

Han didn't want to study solo so he invited his friend Dominik to come over. After an eventful evening that will be remembered for a record number of solved tasks from the field of electronics, Dominik went home. To his surprise, the police stopped him thinking he was **drunk**. It is known that in these situations sobriety is proven by solving a series of carefully crafted tasks that test a man's cognitive abilities. If we can trust Dominik, the conversation went something like this:

Policeman: Something easy to begin with... What is the complexity of bubble sort?

Dominik: That is really easy, $O(n^2)$.

Policeman: Say the English alphabet in reverse.

Dominik: Trivial, zyxwvutsrqponmlkjihgfedcba

Policeman: You learned that by heart. Now imagine that all the letters of the English alphabet from 'a' to 'z' are respectively written clockwise in a circle. Begin with the letter 'a' and say the letters clockwise. After each spoken letter, I can tell you to continue saying the alphabet in reverse order or I can ask you how many times so far you've said a certain letter. Are you ready? 3, 2, 1, Go!

Dominik: Um... a, b, c...

Write a programme that solves Dominik's problem.

INPUT

The first line of input contains the integer Q ($1 \leq Q \leq 100\,000$), the number of policeman's orders. Each of the following Q lines contains one of the policeman's order in the form of "**SMJER n**" (Croatian for direction) or "**UPIT n x**" (Croatian for query). The order in the form "**SMJER n**" means that, after the n th spoken letter, Dominik must start saying the alphabet in reverse, whereas the order in the form "**UPIT n x**" means that Dominik must say how many times so far he's said the letter x in the first n spoken letters.

The policeman's order will be given chronologically in the input, or, the numbers n ($1 \leq n \leq 10^9$) from the orders will be strictly ascending. The character x from the order in the form of "**UPIT n x**" is a lowercase letter of the English alphabet.

OUTPUT

For each order in the form of "**UPIT n x**", output how many times Dominik has said the letter x in the first n spoken letters. The answer to each query needs to be written in a separate line, and the queries need to be answered in the order given in the input.

SCORING

In test cases worth 40% of total points, it will additionally hold: $N \leq 1000$.

In test cases worth 60% of total points, it will additionally hold: $N \leq 10^5$.

SAMPLE TESTS

<p>input</p> <p>5 UPIT 1 b UPIT 3 b SMJER 4 UPIT 7 a UPIT 10 z</p> <p>output</p> <p>0 1 2 1</p>	<p>input</p> <p>5 SMJER 1 SMJER 2 SMJER 3 UPIT 5 a UPIT 7 w</p> <p>output</p> <p>2 1</p>	<p>input</p> <p>4 UPIT 100 a UPIT 200 c UPIT 300 a UPIT 400 b</p> <p>output</p> <p>4 8 12 16</p>
---	--	--

Clarification of the first example: Dominik says the following letters: a, b, c, d, c, b, a, z, y, x.

Clarification of the second example: Dominik says the following letters: a, z, a, z, y, x, w.