

## So Many Possibilities...

Input file:            standard input  
Output file:           standard output  
Time limit:            3 seconds  
Memory limit:         1024 megabytes

Hearthstone is a free-to-play online digital collectible card game developed and published by Blizzard Entertainment. The game is a turn-based card game between two opponents, using constructed decks of 30 cards along with a selected hero with a unique power. Players use their limited mana crystals to cast spells or summon minions to attack the opponent, with the goal of destroying the opponent's hero.

Kurusu loves playing Hearthstone and plays her Highlander Mage deck very well. However, sometimes she isn't lucky enough, so that when she plays the card *Reno the Relicologist* (as shown in the image below), it always fails to kill many of the enemy minions. Now she wonders, given the  $n$  enemy minions on board and their remaining health, if she may deal  $m$  damage randomly split among them, what is the expected number of enemy minions that would be killed?



Рис. 1: Reno the Relicologist

One may think of dealing  $m$  damage randomly split among minions as  $m$  separate one-damage hits, processed one by one. Each one-damage hit targets a minion with positive health uniformly at random, and then decreases its health by one. Note that the one-damage hit will never be distributed to a dying minion, i.e., a minion whose health is already zero.

Since there are so many possibilities in this process, please help her calculate the expected number of minions with zero health after the process above.

### Input

The first line of input contains two integers  $n$  ( $1 \leq n \leq 15$ ) and  $m$  ( $1 \leq m \leq 100$ ), the number of minions and the total amount of damage to distribute.

The second line contains  $n$  integers  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 100, \sum a_i \geq m$ ), where  $a_i$  denotes the remaining health of the  $i$ -th minion.

## Output

Output the expected number of minions killed during the process within an absolute error of no more than  $10^{-6}$ .

## Examples

standard input	standard output
2 2 2 2	0.5
3 3 1 2 3	1.0833333333333333