Problem D. Crystals

Input file:	`crystal.in`
Output file:	`crystal.out`
Time limit:	1 second
Memory limit:	64 megabytes
Grading system:	each test is counted

Last generation quantum processors used pairs of crystals of the same characteristics, which are defined by integer number from the interval $-10^9..10^9$. When a crystal is produced, its characteristic is measured in laboratory, then identical crystal is searched in storehouse. If a pair of crystals is found, they sent to a processor production company. Otherwise, newly produced crystal is taken to the storehouse.

Crystal production company is interested in investigating a regularities how crystals got a particular characteristic, and measures various statistics for this purpose. One of these statistics formed as follows: in a given interval of possible characteristics find the longest subinterval of consecutive integers for which there is no crystal in the storehouse.

More formally: in interval $[a, b]$ find the longest subinterval of consecutive integers from x to y ($a \le x \le y \le b$), such that in the storehouse there is no crystal with characteristic z with $x \le z \le y$.

Given information about characteristics in the storehouse, write a program, which answer to queries about various intervals.

Input

The first line of the input contains a single integer N ($1 \le N \le 400\,000$) — total number of the following lines describing input data. Each of the following N lines starts with either «A» or «Q».

Lines starting with «A» will provide details of a newly produced crystal and have a format `A x`, where x — characteristic of a new crystal ($-10^9 \le x \le 10^9$).

Lines starting with «Q» have a format of `Q L R` and represents queries to find in interval $[L, R]$ the number of characteristics in the longest subinterval, which does not contain crystal characteristics, presented in the storehouse.

Output

For each query given in the input file provide a separate line with answer to that query.

Examples

crystal.in	crystal.out
9 A 7 Q -2 30 Q 7 7 A 7 Q -2 30 Q 7 7 A 7 A 25 Q -2 30	23 0 33 1 17

Problem E. Greatest common divisor

Input file:	gcd.in
Output file:	gcd.out
Time limit:	1 second
Memory limit:	64 megabytes
Grading system:	only full solution for subtask receives points

Let us consider a sequences of integers $a_1, a_2, a_3, ..., a_n$ of the length n, $1 \le a_i \le m$. Denote a sequence is *divisible by* x, if the greatest common divisor of its elements is divisible by x. Denote $F(x)$ the number of different sequences of length n, *divisible by* x.

Calculate the number $F(1) + F(2) + ... + F(m)$.

Input

The first line of input contains a single integer $2 \le n \le 10^{100000}$, the second line of input contains a single integer $2 \le m \le 10^9$.

Output

Output the problem answer by modulo $10^9 + 7$.

Examples

gcd.in	gcd.out
3 2	9
2 4	22
3 5	136

Notes

In the second example a sequence consists of two positive integers, which are less or equal to 4. All of them divided by 1, so $F(1) = 4 \cdot 4 = 16$. {2, 2}, {2, 4}, {4, 2}, greatest common divisor of this sequences is 2, {4, 4}, greatest common divisor of this sequence is 4, these sequences form F(2), so $F(2) = 4$. 3 is a divisor of only 3 and 4 is a divisor of only 4, so $F(3) = F(4) = 1$ (sequences {3, 3} and {4, 4}).

Scoring

This task has four subtasks:

1. $2 \le n, m \le 100$. Scored 13 points.
2. $2 \le n, m \le 10^6$. Scored 18 points.
3. $2 \le n, m \le 10^9$. Scored 34 points.
4. $2 \le n \le 10^{100000}$, $2 \le m \le 10^9$. Scored 35 points.

Each of the next subtask will be scored in case of all the previous ones are successfully passed.

Problem F. Magic never happen to be in plenty

Input file:	`magican.in`
Output file:	`magican.out`
Time limit:	2 seconds
Memory limit:	256 megabytes
Grading system:	only full solution for subtask receives points

Every boy in Greatland dreams to have powerful magic word. The longer magic word, the more powerful it is. If there are several equal length, the lexicographic minimum of them is better (more powerful). A sequence of letters (that is, a word) $a_1a_2 \dots a_k$ lexicographically less than $b_1b_2 \dots b_k$ if and only if at the first i where a_i and b_i differ, a_i comes before b_i in the alphabet.

In Greatland, there are N cities and M directed roads between them. Every road on map labeled with one of english letters. Ali-Amir grown up, and now it is time for him to choose magic word. He starts from the city number 1 and proceeds along the roads to other cities. During the passing from a city A to another one, if there is no way back to A, he adds corresponding letter to his magic word. Otherwise, he just goes further.

Ali-Amir wants to go along the path, which gives the longest word. However, even the longest word can be improved, if certain number of first consecutive letters of the respective path initially just to write into the draft, and at the end of the path add them to the end of the word in the same order like in the draft. Among all of these words Ali-Amir wants to find the first in the lexicographical order.

Help Ali-Amir to choose the best magic word in Greatland!

Input

In the first line of input given N and M ($2 \le N \le 100\,000, 1 \le M \le 300\,000$).

In the next M lines given description of each road u, v and c ($1 \le u, v \le n$) — it means road from u to v, labeled with the english lowercase letter.

Output

Print the best magic word Ali-Amir can get.

Examples

magican.in	magican.out
6 7 1 2 a 2 3 a 3 1 a 3 4 a 3 5 b 4 6 z 5 6 a	ab
7 6 1 2 a 2 3 t 3 4 y 4 5 a 5 6 l 6 7 m	almaty

Scoring

This problem has six subtasks:

1. $2 \leq N \leq 100, M = N - 1$. It is guaranteed that it is possible to reach all cities from the first city and each city except for one has one outgoing road. This subtask costs 7 points.

2. $2 \leq N \leq 1000, 1 \leq M \leq 3000$. It is guaranteed, that if there is path from city u to v, then there is no path from v to u. This subtask costs 11 points.

3. $2 \leq N \leq 1000, 1 \leq M \leq 3000$. This subtask costs 24 points.

4. $2 \leq N \leq 100\,000, M = N - 1$. It is guaranteed that it is possible to reach all cities from the first city and each city except for one has one outgoing road. This subtask costs 15 points.

5. $2 \leq N \leq 100\,000, 1 \leq M \leq 300\,000$. It is guaranteed, that if there is path from city u to v, then there is no path from v to u. This subtask costs 17 points.

6. $2 \leq N \leq 100\,000, 1 \leq M \leq 300\,000$. This subtask costs 26 points.