

```

6  */
7
8  int balancedSum(int arr_count, int* arr)
9  {
10     int leftSum=0,rightSum=0;
11     for(int i=0;i<arr_count;i++)
12     {
13         rightSum+=arr[i];
14     }
15     for(int i=0;i<arr_count;i++)
16     {
17         rightSum-=arr[i];
18         if(leftSum==rightSum)
19         {
20             return i;
21         }
22         leftSum+=arr[i];
23     }
24     return 1;
25 }
26

```

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

Passed all tests! ✓

```

1  */
2  * Complete the 'arraySum' function below.
3  *
4  * The function is expected to return an INTEGER.
5  * The function accepts INTEGER_ARRAY numbers as parameter.
6  */
7
8  int arraySum(int numbers_count, int *numbers)
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! ✓

```

1 1 /*
2 2  * Complete the 'minDiff' function below.
3 3  *
4 4  * The function is expected to return an INTEGER.
5 5  * The function accepts INTEGER_ARRAY arr as parameter.
6 6  */
7 7
8 8 int compare(const void* a,const void* b)
9 9 {
10 10     return (*(int*)a)-*(int*)b);
11 11 }
12 12 int minDiff(int arr_count,int* arr)
13 13 {
14 14     qsort(arr,arr_count,sizeof(int),compare);
15 15     int sum=0;
16 16     for(int i=1;i<arr_count;++i)
17 17     {
18 18         sum+=abs(arr[i]-arr[i-1]);
19 19     }
20 20     return sum;
21 21 }
22 22

```

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

Passed all tests! ✓