

## 문제 설명

카카오에 신입 개발자로 입사한 **콘**은 선배 개발자로부터 개발역량 강화를 위해 다른 개발자가 작성한 소스 코드를 분석하여 문제점을 발견하고 수정하라는 업무 과제를 받았습니다. 소스를 컴파일하여 로그를 보니 대부분 소스 코드 내 작성된 괄호가 개수는 맞지만 짝이 맞지 않은 형태로 작성되어 오류가 나는 것을 알게 되었습니다. 수정해야 할 소스 파일이 너무 많아서 고민하던 콘은 소스 코드에 작성된 모든 괄호를 뽑아서 올바른 순서대로 배치된 괄호 문자열을 알려주는 프로그램을 다음과 같이 개발하려고 합니다.

## 용어의 정의

'(' 와 ')' 로만 이루어진 문자열이 있을 경우, '(' 의 개수와 ')' 의 개수가 같다면 이를 균형잡힌 괄호 문자열이라고 부릅니다.

그리고 여기에 '('와 ')'의 괄호의 짝도 모두 맞을 경우에는 이를 올바른 괄호 문자열이라고 부릅니다.

예를 들어, "(()))("와 같은 문자열은 균형잡힌 괄호 문자열 이지만 올바른 괄호 문자열은 아닙니다.

반면에 "(()())"와 같은 문자열은 균형잡힌 괄호 문자열 이면서 동시에 올바른 괄호 문자열 입니다.

'(' 와 ')' 로만 이루어진 문자열  $w$  가 균형잡힌 괄호 문자열 이라면 다음과 같은 과정을 통해 올바른 괄호 문자열로 변환할 수 있습니다.

1. 입력이 빈 문자열인 경우, 빈 문자열을 반환합니다.
2. 문자열  $w$ 를 두 "균형잡힌 괄호 문자열"  $u$ ,  $v$ 로 분리합니다. 단,  $u$ 는 "균형잡힌 괄호 문자열"로 더 이상 분리할 수 없어야 하며,  $v$ 는 빈 문자열이 될 수 있습니다.
3. 문자열  $u$ 가 "올바른 괄호 문자열" 이라면 문자열  $v$ 에 대해 1 단계부터 다시 수행합니다.
  - 3-1. 수행한 결과 문자열을  $u$ 에 이어 붙인 후 반환합니다.
4. 문자열  $u$ 가 "올바른 괄호 문자열"이 아니라면 아래 과정을 수행합니다.
  - 4-1. 빈 문자열에 첫 번째 문자로 '('를 붙입니다.
  - 4-2. 문자열  $v$ 에 대해 1 단계부터 재귀적으로 수행한 결과 문자열을 이어 붙입니다.
  - 4-3. ')'를 다시 붙입니다.
  - 4-4.  $u$ 의 첫 번째와 마지막 문자를 제거하고, 나머지 문자열의 괄호 방향을 뒤집어서 뒤에 붙입니다.
  - 4-5. 생성된 문자열을 반환합니다.

균형잡힌 괄호 문자열  $p$ 가 매개변수로 주어질 때, 주어진 알고리즘을 수행해 **올바른 괄호 문자열**로 변환한 결과를 return 하도록 solution 함수를 완성해 주세요.

## 매개변수 설명

- $p$ 는 '(' 와 ')' 로만 이루어진 문자열이며 길이는 2 이상 1,000 이하인 짝수입니다.
  - 문자열  $p$ 를 이루는 '(' 와 ')' 의 개수는 항상 같습니다.
  - 만약  $p$ 가 이미 올바른 괄호 문자열이라면 그대로 return 하면 됩니다.
-

## 입출력 예

p	result
"((()))"	"((()))"
")("	"("
"()))(("	"()((())"

## 입출력 예에 대한 설명

### 입출력 예 #1

이미 올바른 괄호 문자열 입니다.

### 입출력 예 #2

- 두 문자열  $u, v$  로 분리합니다.
  - $u = ")("$
  - $v = ""$
- $u$  가 올바른 괄호 문자열이 아니므로 다음과 같이 새로운 문자열을 만듭니다.
  - $v$  에 대해 1 단계부터 재귀적으로 수행하면 빈 문자열이 반환됩니다.
  - $u$  의 앞뒤 문자를 제거하고, 나머지 문자의 괄호 방향을 뒤집으면 ""이 됩니다.
  - 따라서 생성되는 문자열은 "(" + "" + ")" + ""이며, 최종적으로 "("로 변환됩니다.

### 입출력 예 #3

- 두 문자열  $u, v$  로 분리합니다.
  - $u = "()"$
  - $v = "()))(("$
- 문자열  $u$  가 올바른 괄호 문자열이므로 그대로 두고,  $v$  에 대해 재귀적으로 수행합니다.
- 다시 두 문자열  $u, v$  로 분리합니다.
  - $u = "))(("$
  - $v = "()"$
- $u$  가 올바른 괄호 문자열이 아니므로 다음과 같이 새로운 문자열을 만듭니다.
  - $v$  에 대해 1 단계부터 재귀적으로 수행하면 "("이 반환됩니다.
  - $u$  의 앞뒤 문자를 제거하고, 나머지 문자의 괄호 방향을 뒤집으면 "("이 됩니다.
  - 따라서 생성되는 문자열은 "(" + "(" + ")" + "("이며, 최종적으로 "((())("를 반환합니다.
- 처음에 그대로 둔 문자열에 반환된 문자열을 이어 붙이면 "(" + "((())(" = "((())("가 됩니다.