

7462 Social Network

Raymond is a big data analyst. He constructs a social network G from n persons numbered $1, 2, \dots, n$. For two groups of persons G_1 and G_2 , he utilizes the following two rules to construct the network:

Rule 1: If two persons $p_1 \in G_1$ and $p_2 \in G_2$ have a common interest, then G_1 and G_2 can be merged into one group by connecting every person in G_1 to all the persons in G_2 . This construction is called the join operation, denoted by $G_1 \otimes G_2$. For each person $p_1 \in G_1$ and $p_2 \in G_2$, we say that p_1 and p_2 have a connecting relation after applying Rule 1.

Rule 2: If any two persons $p_1 \in G_1$ and $p_2 \in G_2$ have no common interest, then no connecting relation is created between G_1 and G_2 . This construction is called the union operation, denoted by $G_1 \cup G_2$.

According to the above two rules, Raymond constructs a social network using the following recursive manner:

- (1) A social network can contain only one person;
- (2) If G_1 and G_2 are social networks, then $G = G_1 \otimes G_2$ is a social network;
- (3) If G_1 and G_2 are social networks, then $G = G_1 \cup G_2$ is a social network.

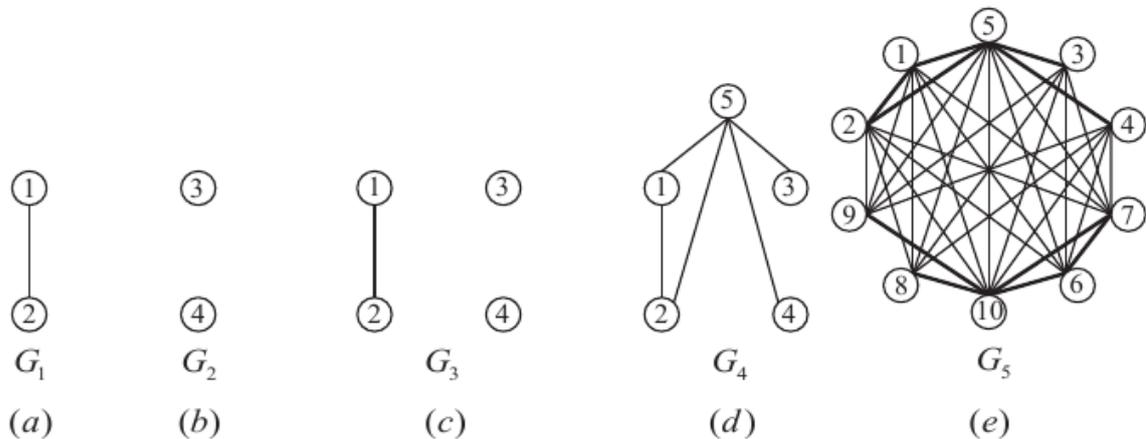


Figure 1: The construction of a social network.

For ease of input, we use ‘J’ to represent the join operation \otimes , and use ‘U’ to represent \cup the union operation. A social network can be represented by a construction expression. For example, consider the following construction expression

$$((((1)J(2))U((3)U(4)))J(5))J((((6)J(7))U((8)U(9)))J(10))$$

The corresponding social network can be constructed as follows. In Step 1, $G_1 = (1)J(2)$ and $G_2 = (3)U(4)$ are built (Figure 1(a) and Figure 1(b)). In Step 2, $G_3 = ((1)J(2))U((3)U(4))$ is built by applying the union operation to G_1 and G_2 (Figure 1(c)). In Step 3, $G_4 = (((1)J(2))U((3)U(4)))J(5)$ is built by applying the join operation to G_3 and Person 5 (Figure 1(d)). In Step 4, $G'_4 = (((6)J(7))U((8)U(9)))J(10)$ is built, where G'_4 has the same structure with that of G_4 .

In Step 5, $G_5 = (((1)J(2))U((3)U(4)))J(5))J((((6)J(7))U((8)U(9)))J(10))$ is built by applying the join operation to G_4 and G'_4 (Figure 1(e)).

After constructing the network, Raymond would like to assign all the persons in the network to nature numbers (this weighting is used for some special survey). For each person i , we use $r(i)$ to denote the number assigned to i .

The assignment must satisfy the following property that for **every** path between two persons, i and j , having the same number (i.e., $r(i) = r(j)$) in the network, there exists at least one person k (on that path) whose number is higher than $r(i) (= r(j))$, that is $r(k) > r(i) = r(j)$. An assignment is *optimal* if the largest used number is the smallest among all assignments. Moreover, the largest used number in an optimal assignment is called the *assignment number*. Figure 2 illustrates an optimal assignment of the social network shown in Figure 1(e). The assignment number equals 8.

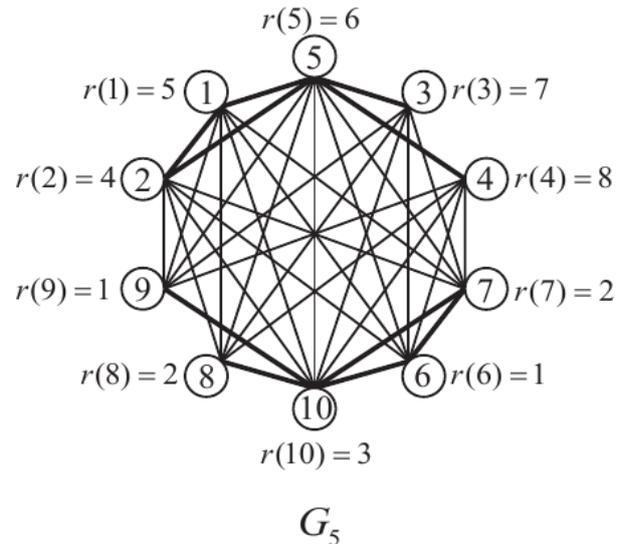


Figure 2: An optimal assignment of a social network.

Given a construction expression corresponding to a social network G , your task is to write a computer program to compute the assignment number of G .

Technical Specification

1. $2 \leq n \leq 10000$.
2. The number of operations is exactly $n - 1$, which constructs one social network from n persons.

Input

The first line of the input file contains an integer indicating the number of test cases to follow. Each test case consists of two lines: the first line contains an integer n and the second line contains a construction expression, where the construction expression consists of digits, alphabets ‘J’ and ‘U’, and parentheses indicating the order of construction. Formally, a construction expression, denoted by Exp , can be recursively defined as follows: Exp can contain only one positive integer i , where $1 \leq i \leq n$. For each Exp containing more than one positive integer, it has the form: $Exp = (Exp)J(Exp)$ or $Exp = (Exp)U(Exp)$. Note that the most outer part of construction expression will not be enclosed by redundant parentheses. For each test case, each number between 1 and n appears exactly once in the construction expression.

Output

The output contains one line for each test case. Each line contains the assignment number of the corresponding social network.

Sample Input

```
2
10
(((1)J(2))U((3)U(4)))J(5))J(((6)J(7))U((8)U(9)))J(10))
10
(((1)J(2))J((3)J(4)))J(5))J(((6)J(7))J((8)J(9)))J(10))
```

Sample Output

8
10