

## Explanation 1

- The first and last elements are equal to 1.
- Using zero based indexing, arr[1]=2 is the pivot between the two subarrays.
- The index of the pivot is 1.

**Answer:** (penalty regime: 0 %)

Reset answer

```
1  /*
2  * Complete the 'balancedSum' function be
3  *
4  * The function is expected to return an
5  * The function accepts INTEGER_ARRAY arr
6  */
7
8  int balancedSum(int arr_count, int* arr)
9  {
10     int left,right=0;
11     for(int i=0;i<arr_count;i++)
12     {
13         right+=arr[i];
14     }
15     for(int i=0;i<arr_count;i++)
16     {
17         if(left==(right-arr[i]))
18             return 2;
19         left+=arr[i];
20         right-=arr[i];
21     }
22     return 1;
23 }
24
```

	Test	Expected
✓	int arr[] = {1,2,3,3}; printf("%d", balancedSum(4, arr))	2

Passed all tests! ✓

Sample Output 1

24

Explanation 1

12 + 12 = 24.

**Answer:** (penalty regime: 0 %)

Reset answer

```
1  /*
2  * Complete the 'arraySum' function below
3  *
4  * The function is expected to return an
5  * The function accepts INTEGER_ARRAY num
6  */
7
8  int arraySum(int numbers_count, int *numb
9  {
10     int sum=0;
11     for(int i=0;i<numbers_count;i++)
12     {
13         sum+=numbers[i];
14     }
15     return sum;
16 }
17
```

	Test	Expected	Got
✓	int arr[] = {1,2,3,4,5}; printf("%d", arraySum(5, arr))	15	15

Passed all tests! ✓

Sample Output 1 Explanation  $n = 2$   $arr = [3, 2]$  There is no need to rearrange because there are only two elements. The final answer is  $|3 - 2| = 1$ .

**Answer:** (penalty regime: 0 %)

Reset answer

```
1  /*
2  * Complete the 'minDiff' function below.
3  *
4  * The function is expected to return an
5  * The function accepts INTEGER_ARRAY arr
6  */
7
8  int minDiff(int arr_count, int* arr)
9  {
10     for(int i=0;i<arr_count-1;i++)
11     {
12         for(int j=0;j<arr_count-i-1;j++)
13         {
14             if(arr[j]>arr[j+1])
15             {
16                 int temp=arr[j];
17                 arr[j]=arr[j+1];
18                 arr[j+1]=temp;
19             }
20         }
21     }
22     int sum=0;
23     for(int i=0;i<arr_count-1;i++)
24     {
25         sum+=abs(arr[i]-arr[i+1]);
26     }
27     return sum;
28 }
29
```

	Test	Expected	Got
✓	<pre>int arr[] = {5, 1, 3, 7, 3}; printf("%d", minDiff(5, arr))</pre>	6	6

Passed all tests! ✓