7454 Parentheses

A bracket is a punctuation mark, which is used in matched pairs, usually used within articles or programs. Brackets include round brackets, square brackets, curly brackets, angle brackets, and various other pairs of symbols. Let's focus on the round brackets, also called *parentheses*.

A sequence of parentheses is said to be *well-formed* if the parentheses are properly nested. For example, $A = a_1a_2 \ldots a_{18} = $ "(()())()()()(())()" is well-formed, but $B = b_1b_2 \ldots b_{18} = $ "(()())))((((())((" is not. (See Figure 1.) More formally, a sequence of parentheses $P = p_1p_2 \ldots p_n$ is well-formed if

(1) when scanning it from p_1 to p_n, the number of right parentheses does not exceed the number of left parentheses at any state, and

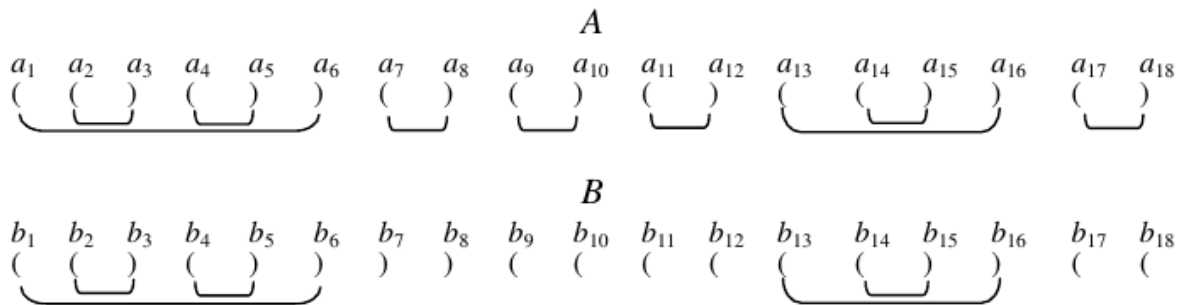(2) the numbers of left and right parentheses are equal.

Figure 1. Two sequences of parentheses.

AutoText is a company, which is developing a text editor for programmers. The new editor will provide many powerful functions to automatically correct typing errors. On a keyboard, the left and right parentheses are adjacent. Thus, it is often that ")" is mistyped as "(" or vice versa. And therefore, one of the functions AutoText wants to provide is to automatically convert a sequence of parentheses P (that may not be well-formed) into a wellformed sequence P'. In the conversion, the only allowed operation is to reverse a parenthesis (i.e., either to replace a "(" with a ")" or to replace a ")" with a "("). For example, in Figure 1, we can convert B into the well-formed sequence A by performing 4 reverse operations on b_7, b_{10}, b_{12}, b_{18}. Of course, there may be several ways to convert a sequence into a well-formed sequence. A conversion is optimal if it uses the minimum number of reverse operations.

Please write a program to compute the minimum number of reverse operations that make a given sequence of parentheses $P = p_1p_2 \ldots p_n$ well-formed.

Input

The first line contains an integer $T \leq 10$ indicating the number of test cases. The first line of each test case contains an even integer n, $2 \leq n \leq 100$, indicating the length of P. Next, the second line gives the sequence P.

Output

For each test case, output the minimum number of reverse operations that make P well-formed.

Sample Input

```
3
18
(()()))))((((((())((
2
()
8
(())))()(
```

Sample Output

```
4
0
2
```