Drawing Lots

Lea is organizing a type of lottery where customers can, for a certain cost, draw lots from an urn. Each lot is either winning or losing, with a certain probability, and the customers win prizes if they draw winning lots. However, instead of the standard lottery where they draw one lot and win the prize if this one lot is winning, Lea puts a certain twist on the lottery: there are several prizes, numbered from 1 to n, with increasing value, and each customer draws one lot for the first prize, two lots for the second one, and so on, until they draw n lots for the last prize. They win a prize if and only if all the lots for that prize show up winning. They can win several prizes if the lots for all of these prizes show up winning. All the lots have the same probability of showing winning, and Lea ensures that this probability always stays the same during drawings (e.g. by appropriately replacing lots). There is one fixed cost to draw lots for all prizes.

As Lea is running the lottery, she wants to ensure that she does not make a loss. This means that the expected total payoff, which is the sum of the value of the prizes times the expected winning probability to win that prize, minus the cost to draw lots, should not be positive. Still, she wants the winning probabilty to be as high as possible so the customers stay happy. Lea has already fixed the number of prizes, their values and the cost. Can you tell her what the maximal probability for a lot to show up winning should be such that she does not make a loss?

Input

The first line of the input contains an integer t. t test cases follow, each of them separated by a blank line.

Each test case starts with a line containing two integers n b, where n is the number of prizes and b is the cost to draw lots. A second line follows, containing n space-separated integers $a_1 \dots a_n$, where a_i is the value of prize i.

Output

For each test case, output one line containing "Case #i: x" where i is its number, starting at 1, and x is the maximal winning probability of a lot such that the expected total payoff is less or equal than 0 with an absolute error of up to 10^{-6}. Each line of the output should end with a line break.

Constraints

- $1 \leq t \leq 20$
- $1 \leq n \leq 10$
- $1 \leq b \leq 10000$
- $1 \leq a_i \leq 10000$ for all $1 \leq i \leq n$
- $a_i \leq a_{i+1}$ for all $1 \leq i \leq n-1$

Sample Explanation

In the first sample, in the first case, there are 3 prizes with values 1, 5 and 10. With a winning probability of 0.5, the expected winning probabilty is 0.5 for the first prize, $0.5^2 = 0.25$ for the second prize and $0.5^3 = 0.125$ for the third prize. The expected payoff is therefore $1 \cdot 0.5 + 5 \cdot 0.25 + 10 \cdot 0.125 = 3$. With cost 3, the total expected payoff is 0.

Sample Input 1

```
5
3 3
1 5 10

5 1
2 3 5 7 11

2 10
1 100

3 10
2 3 5

4 10
1 10 100 10000
```

Sample Output 1

```
Case #1: 0.5000000000
Case #2: 0.2856243953
Case #3: 0.3112672920
Case #4: 1.0000000000
Case #5: 0.1732375136
```

Sample Input 2

```
13
1 3
10

2 1
5 9

3 1
5 8 10

3 9
3 6 8

4 2
1 4 4 7

4 2
4 4 4 5

5 5
1 2 7 7 10

5 9
2 3 5 6 6

6 5
1 1 2 8 8 8

7 7
1 1 1 2 3 3 6

8 4
2 3 4 4 5 7 7 8

9 1
1 2 5 6 8 8 9 10 10

10 3
1 3 3 3 5 6 8 9 10 10
```

Sample Output 2

```
Case #1: 0.3000000000
Case #2: 0.1561249820
Case #3: 0.1544582860
Case #4: 0.7500000000
Case #5: 0.4590767260
Case #6: 0.3347431820
Case #7: 0.6197041759
Case #8: 0.7608172953
Case #9: 0.6652758048
Case #10: 0.8344954285
Case #11: 0.5677976774
Case #12: 0.3554659118
Case #13: 0.5547896661
```