Fully Parenthesized Expression

Clyde has written a program that can evaluate arithmetic expressions. There is just one problem - the expressions must be fully parenthesized!

Help Clyde by making sure all of his expressions are fully parenthesized - he's willing to reward you.

Input

The first line of input contains a single integer n (1 ≤ n ≤ 1000) that indicates the amount of test cases.
On the next n lines will be a string s representing the expression. S will always contain a valid expression and will be strictly less than 256 characters in length.

The operators used in s are */% of high precedence and +- of low precedence. All operators are binary. All input strings will consist of only characters in the set "0123456789+-*/%()" (no whitespace).

Output

Your program should output n lines, each containing the fully parenthesized expression s.

A fully parenthesized expression is one where each operator is one where all operands are surrounded by either one parenthesis and one operator, or two parentheses. There should not be any extra parentheses.

The outputted expression does not need to be a solvable expression.

Fully Parenthesized	Not Fully Parenthesized
(42)	42
(1+(2*3))	1+(2*3)
(1+((61%(3+6))*7))	1+61%(3+6)*7

Example

Input
6
42
1+2*3
1%6*2
2/0
(1+((61%(3+6))*7))
(((((1)))))

Output
(42)
(1+(2*3))
((1%6)*2)
(2/0)
(1+((61%(3+6))*7))
(1)

Score

Your score is the length of your source code.