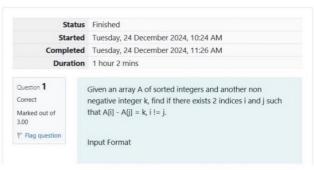
## GE23131-Programming Using C-2024





REC-CIS

```
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:
3135
Output:
Input:
99
Output:
Answer: (penalty regime: 0 %)
  1 |#include<stdio.h>
         int main(){
   int T;
   scanf("%d",&T);
               scanf("%d",&T);
while(T--){
   int n,k;
   scanf("%d",&n);
   int a[n];
   for(int i=0;i<n;i++){
      scanf("%d",&a[i]);
}</pre>
    10
    11
                   12
    13
   15 +
    17
    19
    20
                    if(found) printf("1\n");
else printf("0\n");
   21
   22
23
               return 0:
    24
                    Expected Got
         Input
```

3 1 3 5

Passed all tests! 🗸

	Input	Expected	Got	
/	1 3 1 3 5 4	1	1	~
,	1 3 1 3 5 99	Θ	0	~

Question 2
Correct
Marked out of 5.00 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 Ni (where  $1 \le x \le N \le Y$ ) in array arr, the number of chocolat 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.

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 test case as an integer, Ni (the number of days).

1 ≤ T ≤ 2 × 105 1 ≤ N ≤ 2 × 106

 $1 \le x \le N \le Y$ 

**Output Format** 

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

Sample Input 0

2

1 4

Explanatio

Test Case 0: N = 1

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

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

ver: (penalty regime: 0 %)

1	#include <stdio.h></stdio.h>	
2 +	void nchoc(int a[],int T){	
3 ,	for(int i=0;i <t;i++){< td=""></t;i++){<>	
4	int N=a[i];	
5	int sum=0;	
6 +	for(int day=1;day<=N;day+=2){	
7	sum+=day;	
3	)	
9	printf("%d\n",sum);	
9	)	
	}	
2 +	int main()(	
3	int T;	
1	scanf("%d",&T);	
5	int a[T];	
6 +	for(int i=0;i <t;i++){< td=""></t;i++){<>	
7	scanf("%d",&a[i]);	
8	}	
9	nchoc(a,T);	
0	return 0;	
1	}	



```
Sample Output 0
4
Explanation 0
We are given n = 4, nums = [1, 4, 2, 4], m = 2, and ma

    For maxes(0) = 3, we have 2 elements
    1 and nums(2) = 2) that are ≤ maxes(0).

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

Sample Case 1
Sample Input 1
5
2
10
8
8
Sample Output 1
0
4
Explanation 1
We are given, n = 5, nums = [2, 10, 5, 4, 8], m = 4, a = [3, 1, 7, 8].

    For maxes[0] = 3, we have 1 element in nums (nums[0] = 2) that is ≤ maxes[0].

     For maxes[1] = 1, there are 0 elements in nums that a
≤ maxes[1].
3. For maxes[2] = 7, we have 3 elements in nums (nun = 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[= 2, nums[2] = 5, nums[3] = 4, and nums[4] = 8) that are \le maxes[3].
  Thus, the function returns the array [1, 0, 3, 4] as the an
                Input Expected Got
  Passed all tests! ✓
```