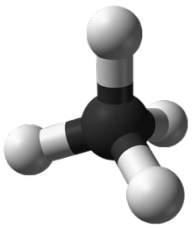
Chemistry



Ryan is a chemistry lecturer in NUS. His daily job is to help students understand useful and important chemistry concepts and apply them in practice. He also sets up exam questions for the modules that he teaches. As we all know, setting exam questions is hard and requires a lot of effort.

When designing the final exam for one of his modules, CM1020, Ryan needs a lot of chemical formulas and their mass. A chemical formula represents a molecule and Ryan is interested in knowing the mass for that molecule fast enough (and accurately, of course) so that he can focus on designing questions rather than computing the mass of the molecule (after all, he wants to put the mass in the question so that the students do not need to compute it themselves).

To count the mass of a chemical formula, Ryan needs to sum up the mass of all atoms in the molecule. A few examples of the calculation are illustrated in the following table:

No.	Formula	Calculation	Mass
1	CH_4	1 x C + 4 x H = 1 x 12 + 4 x 1	**16**
2	$N(CH_2CH_2N(CH_3)_2)_3$	1 x N + 3 x (2 x C + 4 x H + 1 x N + 2 x (1 x C + 3 x H)) = 1 x 14 + 3 x (2 x 12 + 4 x 1 + 1 x 14 + 2 x (1 x 12 + 3 x 1))	**230**
3	$(((H_2)(C_2)))$	2 x H + 2 x C = 2 x 1 + 2 x 12	**26**

Note: We assume that the mass of C is 12, H is 1, and N is 14

Ryan approaches Teddy, his colleague and computer science lecturer in SoC, NUS, to develop this simple mass calculator. Teddy agrees to help him build this simple calculator under the following conditions:

1. The system only takes single-character elements. Elements such as **C**, **N, and O** are valid while elements such as **Na**, **Pb**, and **Li** are not.
2. Numbers are single digits and can only appear after an element or after a closing parenthesis. It cannot appear after another number (hence there are no “10” or “11”).
3. An opening parenthesis will always be validly matched with a closing parenthesis. There are no limits to the number of parentheses as long as it is valid, i.e. (((H2)(C2))) is a valid input.
4. The system does not care whether the formula provided is a valid chemical formula. It just calculates the mass based on the formula given.

Ryan agrees and let Teddy does his work. However, Ryan has a tight deadline and wants the system to be finished in one week. Being busy himself, Teddy asks you, his student, to help him design and build this system together. Help Teddy (and Ryan) finish this simple calculator so that Ryan can continue to focus on designing his exam questions.

Input

The first line of the input consists of a single integer, **N (1 <= N <= 26)**, the number of different elements used in the input. The following **N** lines each contains a chemical element and its respective mass (in integer) separated by a single space.

The next line consists of a single chemical formula consisting of only uppercase English letters (‘A’-‘Z’), numbers, and parentheses. The length of the chemical formula is at most 100,000 (one-hundred thousand) characters.

Output

Print the mass of the given chemical formula. Your output should contain a newline character.

Sample Input 1

```
2
C 12
H 1
CH4
```

Sample Output

```
16
```

Sample Input 2

```
3
C 12
N 14
H 1
N(CH2CH2N(CH3)2)3
```

Sample Output

```
230
```

Skeleton

You are given the skeleton file `Chemistry.java`.

Notes

1. You must either use **stack** or **queue** to solve this problem, whichever is suitable.
2. Your program might give out the correct answer, but killed on CodeCrunch. It means that your program is not efficient enough (i.e. it runs too slow). You should design a more efficient algorithm to solve this problem if this is the case. Consider the above note as a hint.