

BRACKETS: Brackets

题目描述

合法的括号序列是非空的仅包含“(”和“)”的字符串。使得你可以通过反复删除相邻的“()”清空整个字符串。

例如，“(())”和“()((()))”是合法的括号序列。“)()”和“(())”不是。

Mike 拥有一个合法的括号序列。Mike 十分喜欢这个序列，但唯一的缺点是它太长了。为了完成替换，需要引入下面的函数 $F(S)$ 。

```
FUNCTION F( S - a valid parentheses sequence )
BEGIN
balance = 0
max_balance = 0
FOR index FROM 1 TO LENGTH(S)
BEGIN
if S[index] == '(' then balance = balance + 1
if S[index] == ')' then balance = balance - 1
max_balance = max( max_balance, balance )
END
RETURN max_balance
END
```

$F(S)$ 等于 S 的最大 **balance**。

Mike 将寻找一个长度最短的序列 B ，来替换当前的序列 A 。要求满足 $F(A) = F(B)$ 。如果有多解，返回字典序最小的。

输入格式

输入数据的第一行包含一个整数 T ，表示数据组数。接下来的 T 行，每行包含一个非空的仅包含“(”和“)”的字符串 A 。

输出格式

对于每组数据，输出对应的答案。

数据范围

- $1 \leq T \leq 5$
- $1 \leq |A| \leq 10^5$

样例数据

输入

1

() ((() ()))

输出

((()))

Problem Setter: Kostya Sokol

Problem Tester: Roman Rubanenko

Translated by : Minako Kojima