A: Album of numbers

Memory limit **128 MB**

Johnny is a collector of numbers — he has a large album with them, which he keeps updating by inserting new numbers and deleting old ones.

This album is Johnny's treasure most prized, so Johnny wants to be sure that no one will ever replace his numbers with different ones. Therefore, he maintains a checksum for his album, determined as follows. He considers all distinct nonempty subsets of the numbers in his album, in each such subset he determines the minimum, and finally he calculates the arithmetic mean of these minima. As any number may appear multiple times in Johnny's album, the "subsets" are multisets, i.e., each number may appear in any subset multiple times (but no more than in the whole album). Two such multisets are equal if and only if all numbers have the same multiplicity in both. For instance, an album $\{1, 2, 2, 5\}$ has the following 11 distinct subsets: $\{1\}$, $\{1, 2\}$, $\{1, 2, 2\}$, $\{1, 2, 2, 5\}$, $\{1, 2, 5\}$, $\{1, 5\}$, $\{2\}$, $\{2, 2\}$, $\{2, 2, 5\}$, $\{2, 5\}$ and $\{5\}$.

Unfortunately, Johnny's album changes quite often, making maintaining the checksum manually rather tedious. Help him out by writing a program that will: read the number of updates to be applied to the album and descriptions of these operations, determines the checksum after each successive operation, and prints the checksums to the standard output.

Input

In the first line of input, there is a single positive integer n ($1 \leq n \leq 250\,000$) that specifies the number of updates to the album. The n lines that follow specify these operations, one per line, as follows. Each update is given as a `+` or `-` sign, a single space, and a number a_i ($1 \leq a_i \leq 1\,000\,000\,000$). These describe inserting the number a_i to the album and removing a appearance of the number a_i from the album, respectively.

You may assume that the album is initially empty, afterwards it is always nonempty, and that only numbers present in the album are ever deleted.

Output

Your program should print exactly n lines to the output. The i-th such line should contain the checksum after the i-th update, as prescribed by Johnny's formula.

An answer is deemed correct if, for each checksum, the absolute or relative error with respect to the correct value does not exceed 10^{-6}.

Example

Input	Output
4 + 1 + 2 + 2 + 5	1.000000000 1.333333333 1.400000000 1.727272727

After all the insertions, the album's contents are $\{1, 2, 2, 5\}$. At that point, Johnny considers the following 11 subsets: $\{1\}$, $\{1, 2\}$, $\{1, 2, 2\}$, $\{1, 2, 2, 5\}$, $\{1, 2, 5\}$, $\{1, 5\}$, $\{2\}$, $\{2, 2\}$, $\{2, 2, 5\}$, $\{2, 5\}$ and $\{5\}$. The respective minima of these subsets are: $1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 5$. Hence, the corresponding checksum is $\frac{19}{11} \approx 1.727272727$.

Input	Output
6 + 10 + 13 - 10 + 7 + 2 + 5	10.000000000 11.000000000 13.000000000 9.000000000 5.000000000 4.200000000