

Problem B. Brackets and Reverse String

Input file: *standard input*
Output file: *standard output*
Time limit: 1 second
Memory limit: 1024 mebibytes

A correct parenthesis sequence is a string that satisfies any of the following conditions:

- It is an empty string.
- It is formed by concatenating ‘(’, A , ‘)’ in this order where A is a correct parenthesis sequence.
- It is formed by concatenating A and B where A and B are correct parenthesis sequences.

You are given a string s of length n consisting of characters ‘(’ and ‘)’.

For each i where $0 \leq i \leq n$, define t_i as the string obtained by concatenating the suffix of s of length $n - i$ and the reversed prefix of s of length i , in this order. That is, if we denote the i -th character of s as s_i , the string t_i is formed by arranging the characters $s_{i+1}, s_{i+2}, \dots, s_n, s_i, \dots, s_2, s_1$ in sequence.

For each t_i where $0 \leq i \leq n$, solve the following problem. Consider an operation where you replace one character in t_i with either ‘(’ or ‘)’. Find the minimum number of such operations required to make t_i a correct parenthesis sequence.

Input

The first line of the input contains one integer n ($2 \leq n \leq 2 \cdot 10^5$, n is even).

The second line contains a string s of length n consisting only of ‘(’ and ‘)’.

Output

Print $n + 1$ lines. On line $i + 1$, print the answer for t_i .

Examples

<i>standard input</i>	<i>standard output</i>
4 () ()	0 2 2 2 2
6))) (((4 2 2 0 0 0 0
8) () () ((3 1 3 1 1 1 1 1 1