Telerik Academy Alpha / C# - 17 July 2017

Task 2: Moving back and forward

Description

Your task is to calculate the sum of all forward and all backwards moves in an array. On the input you will receive the starting point, the array itself with all the values. After that you will receive multiple lines with instructions in the format **[Steps Direction Size]** separated by single space until you receive **exit**.

The starting position must be read from the console. Direction could be **forward** or **backwards**. If you get out of the boundaries of the array you should enter from the other side. The starting position is not included in the sum.

As final result you must print the result for forward moves and the result for the backwards as well.

Example:

0
10,20,30,40,50
2 forward 1
2 backwards 1
3 forward 2
3 backwards 2
exit

- We start at **position 0**.
- We move **2 times forward** with **size 1** => **sum = 20 + 30 = 50**
- We move **2 times backwards** with **size 1** => **sum = 20 + 10 = 30**
- We move **3 times forward** with **size 2** => **sum = 30 + 50 + 20 = 100**
- We move **3 time backwards** with **size 2** => **sum = 50 + 30 + 10 = 90**
- **Forward = 150**
- **Backwards = 120**

Input

All input data is read from the standard input (the console)

- On the first line you will receive the starting point in the array (it is not included in the sum)
- On the second line you will receive the array as a string (**comma delimited integers**)
- On the next lines until you find **exit** command you will receive strings in format **[Steps Direction Size]** delimited by **single space**

Output

The output data is printed on the standard output (the console)

- On the first line you should print the sum of forward moves **Forward: forward sum**
- On the second line you should print the sum of backwards moves **Backwards: backwards sum**

Constraints

- The integers in the array will be in the range [0, 10000]
- The **steps** will be an integer in the range [0, 10000]
- The **direction** will be string in format **"forward"** or **"backwards"**
- The **size** will be an integer in the range [0, 10000]
- **The input data will always be correct and there is no need to check it explicitly**
- **Time limit: 0.5 s**
- **Memory limit: 16 MB**

Sample Tests

Sample Input 1

```
0
10,20,30,40,50
2 forward 1
2 backwards 1
3 forward 2
3 backwards 2
exit
```

Sample Output 1

```
Forward: 150
Backwards: 120
```

Sample Input 2

```
4
10,20,30,40,50
2 forward 1
2 backwards 1
3 forward 2
3 backwards 2
exit
```

Sample Output 2

```
Forward: 100
Backwards: 170
```

Sample Input 3

```
0
10,20
1 forward 1
2 backwards 1
1 forward 1
exit
```

Sample Output 3

```
Forward: 30
Backwards: 30
```