Plus Minus

by vatsalchanana

Problem | Submissions | Leaderboard | Discussions | Editorial

Given an array of integers, calculate which fraction of its elements are *positive*, which fraction of its elements are *negative*, and which fraction of its elements are *zeroes*, respectively. Print the decimal value of each fraction on a new line.

Note: This challenge introduces precision problems. The test cases are scaled to six decimal places, though answers with absolute error of up to 10^{-4} are acceptable.

Input Format

The first line contains an integer, N, denoting the size of the array.
The second line contains N space-separated integers describing an array of numbers $(a_0, a_1, a_2, \ldots, a_{n-1})$.

Output Format

You must print the following 3 lines:

1. A decimal representing of the fraction of *positive* numbers in the array.
2. A decimal representing of the fraction of *negative* numbers in the array.
3. A decimal representing of the fraction of *zeroes* in the array.

Sample Input

```
6
-4 3 -9 0 4 1
```

Sample Output

```
0.500000
0.333333
0.166667
```

Explanation

There are 3 positive numbers, 2 negative numbers, and 1 zero in the array.
The respective fractions of positive numbers, negative numbers and zeroes are $\frac{3}{6} = 0.500000$, $\frac{2}{6} = 0.333333$ and $\frac{1}{6} = 0.166667$, respectively.

```python
#!/bin/python

import sys


n = int(raw_input().strip())
arr = map(int,raw_input().strip().split(' '))
pos = 0
neg = 0
zero = 0

for i in range(0,n):
    if arr[i]>0:
        pos+=1
    if arr[i]<0:
        neg+=1
    if arr[i]==0:
        zero+=1

pos1=(float(pos)/float(n))
neg1=(float(neg)/float(n))
zero1=(float(zero)/float(n))
print("%.6f" %pos1)
print("%.6f" %neg1)
print("%.6f" %zero1)
```

Line: 19 Col: 9

Upload Code as File Test against custom input Run Code Submit Code

Congrats, you solved this challenge!

- ✔ Test Case #0
- ✔ Test Case #1
- ✔ Test Case #2
- ✔ Test Case #3
- ✔ Test Case #4
- ✔ Test Case #5
- ✔ Test Case #6
- ✔ Test Case #7
- ✔ Test Case #8
- ✔ Test Case #9
- ✔ Test Case #10

Next Challenge