TRAM

Seats in a new tram operating in Zagreb are organized into a grid consisting of N rows numbered 1 through N and two columns numbered 1 and 2. The distance between two seats, one at row R_A, column C_A and other at row R_B, column C_B is the Euclidean distance between the centers of the corresponding grid squares – namely $\sqrt{(R_A - R_B)^2 + (C_A - C_B)^2}$.

Most passengers prefer solitude when using public transportation and they always try to choose a seat that is as far away from other passengers as possible. More precisely, when a passenger enters the tram he or she will choose a free seat whose **distance from the closest occupied seat is the highest possible**. If there is more than one such seat, they will always choose one with the lower row number and if there is still more than one such seat, they will choose the one with the lower column number. After the passenger chooses a seat, he or she will sit there until leaving the tram. If the tram is empty, the next passenger to enter will always choose the seat in row 1 and column 1.

TASK

Write a program that will, given a sequence of events, each event either a passenger entering or leaving the tram, determine where each of the passengers was sitting. The tram is initially empty.

There are M events in the input numbered 1 through M in the order in which they occurred. There are two kinds of events: event of type 'E' corresponds to a passenger entering the tram, while the event of type 'L' corresponds to a passenger leaving the tram. For an event of type 'L', an integer P is also given – it specifies that the passenger leaving in this event is the one that entered at event P.

Test data will be such that there will always be at least one free seat in the tram whenever a passenger is trying to enter.

INPUT

The first line of input contains two integers N and M ($1 \leq N \leq 150\,000$, $1 \leq M \leq 30\,000$), the number of rows in the tram and the number of events. The following M lines contain the description of the events, K-th of those M lines contains the description of event K – either the character 'E', or the character 'L' followed by a single space and the integer P_K ($1 \leq P_K < K$). Each P_K will be valid– event P_K is of type 'E' and no passenger will try to leave twice.

OUTPUT

The number of lines in the output should be equal to the number of events of type 'E' in the input. For each event of type 'E', in the order in which they occurred, output on a single line the row and the column number of the seat chosen by the passenger, separated by a single space.

GRADING

- In test cases worth a total of 25 points, it holds N ≤ 150 and M ≤ 150.
- In test cases worth a total of 45 points, it holds N ≤ 1500 and M ≤ 1500.
- In test cases worth a total of 65 points, it holds N ≤ 150 000 and M ≤ 1500.

DETAILED FEEDBACK WHEN SUBMITTING

During the contest, you may select up to 50 submissions for this task to be evaluated on a part of the official test data. When the results are ready, a summary of the results will be available on the contest system.

EXAMPLES

input	input	input
3 7	13 9	10 9
E	E	E
E	E	E
E	E	E
L 2	E	E
E	E	L 3
L 1	E	E
E	E	E
	E	L 6
	E	E
output	E	
		output
1 1	**output**	
3 2		1 1
1 2	1 1	10 2
3 1	13 2	5 2
1 1	7 1	7 1
	4 2	4 2
	10 1	2 2
	2 2	4 1
	3 1	
	5 1	
	6 2	