A regular bracket-sequence is a string of characters consisting only of opening and closing brackets, and satisfying the following conditions:

- An empty string is a regular bracket-sequence.
- If A is a regular bracket-sequence, then (A), [A] and {A} are also regular bracket-sequences.
- If A and B are regular bracket-sequences, then AB is also a regular bracket-sequence.

For example, the sequences [({})], []{} i [{}]{}[{}] are regular, but the sequences [({{{[, []({} and [{}]){[{}] are not.

Ivica has found a string which looks like it could be a regular bracket-sequence. Some of the characters have become smudged and illegible, and could have been any character.

Write a program that calculates how many ways the illegible characters in the string can be replaced by brackets so that the result is a regular bracket-sequence. This number can be very large, so output only its **last 5 digits**.

Input

The first line contains an even integer N ($2 \leq N \leq 200$), the length of the string.

The second line contains the string. Illegible characters are represented by the '?' character.

Output

Output the number of regular bracket-sequences the string could have read.

Sample test data

input	**input**	**input**
6 ()()()	10 (?([?)]?}?	16 ???[???????]????
output	**output**	**output**
1	3	92202

In the second example, the three matching regular bracket-sequences are `({([()])})`, `()([()]{})` and `([([])]{})`.