Problem A. Subway Tickets

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

In order to avoid traffic jams, the organizers of the International Olympiad of Metropolises have decided to use Subway for bringing participants from the hotel to the location of the contest. Now the organizers need to buy tickets.

There are n participants of the Olympiad. The organizing committee has a travel card that can be used to pay for at most m rides. They can add rides to the card. There are two options to add rides to the card. It is possible to add 1 ride to the card at a cost of a rubles, or add 4 rides at a cost of b rubles. Any option can be used any number of times (possibly zero).

Help the organizers to get at least n rides on the card, spending minimum amount of money.

Input

The first line of the input file contains integer m ($1 \le m \le 10^{15}$) — the initial number of rides on the travel card.

The second line contains one integer n ($1 \le n \le 10^{15}$) — the number of participants of the Olympiad.

The third line contains one integer a ($1 \le a \le 1000$) — the cost of adding 1 ride to the card.

The fourth line contains one integer b ($1 \le b \le 1000$) — the cost of adding 4 rides to the card.

Output

Output one integer number — the minimum cost of adding rides to the card, so that it will have at least n rides.

Examples

standard input	standard output
1 3 1 10	2
2 4 9 10	10
3 8 9 10	19

Note

In the first example it is optimal to add 1 ride twice.

In the second example it is better to add 4 rides to the travel card (only 2 of them will be used).

In the third example it is optimal to add 1 ride once and add 4 rides once.

Scoring

Tests for this problem are divided into four groups. For each of the groups you earn points only if your solution passes all tests in this group and all tests in all **previous** groups.

Group	Tests	Points	Additional constraints	Comments
			n, m	
0	1 – 3	0	–	Sample tests.
1	4 – 23	19	$n, m \leq 1000$	
2	24 – 43	27	$n, m \leq 1\,000\,000$	
3	44 – 63	54	–	

Problem B. Show business

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

Evgeni is a rising star of pop music. At the beginning of the new season he wants to schedule his performances at different music festivals of his city. However, things are not that simple in the show business world.

Evgeni knows that there will be n festivals this season. The i-th festival will take place from day l_i to day r_i, inclusive. **Festivals do not overlap**.

Evgeni wants to maximize the number of festivals that he will take part in at least once. There is an additional limitation: according to Evgeni's contract, he is given some value d and has to choose some day x ($x \geq 0$). Then he will be able to take part in festivals only on days x, $x + d$, $x + 2d$ and so on. Note, that he may skip some of these days, i.e. he is not required to use all these days to take part in the festivals.

Evgeni may pick any of the days x. He wants to maximize the number of festivals he will be able to participate at least once. If there are several optimal values x, he wants to pick the minimum of them.

Input

The first line of the input contains two integers n and d ($1 \leq n \leq 100\,000$, $1 \leq d \leq 10^{15}$) — the number of festivals and the value of difference between two consecutive days that Evgeni is allowed to use for performance at festivals.

Each of the next n lines defines the time range of some festival: the i-th of them contains two integers l_i and r_i ($0 \leq l_i \leq r_i \leq 10^{15}$) — the number of the first and the last day of the i-th festival.

Output

Print two integers: the maximum number of festivals Evgeni can take part in and the day his first performance should happen in order to achieve this number of participations. Days are numbered starting with zero. If there are many optimal days to start, print the minimum one.

Examples

standard input	standard output
4 4 5 5 0 2 15 23 10 11	3 1
2 5 8 9 16 18	2 3

Note

In the first sample, if Evgeni starts on day 1 or day 2, he will be able to participate in three festivals. If he starts on day 1, he will be able to participate in festival 1 (day 5), festival 2 (day 1) and festival 3 (days 17 and 21). If he starts on day 2, he will be able to participate in festival 2 (day 2), festival 3 (days 18 and 22) and festival 4 (day 10).

In the second sample Evgeni can take part in both festivals if he starts on day 3 or day 8.

Scoring

Tests for this problem are divided into four groups. For each group you earn points only if your solution passes all tests of this group and all tests of all previous groups.

Groups	Tests	Points	Additional constraints			Comments
			n	d	l_i, r_i	
0	1 – 2	0	–	–	–	Sample tests.
1	3 – 25	21	$n \le 1\,000$	$d \le 1\,000$	$l_i, r_i \le 1\,000\,000$	
2	26 – 43	30	$n \le 1\,000$	$d \le 1\,000$	–	
3	44 – 62	49	–	–	–	

Problem C. "All for $O(1)$" shop

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

There are two cashiers in the shop "All for $O(1)$". You have to simulate the behaviour of the customers by the log of the shop workflow.

You are given the description of the following events in the chronological order:

- `a` — new customer comes to the end of the line to the first cashier;
- `b` — new customer comes to the end of the line to the second cashier;
- `A` — first cashier serves one customer;
- `B` — second cashier serves one customer;
- `>` — first cashier closes;
- `]` — second cashier closes;
- `<` — first cashier opens;
- `[` — second cashier opens.

Any time some cashier closes, each person in the line to this cashier goes to another cashier in reverse order. Persons move one by one starting from the last. That means the person that was the last one in this line leaves first and goes to the other line, then goes person that was standing before the last person and so on. At the end, person that was first in the line that just closed becomes the last one in another line.

When a cashier opens, people from the other line (again, in reverse order starting with the last one) leave and go to the new line if their position in it will be strictly smaller than their position in the line they are now.

The list of events is correct if:

- Only closed cashiers may open;
- Only open cashiers may close;
- No customer comes to the line to the closed cashier;
- No cashier tries to serve the customer if it is closed;
- No cashier serves the customer if the corresponding line is empty;
- At any moment of time at least one cashier works.

Customers are numbered starting with 1 in order they appear in the event log. At the beginning both cashiers are open and both lines are empty.

Input

The first line of the input contains an integer n ($2 \le n \le 10\,000\,000$) — the number of events. The second line contains n characters that define events as described above.

It's guaranteed that the list is correct and the input will contain at least one serve query.

Output

Print one line: for each event of serving a customer print the last digit of his index without any blanks.

Examples

standard input	standard output
`15` `aaabA>bBBb<BBAa`	`143256`
`12` `aaaaa><AABBB`	`12543`

Note

Explanation of the first sample:

No.	Team	Explanation	Line 1	Line 2
01	`a`	Customer 1 comes to the line 1	1	∅
02	`a`	Customer 2 comes to the line 1	1, 2	∅
03	`a`	Customer 3 comes to the line 1	1, 2, 3	∅
04	`b`	Customer 4 comes to the line 2	1, 2, 3	4
05	`A`	Customer 1 served in line 1	2, 3	4
06	`>`	Cashier 1 closes	∅	4, 3, 2
07	`b`	Customer 5 comes to the line 2	∅	4, 3, 2, 5
08	`B`	Customer 4 served in line 2	∅	3, 2, 5
09	`B`	Customer 3 served in line 2	∅	2, 5
10	`b`	Customer 6 comes to the line 2	∅	2, 5, 6
11	`<`	Cashier 1 opens	6	2, 5
12	`B`	Customer 2 served in line 2	6	5
13	`B`	Customer 5 served in line 2	6	∅
14	`A`	Customer 6 served in line 1	∅	∅
15	`a`	Customer 7 comes in to the line 1	7	∅

Scoring

Tests for this problem are divided into four groups. For each group you earn points only if your solution passes all tests of this group and all tests of all previous groups.

Groups	Tests	Points	Additional constraints n	Comments
0	1 – 2	0	–	Sample tests.
1	3 – 28	26	$n \le 1000$	
2	29 – 47	37	$n \le 200\,000$	
3	48 – 63	37	–	

Problem D. Advertisement

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	128 megabytes

Company BubbleGum plans to launch its new version of world famous operation system BubbleGum OS 10. The director of the company is the billionaire Bubble Gum. He wants all devices of the popular company Phony to have his new OS pre-installed.

Last night Bubble Gum got an idea to place a huge ad in the city, such that the director of the Phony Fi Lin will see it on his way to work and back as many times as possible. Bubble Gum already knows the route of Fi Lin for tomorrow, so it's now time to find out the best position for the ad.

If we look on the city from the height, its road system will look like an infinite uniform grid on the coordinate plane. All roads that go from north to south are numbered from west to east with consecutive integers, same as roads that go from west to east are numbered from south to north. Thus, there is exactly one pair of roads intersecting at any integer point (x, y). Fi Lin uses his personal driver, starts at point $(0, 0)$ and moves to the north. While he is in the car he always looks rightward in respect to the direction of movement (for example, if the car is moving north, than Fi Lin is looking eastward, while if the car is moving west, than Fi Lin is looking northward). This means, he sees all the objects that are located on the ray of his sight. Moreover, when the car is making a turn, Fi Lin is able to see all the objects located in the corresponding angle.

You are given n records of form a_i, c_i, where a_i is the length of the corresponding segment of movement, c_i is equal to '`L`' if the car turns to the left at the end of the segment and to '`R`' if it turns right. Records are numbered from 1 to n. It's guaranteed that at the end of the route the car gets back to the initial point and head to the north. As Fi Lin is a very busy man, the route might look strange, cross itself many times and even use the same road segment multiple times (possibly, in different directions).

Ad is planned to be positioned at the center of some square of the grid. It will be big enough to be seen from any distance. Find the way to place an ad in such a way that Fi Lin will see it as many times as possible during his travel.

Input

The first line of the input contains an integer n ($4 \le n \le 100\,000$) — the number of segments of Fi Lin's car movement.

Each of the next n lines contains a pair of integer a_i ($1 \le a_i \le 2 \cdot 10^9$) and a character c_i. It's guaranteed that the route matches the above description and that during his travel Fi Lin was always on the streets with absolute values **strictly less** than s. Note that the value of s is not given in the input and depends on the group number.

Output

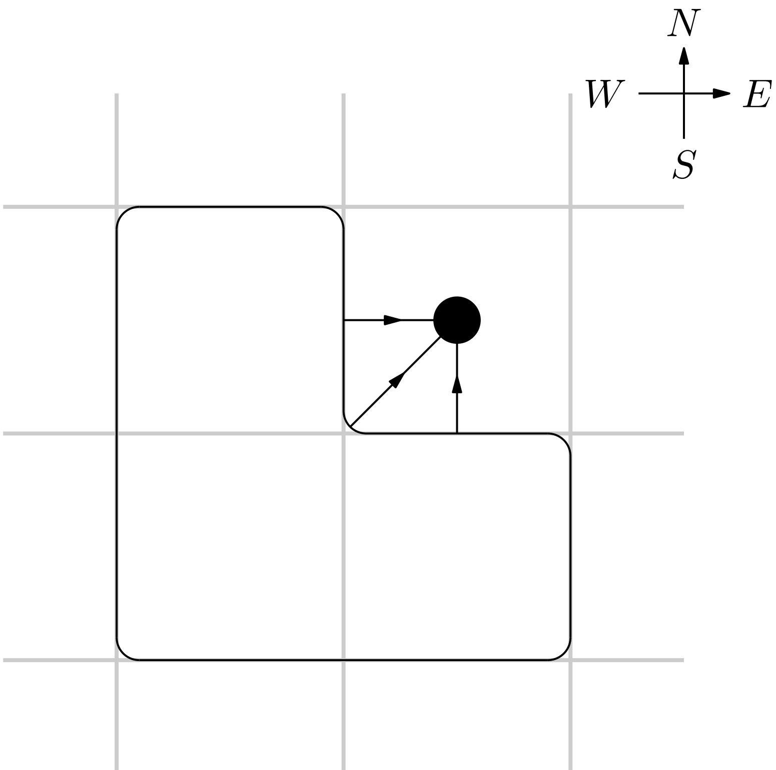
Print two integers x and y, defining the square of the grid that should be used to possess the ad. Integer x should be equal to the number of road to the west of the ad, while y should be equal to the number of the road to the south. Both integers x and y should not exceed 10^9 by their absolute value. If there are several optimal answers, any of them is accepted.

Examples

standard input	standard output
4 1 L 1 L 1 L 1 L	1 0
6 1 L 1 R 1 L 1 L 2 L 2 L	-1 1

Note

In the first sample, Fi Lin will see the ad only once. In the second sample, Fi Lin will see the ad three times: while the car drives through segments 2 and 3, and while turning from segment 2 to segment 3.



Scoring

Tests for this problem are divided into six groups. For each group you earn points only if your solution passes all tests of this group and all tests of all previous groups.

Groups	Tests	Points	Additional constraints		Comments
			n	s	
0	1 – 2	0	—	—	Sample tests.
1	3 – 21	20	$n \le 100$	$s \le 100$	
2	22 – 28	13	$n \le 100$	$s \le 10^9$	
3	29 – 37	20	$n \le 5\,000$	$s \le 10^9$	
4	38 – 46	9	$n \le 20\,000$	$s \le 10^9$	
5	47 – 55	38	–	$s \le 10^9$	