constraints

$$1 \leq T \leq 2 \times 10^5$$

$$1 \leq N \leq 2 \times 10^6$$

$$1 \leq x \leq N \leq Y$$

Output Format

For each test case, T in `arr`, your `calculate` method should print the total number of chocolates Sam purchased by day N_i on a new line.

Sample Input 0

```
3
1
2
3
```

Sample Output 0

```
1
1
4
```

Explanation

Test Case 0: N = 1

Sam buys 1 chocolate on day 1, giving us a total of 1 chocolate. Thus, we print 1 on a new line.

Test Case 1: N = 2

Sam buys 1 chocolate on day 1 and 0 on day 2. This gives us a total of 1 chocolate. Thus, we print 1 on a new line.

Test Case 2: N = 3

Sam buys 1 chocolate on day 1, 0 on day 2, and 3 on day 3. This gives us a total of 4 chocolates. Thus, we print 4 on a new line.

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
2 int main()
3 {
4     int t;
5     scanf("%d",&t);
6     while(t--){
7         int n,c=0;
8         scanf("%d",&n);
9         for(int i=0;i<=n;i++){
10             if(i%2!=0) c=c+i;
11         }
12         printf("%d\n",c);
13     }
14 }
15
16
```

	Input	Expected	Got	
✓	3	1	1	✓
	1	1	1	
	2	4	4	
	3			
✓	10	1296	1296	✓
	71	2500	2500	
	100	1849	1849	
	86	729	729	
	54	400	400	
	40	25	25	
	9	1521	1521	
	77	25	25	
	9	49	49	
	13	2401	2401	
	98			

Passed all tests! ✓

Question 3

Correct

Marked out of
7.00Flag
question

The number of goals achieved by two football teams in matches in a league is given in the form of two lists. Consider:

- Football team A, has played three matches, and has scored $\{1, 2, 3\}$ goals in each match respectively.
- Football team B, has played two matches, and has scored $\{2, 4\}$ goals in each match respectively.
- Your task is to compute, for each match of team B, the total number of matches of team A, where team A has scored less than or equal to the number of goals scored by team B in that match.
- In the above case:
 - For 2 goals scored by team B in its first match, team A has 2 matches with scores 1 and 2.
 - For 4 goals scored by team B in its second match, team A has 3 matches with scores 1, 2 and 3.

Hence, the answer: $\{2, 3\}$.

Complete the code in the editor below. The program must return an array of m positive integers, one for each $\text{maxes}[i]$ representing the total number of elements $\text{nums}[j]$ satisfying $\text{nums}[j] \leq \text{maxes}[i]$ where $0 \leq j < n$ and $0 \leq i < m$, in the given order.

It has the following:

`nums[nums[0]...nums[n-1]]`: first array of positive integers
`maxes[maxes[0]...maxes[m-1]]`: second array of positive integers

Constraints

- $2 \leq n, m \leq 105$
- $1 \leq \text{nums}[j] \leq 109$, where $0 \leq j < n$.
- $1 \leq \text{maxes}[i] \leq 109$, where $0 \leq i < m$.

Input Format For Custom Testing

Input from stdin will be processed as follows and passed to the function.

The first line contains an integer n , the number of elements in `nums`.

The next n lines each contain an integer describing `nums[j]` where $0 \leq j < n$.

The next line contains an integer m , the number of elements in `maxes`.

The next m lines each contain an integer describing `maxes[i]` where $0 \leq i < m$.

Sample Case 0

Sample Input 0

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

Sample Input 0

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

Sample Output 0

```
2
4
```

Explanation 0

We are given $n = 4$, $nums = [1, 4, 2, 4]$, $m = 2$, and $maxes = [3, 5]$.

1. For $maxes[0] = 3$, we have 2 elements in $nums$ ($nums[0] = 1$ and $nums[2] = 2$) that are $\leq maxes[0]$.
2. For $maxes[1] = 5$, we have 4 elements in $nums$ ($nums[0] = 1$, $nums[1] = 4$, $nums[2] = 2$, and $nums[3] = 4$) that are $\leq maxes[1]$.

Thus, the function returns the array $[2, 4]$ as the answer.

Sample Case 1

Sample Input 1

```
5
2
10
5
4
8
4
3
1
7
8
```

Sample Output 1

```
1
0
3
4
```

Explanation 1

We are given, $n = 5$, $nums = [2, 10, 5, 4, 8]$, $m = 4$, and $maxes = [3, 1, 7, 8]$.

1. For $maxes[0] = 3$, we have 1 element in $nums$ ($nums[0] = 2$) that is $\leq maxes[0]$.
2. For $maxes[1] = 1$, there are 0 elements in $nums$ that are $\leq maxes[1]$.
3. For $maxes[2] = 7$, we have 3 elements in $nums$ ($nums[0] = 2$, $nums[2] = 5$, and $nums[3] = 4$) that are $\leq maxes[2]$.
4. For $maxes[3] = 8$, we have 4 elements in $nums$ ($nums[0] = 2$, $nums[2] = 5$, $nums[3] = 4$ and $nums[4] = 8$) that are \leq

Sample Output 1

1
0
3
4

Explanation 1

We are given, $n = 5$, $\text{nums} = [2, 10, 5, 4, 8]$, $m = 4$, and $\text{maxes} = [3, 1, 7, 8]$.

1. For $\text{maxes}[0] = 3$, we have 1 element in nums ($\text{nums}[0] = 2$) that is $\leq \text{maxes}[0]$.
2. For $\text{maxes}[1] = 1$, there are 0 elements in nums that are $\leq \text{maxes}[1]$.
3. For $\text{maxes}[2] = 7$, we have 3 elements in nums ($\text{nums}[0] = 2$, $\text{nums}[2] = 5$, and $\text{nums}[3] = 4$) that are $\leq \text{maxes}[2]$.
4. For $\text{maxes}[3] = 8$, we have 4 elements in nums ($\text{nums}[0] = 2$, $\text{nums}[2] = 5$, $\text{nums}[3] = 4$, and $\text{nums}[4] = 8$) that are $\leq \text{maxes}[3]$.

Thus, the function returns the array $[1, 0, 3, 4]$ as the answer.

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
2 int main()
3 {
4     int s1,s2,ans;
5     scanf("%d",&s1);
6     int ta[s1];
7     for(int i=0;i<s1;i++)
8         scanf("%d",&ta[i]);
9     scanf("%d",&s2);
10    int tb[s2];
11    for(int i=0;i<s2;i++)
12        scanf("%d",&tb[i]);
13    for(int j=0;j<s2;j++)
14    {
15        ans=0;
16        for(int i=0;i<s1;i++)
17        {
18            if(tb[j]>=ta[i])
19                ans++;
20        }
21        printf("%d\n",ans);
22    }
23 }
24
```

	Input	Expected	Got	
✓	4 1 4 2 4 2 3 5	2 4	2 4	✓
✓	5 2 10 5 4 8 4 3 1 7 8	1 0 3 4	1 0 3 4	✓

Passed all tests! ✓

Finish review