

One-Dimensional Arrays

Check pair with difference k

Problem Statement:

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.

Sample Input:

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

Sample Output:

```
1
```

Program:

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

```
29     }  
30     printf("%d\n",flag);  
31 }  
32 }
```

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

Passed all tests! ✓

Chocolates

Problem Statement:

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_i 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
```

Program:

```
1  #include<stdio.h>
2  int main()
3  {
4      int t;
5      scanf("%d",&t);
6      while(t-->0)
7      {
8          int n,c=0;
9          scanf("%d",&n);
10         for(int i=0;i<=n;i++)
11         {
12             if(i%2!=0)
13                 c=c+i;
14         }
15         printf("%d\n",c);
16     }
17 }
```

	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! ✓

Football Scores

Problem Statement:

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:

$\text{nums}[\text{nums}[0], \dots, \text{nums}[n-1]]$: first array of positive integers

$\text{maxes}[\text{maxes}[0], \dots, \text{maxes}[m-1]]$: 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 $\text{nums}[j]$ 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 $\text{maxes}[i]$ where $0 \leq i < m$.

Sample Input

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

Sample Output

```
2
4
```


Program:

```
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 }
```

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

Passed all tests! ✓