Problem J. 'Ello, and What Are You After, Then?

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

You are playing *RunEscape*. *Slayer* is an activity where you repeatedly complete tasks of the form "slay x monsters of type y". You ponder what is the fastest way to train slayer.

There are n slayer masters you can get tasks from. The i-th slayer master has m_i tasks they can give you. You know the following information about each task:

- f_{ij} — the frequency of the task;
- t_{ij} — how many minutes the task takes to complete;
- e_{ij} — how much XP you gain per minute while doing the task.

When you complete a task you receive c slayer points. When you get a task you can spend s slayer points to skip it. You loop through the following steps:

1. Select a slayer master. Let i be the index of the chosen master.
2. Block up to b of their tasks leaving at least one unblocked. Let B be the set of indices of the tasks you blocked. The probability of receiving the j-th task becomes
$$P(j) = \begin{cases} \frac{f_{ij}}{\sum_{k \notin B} f_{ik}} & \text{if } j \notin B \\ 0 & \text{if } j \in B. \end{cases}$$
3. The slayer master randomly assigns you a task using P. You can either skip the task and lose s points or complete it and receive c points.
4. Go back to step 1.

You start with 0 slayer points. Calculate e, the maximum possible expected XP gain per minute such that you never go below 0 slayer points, assuming you make all of your choices optimally. See the Note section below for a formal definition of e.

Input

The first line of input contains three integers b ($0 \le b \le 3 \cdot 10^4$), c and s ($1 \le c, s \le 10^4$).

The second line of input contains a single integers n ($1 \le n \le 10^3$) — the number of slayer masters. The description of their tasks follows.

The first line for each slayer master contains a single integer m_i ($1 \le m_i \le 3 \cdot 10^4$) — the number of tasks the slayer master has.

The following n_i lines contain three integers f_{ij}, t_{ij} and e_{ij} ($1 \le f_{ij}, t_{ij}, e_{ij} \le 10^4$).

It is guaranteed that the sum of all m_i does not exceed $3 \cdot 10^4$.

Output

Print a single floating-point number: e.

Your answer will be considered correct if its absolute or relative error does not exceed 10^{-6}. Formally, let your answer be a, and the jury's answer be b. Your answer is accepted if and only if
$$\frac{|a - b|}{\max(1, |b|)} \le 10^{-6}.$$

Examples

standard input	standard output
0 1 6 2 1 1 1 1 2 1 10 1 1 10 10	7.000000000000
2 1 2 1 4 10 2 1 10 1 1 1 10 1 1 1 10	5.909090909091

Note

We now define e formally.

Let q be a natural number. Consider going through the loop for exactly q iterations. A *strategy* is a sequence of q tuples (i_k, B_k, t_k). The k-th of those tuples describes your actions on the k-th iteration:

- i_k is the index of the slayer master you will go to on the k-th step.
- B_k is the set of tasks you will block on the k-th step.
- t_k is a function $\mathbb{N}_0 \to 2^{\{1,2,\ldots,m_{i_k}\}}$. $t_k(p)$ describes the set of tasks you will skip if you have exactly p slayer points. If $p < s$, then $t_k(p) = \varnothing$.

For each strategy, its *efficiency* is defined as the expected amount of XP you gain, divided by the expected amount of time it will take. Let e_q be the maximum efficiency among all strategies.

Then

$$e = \lim_{q \to \infty} e_q.$$