```c
/*
 * Complete the 'balancedSum' function below.
 *
 * The function is expected to return an INTEGER.
 * The function accepts INTEGER_ARRAY arr as parameter.
 */

int balancedSum(int arr_count, int* arr)
{
    int l=0,r=0;
    for(int i=0;i<arr_count;i++){
        r+=arr[i];
    }
    for(int i=0;i<arr_count;i++){
        if(l == r-arr[i]){
            return i;
        }
        l +=arr[i];
        r -=arr[i];
    }
    return 1;

}

```

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

Passed all tests! ✓

```c
/*
 * Complete the 'arraySum' function below.
 *
 * The function is expected to return an INTEGER.
 * The function accepts INTEGER_ARRAY numbers as parameter.
 */

int arraySum(int numbers_count, int *numbers)
{
    int s=0;
    for(int i=0;i<numbers_count;i++){
        s+=numbers[i];
    }
    return s;

}

```

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

Passed all tests! ✓

```c
/*
 * Complete the 'minDiff' function below.
 *
 * The function is expected to return an INTEGER.
 * The function accepts INTEGER_ARRAY arr as parameter.
 */

int minDiff(int arr_count, int* arr)
{
    for(int i=0;i<arr_count;i++){
        for(int j=i;j<arr_count;j++){
            if(i!=j){
                if(arr[i]>arr[j]){
                    int temp=arr[j];
                    arr[j]=arr[i];
                    arr[i]=temp;
                }
            }
        }
    }
    int m=0;
    for(int i=0;i<arr_count-1;i++){
        m+=arr[i+1]-arr[i];
    }
    return m;
}

```

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

Passed all tests! ✓