

Problem C. Coconuts

Input file: *standard input*
Output file: *standard output*
Time limit: 3 seconds
Memory limit: 512 mebibytes

Consider n coconuts and an array d describing them. The value d_i is the durability of the i -th coconut. It means that the i -th coconut will be cracked after d_i hits.

Then the coconuts were shuffled, so it is now impossible to determine which coconut has which durability.

Your goal is to crack as many coconuts as possible using exactly k hits. What is the expected number of cracked coconuts if you use the best possible strategy?

For each hit, you can arbitrarily select one coconut and hit it. After that, you see if the coconut has cracked or not.

Input

The first line contains two integers n and k : the number of coconuts and the number of hits required, respectively.

The second line contains n integers d_1, d_2, \dots, d_n : durabilities of coconuts.

Constrains:

- $1 \leq n \leq 10$;
- $1 \leq d_i \leq 10$;
- $1 \leq k \leq \sum_{i=1}^n d_i$.

Output

Output one real number which denotes the expected number of cracked coconuts if you use the best possible strategy. The relative or absolute error of the result should not exceed 10^{-6} .

Examples

standard input	standard output
3 2 1 3 3	0.666666666667
4 5 2 2 3 3	1.833333333333

Explanations

Here are the best strategies for examples:

1. Hit a random coconut, then hit another one.
2. Choose a coconut, hit it until it cracks, then switch to another until it cracks, etc.