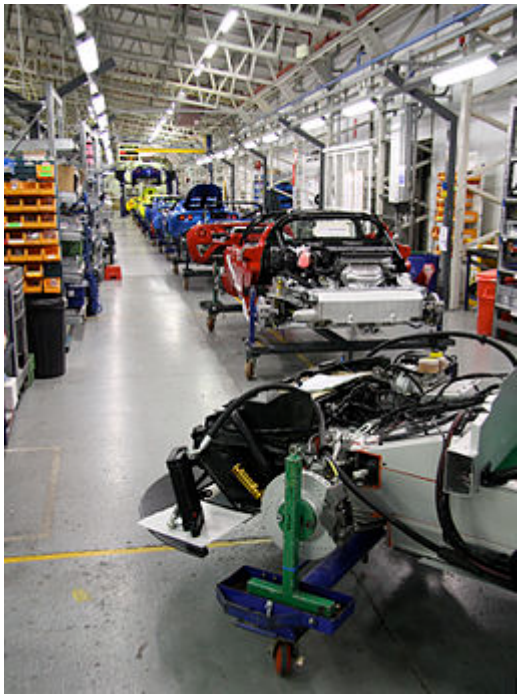
Problem B - In the assembly line

Description

The CEO of a well-known car manufacturer wants to open a new modernized assembly line that produces cars with a top quality construction. An assembly line is designed for the sequential production of cars. It consists of several specialized machines that perform specific assembly operations, such as the installation of the engine or the wheels. It is known that each machine for a given operation has only a certain positive probability of completing the operation successfully, which poses constraints on the final quality of the produced cars.

Given the short budget, the CEO wants to buy machines to each of the operations in the assembly line. In order to maximize the probability of not having a single failure at the end of the assembly process, the CEO wants to buy more than one machine to each operation: if a machine fails, the assembling process is immediately taken by other. Can you help him to decide how many machines should be bought for each operation?



Note : Only implementations that use dynamic programming principles will be considered for the assessment.

Input

Each test case starts with the number of operations (n) that need to be

performed by machines at the assembly line. Then, n lines follow; the i-th line gives the success probability of the machine for operation i as a real positive number in the interval]0,1] as well as the cost of buying that machine as a positive integer. The test case finishes with the value for the budget available as a positive integer.

Output

For each test case, print the maximum probability of not having a single failure at the end of the assembly line in the first row (with 12 digits of precision), as well as number of machines to buy for each operation in the following row. At least one machine must be bought for each operation. You can assume that the budget available is enough to buy at least one machine for each operation. Since there could be many solutions that maximize this probability, print only those that minimize the budget, each per row. The solutions should be sorted in lexicographic order from right to left.

Constraints

- $n \leq 400$

Example

Example input:

```
6
0.01 2
0.01 1
0.01 2
0.01 2
0.01 1
0.01 2
100
```

Example output:

```
0.000000984953
9 16 9 8 16 8
9 16 8 9 16 8
8 16 9 9 16 8
9 16 8 8 16 9
8 16 9 8 16 9
8 16 8 9 16 9
```
