PRODUCT COMPANIES PROGRAMS - SET 002

Solved Challenges **9/10**

Back To Challenges List

Robot No Movement Count

ID:3171   **Solved By 440 Users**

As a final year project certain students in a college have designed a Robot which can move front, back, left or right in a given rectangular grid of dimension L*B units (L denotes the length from left to right and B denotes the breadth from top to bottom). Always the robot moves in units which are of integer values. The robot cannot move outside the grid (That is the robot cannot go beyond L and B units). A sequence of N movement instructions are given to the robot to move in the desired direction (F-front or up, B-back or down, L-left, R-right) and the robot moves if the destination falls within the limit of the grid dimensions. Else the robot does not move. Assume the robot always starts at the bottom left of the grid. The program must print the number of movement instructions C for which the robot did not move (as the destination was outside the grid)

Input Format:
The first line contains L and B separated by a space.
The second line contains N which denotes the number of instructions.
The third line contains N instructions separated by one or more spaces.

Output Format:
The first line contains C which denoted the count of instructions for which the robot did not move.

Boundary Conditions:
1 <= L, B <= 999
1 <= N <= 999
C <= N

Example Input/Output 1:
Input:
6 5
9
3R 2L 11R 4F 4R 2F 3B 5L 4B

Output:
3

Explanation:
The robot did not move for the instructions 11R, 2F and 4B as they will make the robot go outside the grid.

Max Execution Time Limit: 5000 millisecs

Ambiance

Python3 (3.x)

Reset

```python
1   L,B = map(int,input().strip().split())
2   N = int(input())
3   inst = list(input().strip().split())
4
5   x=0
6   y=0
7   count=0
8   for i in inst:
9       if(i[-1]=="R"):
10          x+=int(i[:-1])
11          if(x>L):
12              count+=1
13              x-=int(i[:-1])
14      elif(i[-1]=="L"):
15          x-=int(i[:-1])
16          if(x<0):
17              count+=1
18              x+=int(i[:-1])
19      elif(i[-1]=="F"):
20          y+=int(i[:-1])
21          if(y>B):
22              count+=1
23              y-=int(i[:-1])
24      elif(i[-1]=="B"):
25          y-=int(i[:-1])
26          if(y<0):
27              count+=1
28              y+=int(i[:-1])
29  print(count)
30
31
```

Please wait while we run the program ....

☐ Run with a custom test case (Input/Output)