TLE '16 Contest 5 P5 - Better Ranking List

The Don Mills Online Judge (DMOJ) is a source of countless debates on how to accurately rank users from best to worst. One measurement, called PP, catches the attention of Nathan.

Better ranking list (December 7, 2016):

1. FatalEagle (649.60)
2. bruce (597.08)
3. d (578.33)
4. jeffreyxiao (568.33)
5. SourSpinach (541.11)
6. kobortor (521.10)
7. thomas0115 (493.93)
8. nullptr (484.38)
9. ZQFMGB12 (483.64)
10. awaykened (477.50)
11. imaxblue (467.72)
12. ImbaCalvin (449.56)
13. pyrexshorts (447.92)
14. NoWoDeYo (413.17)
15. PogChamp (406.38)
16. Butane (401.10)

The users with the highest PP on the DMOJ.

The problems on the DMOJ are numbered from 1 to $100\,000$. When a user submits to a problem, the user will be given a point value (a real number between 0 and $10\,000$ inclusive), but *only the highest point value reached for that problem* will be considered in the rankings. A user will get 0 points from a problem with 0 submissions.

To measure the PP of a user, the first step is to get all of the user's points as a list. Then, this list is sorted in non-increasing order. For simplicity, this sequence will be labelled from s_1 to $s_{100\,000}$.

The actual calculations can now begin, with the result starting at 0, and r $(0 < r \leq 1)$ as a given ratio. In the 1^{st} step, the result increases by $s_1 \cdot r^0$ (it is known that $r^0 = 1$). In the 2^{nd} step, the result increases by $s_2 \cdot r^1$. In general, the result increases by $s_i \cdot r^{i-1}$. This continues until the list is exhausted, and the final result is the user's PP.

In other words, a user's PP is equal to $\sum_{i=1}^{100\,000} s_i \cdot r^{i-1}$.

Although it is nice to sort everyone by PP, Nathan does not want to stop there. He wants to plot a user's PP against the date. Since Nathan's code somehow crashes on a user with N submissions, can you calculate the information for him?

Constraints

In all subtasks, $1 \leq N \leq 200\,000$.

Subtask	Points	Additional Constraints
1	5	$P_i = 1$
2	10	$N \leq 500$ Also, the user will only make submissions to problems in the range $1 \ldots 500$. In other words, $1 \leq P_i \leq 500$.
3	10	$r = 1$
4	20	$r \leq 0.01$
5	30	$N \leq 80\,000$
6	25	No additional constraints.

Input Specification

The first line contains a real number r $(10^{-6} \leq r \leq 1)$, a ratio given to 6 decimal places.

The second line contains N, the number of submissions that a user made.

N lines of input follow. The i^{th} line contains an integer, P_i $(1 \leq P_i \leq 100\,000)$, followed by a real number given to three decimal places, V_i $(0 \leq V_i \leq 10\,000)$. This signifies that on the i^{th} submission, the user submitted to problem number P_i and received a point value of V_i.

Output Specification

The output will consist of N lines. On the i^{th} line, output the user's PP directly after their i^{th} submission.

Your answer will be judged as correct if the absolute or relative error does not exceed 10^{-7}.

Sample Input 1

```
0.500000
5
2 4.000
1 6.000
2 10.000
3 2.000
3 0.000
```

Sample Output 1

```
4.000000000
8.000000000
13.000000000
13.500000000
13.500000000
```

Explanation for Sample Output 1

After the 1^{st} submission, $s_1 = 4$ and the remaining terms are 0. Therefore, the user's PP is $4 \cdot 1 = 4$.

After the 2^{nd} submission, $s_1 = 6$, $s_2 = 4$, and the remaining terms are 0. Therefore, the user's PP is $6 \cdot 1 + 4 \cdot 0.5 = 8$.

After the 3^{rd} submission, problem 2 gets bumped from 4 to 10. As a result, $s_1 = 10$, $s_2 = 6$, and the remaining terms are 0. Therefore, the user's PP is $10 \cdot 1 + 6 \cdot 0.5 = 13$.

After the 4^{th} submission, the user's PP is $10 \cdot 1 + 6 \cdot 0.5 + 2 \cdot 0.25 = 13.5$.

The 5^{th} submission does not change the sequence s, therefore the user's PP stays the same.

Sample Input 2

```
0.010000
8
1 1.000
2 2.000
3 3.000
4 4.000
5 5.000
6 6.000
3 2.000
3 5.000
```

Sample Output 2

```
1.000000000
2.010000000
3.020100000
4.030201000
5.040302010
6.050403020
6.050403020
6.050504020
```