

---

## Problem Wimbledon. Unusual competitions

Input file: `input.txt` or standard input  
Output file: `output.txt` or standard output  
Time limit: 1 second  
Memory limit: 512 megabytes

The teacher gave Dmitry's class a very strange task — she asked every student to come up with a sequence of arbitrary length, consisting only of opening and closing brackets. After that all the students took turns naming the sequences they had invented. When the Dima's turn come, he suddenly realized that all his classmates got the right bracketed sequence, and whether he got the right bracketed sequence, he did not know.

Dima suspects now that he simply missed the word “right” in the task statement, so now he wants to save the situation by modifying his sequence slightly. More precisely, he can arbitrary amount of times (possibly zero) perform the *reorder* operation. The reorder operation consists of choosing an arbitrary subsegment of the sequence and then reordering all the characters in it in an arbitrary way. Such operation takes  $l$  nanoseconds, where  $l$  is the length of the subsegment being reordered. It's easy to see that reorder operation doesn't affect the number of opening and closing brackets doesn't change.

Since Dima will soon have to answer, he wants to make his sequence right as fast as possible. Help him to do this, or determine that it's impossible.

### Input

The first line contains a single integer  $n$  ( $1 \leq n \leq 10^6$ ) — the length of Dima's sequence.

The second line contains string of length  $n$ , consisting of characters “(” and “)” only.

### Output

Print a single integer — the minimum number of nanoseconds to make the sequence right or “-1” if it is impossible to do so.

### Examples

input	output
8 ) ) ( ( ( ) (	6
3 ( (	-1

### Note

A bracketed sequence is called right if by inserting “+” and “1” you can get a well-formed mathematical expression from it. For example, sequences “( ) ( )”, “( )” and “( ( ) ( ) )” are right, while “( ) (”, “( ( )” and “( ( ) ) (” are not.

In the first example we can firstly reorder the segment from first to the fourth character, replacing it with “( ) ( )”, the whole sequence will be “( ) ( ) ( )”. And then reorder the segment from the seventh to eighth character, replacing it with “( )”. In the end the sequence will be “( ) ( ) ( ) ( )”, while the total time spent is  $4 + 2 = 6$  nanoseconds.

---