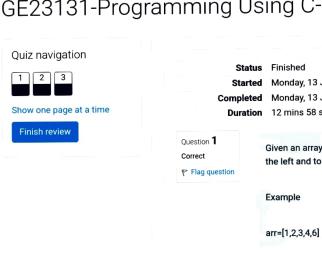
## GE23131-Programming Using C-2024



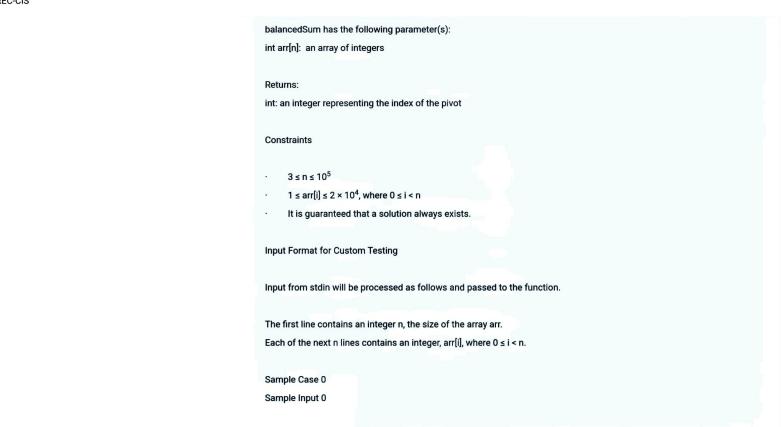


Given an array of numbers, find the index of the smallest array element (the pivot), for which the sums of all elements to the left and to the right are equal. The array may not be reordered.

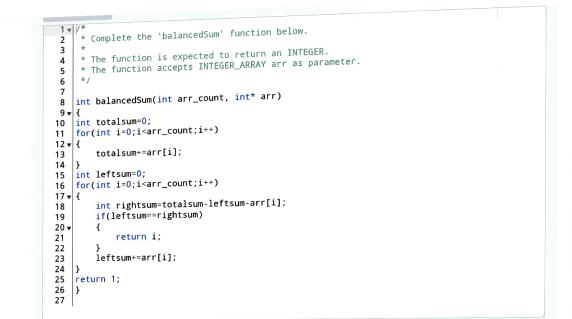
- the sum of the first three elements, 1+2+3=6. The value of the last element is 6.
- Using zero based indexing, arr[3]=4 is the pivot between the two subarrays.
  - The index of the pivot is 3.

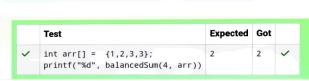
**Function Description** 

Complete the function balancedSum in the editor below.









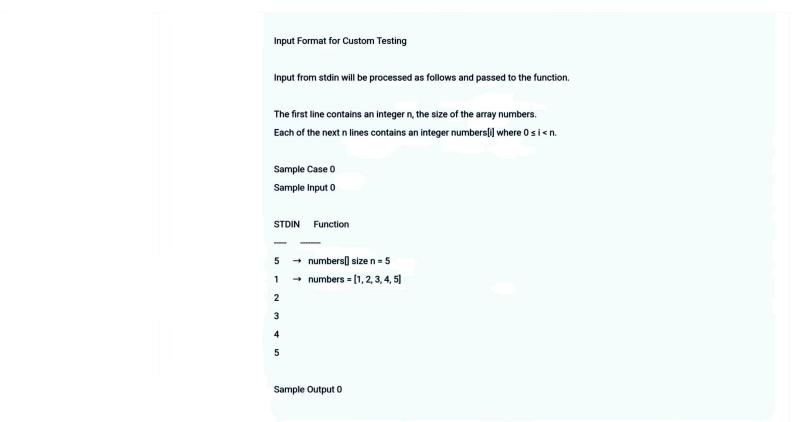
```
THE TURNELLOR IS EXPECTED to return on integer.
 5
     * The function accepts INTEGER_ARRAY arr as parameter.
    int balancedSum(int arr_count, int* arr)
 9 ₩
   int totalsum=0;
    for(int i=0;i<arr_count;i++)</pre>
12 ▼ {
13
        totalsum+=arr[i];
14
   int leftsum=0;
   for(int i=0;i<arr_count;i++)</pre>
17 ▼ {
18
        int rightsum=totalsum-leftsum-arr[i];
        if(leftsum==rightsum)
19
20 ▼
21
            return i;
22
23
        leftsum+=arr[i];
24
   return 1;
26
27
```

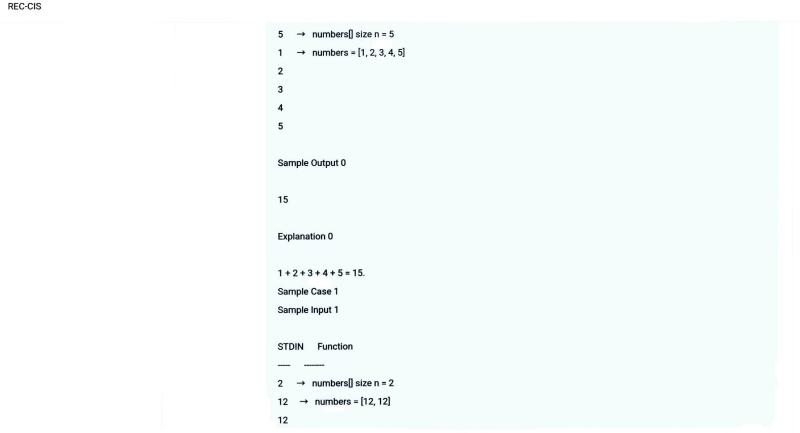
	Test	Expected	Got	
~	<pre>int arr[] = {1,2,3,3}; printf("%d", balancedSum(4, arr))</pre>	2	2	

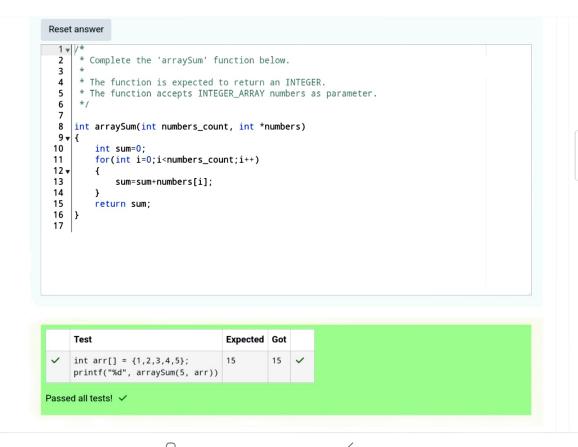
Question <b>2</b>
Correct
Flag question

1 < numbers[i] < 104

Calculate the sum of an array of integers.
Example
numbers = [3, 13, 4, 11, 9]
The sum is 3 + 13 + 4 + 11 + 9 = 40.
Function Description
Complete the function arraySum in the editor below.
arraySum has the following parameter(s):
int numbers[n]: an array of integers
Returns
int: integer sum of the numbers array
Constraints
$1 \le n \le 10^4$







## Answer: (penalty regime: 0 %)

## Reset answer

```
1 ▼
      * Complete the 'minDiff' function below.
 3
     * The function is expected to return an INTEGER.
 4
     * The function accepts INTEGER_ARRAY arr as parameter.
 5
 6
      */
 7
    #include<stdlib.h>
    int compare(const void*a,const void*b)
 8
 9 ▼ {
         return (*(int*)a-*(int*)b);
10
11
    int minDiff(int arr_count, int* arr)
12
13 ▼ {
        qsort(arr,arr_count,sizeof(int),compare);
14
         int totaldiff=0:
15
        for(int i=1;i<arr_count;i++)</pre>
16
17 ▼
```

```
Complete the 'minDiff' function below.
     * The function is expected to return an INTEGER.
     * The function accepts INTEGER_ARRAY arr as parameter.
    #include<stdlib.h>
    int compare(const void*a,const void*b)
9 ▼ {
10
        return (*(int*)a-*(int*)b);
11
12
    int minDiff(int arr_count, int* arr)
13 ▼ {
14
        qsort(arr,arr_count,sizeof(int),compare);
15
        int totaldiff=0;
16
        for(int i=1;i<arr_count;i++)</pre>
17 ▼
18
            totaldiff+=abs(arr[i]-arr[i-1]);
19
20
        return totaldiff;
21
22
```

1 ▼ /\*