

Your government has finally solved the problem of universal health care! Now everyone, rich or poor, will finally have access to the same level of medical care. Hurrah!

There's one minor complication. All of the country's hospitals have been condensed down into one location, which can only take care of one person at a time. But don't worry! There is also a plan in place for a fair, efficient computerized system to determine who will be admitted. You are in charge of programming this system.

Every citizen in the nation will be assigned a **unique** number, from 1 to P (where P is the current population). They will be put into a queue, with 1 in front of 2, 2 in front of 3, and so on. The hospital will process patients one by one, in order, from this queue. Once a citizen has been admitted, they will immediately move from the front of the queue to the back.

Of course, sometimes emergencies arise; if you've just been run over by a steamroller, you can't wait for half the country to get a routine checkup before you can be treated! So, for these (hopefully rare) occasions, an expedite command can be given to move one person to the front of the queue. Everyone else's relative order will remain unchanged.

Given the sequence of processing and expediting commands, output the order in which citizens will be admitted to the hospital.



Input

Input consists of at most ten test cases. Each test case starts with a line containing P , the population of your country ($1 \leq P \leq 1000000000$), and C , the number of commands to process ($1 \leq C \leq 1000$).

The next C lines each contain a command of the form 'N', indicating the next citizen is to be admitted, or 'E x ', indicating that citizen x is to be expedited to the front of the queue.

The last test case is followed by a line containing two zeros.

Output

For each test case print the serial of output. This is followed by one line of output for each 'N' command, indicating which citizen should be processed next. Look at the output for sample input for details.

Sample Input

```
3 6
N
N
E 1
N
N
N
10 2
N
N
0 0
```

Sample Output

```
Case 1:
1
2
1
3
2
Case 2:
1
2
```