

## TEST 17

1.

카카오에 신입 개발자로 입사한 남영이는 선배 개발자로부터 개발역량 강화를 위해 다른 개발자가 작성한 소스코드를 분석하여 문제점을 발견하고 수정하라는 업무 과제를 받았습니다. 소스를 컴파일하여 로그를 보니 대부분 소스 코드 내 작성된 괄호가 개수를 맞지만 짝이 맞지 않은 형태로 작성되어 오류가 나는 것을 알게 되었습니다.

수정해야 할 소스 파일이 너무 많아서 고민하던 응경이는 소스 코드에 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 문자열** 이라고 부릅니다.

그리고 여기에 ‘(‘ 와 ‘)’ 의 괄호의 짝도 모두 맞을 경우에는 이를 **올바른 괄호 문자열** 이라고 부릅니다.

예를 들어, “((()))(“ 와 같은 문자열은 “균형잡힌 괄호 문자열” 이지만 “올바른 괄호 문자열”은 아닙니다. 반면에 “((()))” 와 같은 문자열은 “균형잡힌 괄호 문자열” 이면서 동시에 “올바른 괄호 문자열” 입니다.

## TEST 17

‘(‘ 와 ’)’ 로만 이루어진 문자열  $w$ 가 “균형잡힌 괄호 문자열” 이라면 다음과 같은 과정을 통해 “올바른 괄호 문자열”로 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 1단계부터 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 1단계부터 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 완성해주세요.

## TEST 17

### 제한사항

- ▶ p는 '(' 와 ')' 로만 이루어진 문자열이며 길이는 2 이상 1,000 이하인 짝수입니다.
- ▶ 문자열 p를 이루는 '(' 와 ')' 의 개수는 항상 같습니다.
- ▶ 만약 p가 이미 “올바른 괄호 문자열” 이라면 그대로 return 하면 됩니다.

### 입출력 예

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

#1> 이미 “올바른 괄호 문자열” 입니다.

#2> 두 문자열 u, v로 분리합니다. u = "()", v = "("

u가 “올바른 괄호 문자열”이 아니므로 다음과 같이 새로운 문자열을 만듭니다.

- > v에 대해 1단계부터 재귀적으로 수행하면 빈 문자열이 반환됩니다.
- > u의 앞뒤 문자를 제거하고, 나머지 문자의 괄호 방향을 뒤집으면 ")" 이 됩니다.
- > 따라서 생성되는 문자열은 "(" + ")" + ")" + ")"이며, 최종적으로 "())"로 변환됩니다.

#3> 두 문자열 u, v로 분리합니다. u = "()", v = "())((("

문자열 u가 “올바른 괄호 문자열” 이므로 그대로 두고, v에 대해 재귀적으로 수행합니다.

다시 두 문자열 u, v로 분리합니다. u = "())(", v = "("

u가 “올바른 괄호 문자열” 이 아니므로 다음과 같이 새로운 문자열을 만듭니다.

- > v에 대해 1단계부터 재귀적으로 수행하면 "()" 이 반환됩니다.
- > u의 앞뒤 문자를 제거하고, 나머지 문자의 괄호 방향을 뒤집으면 "())"이 됩니다.
- > 생성되는 문자열은 "(" + "())" + "()" + "()"이며, 최종적으로 "(()())"를 반환합니다.

처음 문자열에 반환된 문자열을 이어 붙이면 "()" + "(()())" = "()()()()"가 됩니다.

## TEST 17

2.

나만의 카카오 성격 유형 검사지를 만들려고 합니다.

성격 유형 검사는 다음과 같은 4개 지표로 성격 유형을 구분합니다. 성격은 각 지표에서 두 유형 중 하나로 결정됩니다.

지표 번호	성격 유형
1번 지표	라이언형(R), 튜브형(T)
2번 지표	콘형(C), 프로도형(F)
3번 지표	제이지형(J), 무지형(M)
4번 지표	어피치형(A), 네오형(N)

4개의 지표가 있으므로 성격 유형은 총 16가지가 나올 수 있습니다. 예를 들어, “RFMN”이나 “TCMA”와 같은 성격 유형이 있습니다.

검사지에는 총  $n$ 개의 질문이 있고, 각 질문에는 아래와 같은 7개의 선택지가 있습니다.

- 매우 비동의
- 비동의
- 약간 비동의
- 모르겠음
- 약간 동의
- 동의
- 매우 동의

## TEST 17

각 질문은 1가지 지표로 성격 유형 점수를 판단합니다.

예를 들어, 어떤 한 질문에서 4번 지표로 아래 표처럼 점수를 매길 수 있습니다.

선택지	성격 유형 점수
매우 비동의	네오형 3점
비동의	네오형 2점
약간 비동의	네오형 1점
모르겠음	어떤 성격 유형도 점수를 얻지 않습니다
약간 동의	어피치형 1점
동의	어피치형 2점
매우 동의	어피치형 3점

이때 검사자가 질문에서 약간 동의 선택지를 선택할 경우 어피치형(A) 성격 유형 1점을 받게 됩니다. 만약 검사자가 매우 비동의 선택지를 선택할 경우 네오형(N) 성격 유형 3점을 받게 됩니다.

위 예시처럼 네오형이 비동의, 어피치형이 동의인 경우만 주어지지 않고, 질문에 따라 네오형이 동의, 어피치형이 비동의인 경우도 주어질 수 있습니다.

하지만 각 선택지는 고정적인 크기의 점수를 가지고 있습니다.

- 매우 동의나 매우 비동의 선택지를 선택하면 3점을 얻습니다.
- 동의나 비동의 선택지를 선택하면 2점을 얻습니다.
- 약간 동의나 약간 비동의 선택지를 선택하면 1점을 얻습니다.
- 모르겠음 선택지를 선택하면 점수를 얻지 않습니다.

검사 결과는 모든 질문의 성격 유형 점수를 더하여 각 지표에서 더 높은 점수를 받은 성격 유형이 검사자의 성격 유형이라고 판단합니다. 단, 하나의 지표에서 각 성격 유형 점수가 같으면, 두 성격 유형 중 사전 순으로 빠른 성격 유형을 검사자의 성격 유형이라고 판단합니다.

질문마다 판단하는 지표를 담은 1차원 문자열 배열 `survey`와 검사자가 각 질문마다 선택한 선택지를 담은 1차원 정수 배열 `choices`가 매개변수로 주어집니다. 이때, 검사자의 성격 유형 검사 결과를 지표 번호 순서대로 `return` 하도록 `solution` 함수를 완성해주세요.

## TEST 17

### 제한사항

▶  $1 \leq \text{survey의 길이} (= n) \leq 1,000$

- survey의 원소는 “RT”, “TR”, “FC”, “CF”, “MJ”, “JM”, “AN”, “NA” 중 하나입니다.
- survey[i]의 첫 번째 캐릭터는 i+1번 질문의 비동의 관련 선택지를 선택하면 받는 성격 유형을 의미합니다.
- survey[i]의 두 번째 캐릭터는 i+1번 질문의 동의 관련 선택지를 선택하면 받는 성격 유형을 의미합니다.

▶ choices 의 길이 = survey 의 길이

- choices[i] 는 검사자가 선택한 i+1번째 질문의 선택지를 의미합니다.
- $1 \leq \text{choices의 원소} \leq 7$

choices	뜻
1	매우 비동의
2	비동의
3	약간 비동의
4	모르겠음
5	약간 동의
6	동의
7	매우 동의

### 입출력 예

survey	choices	result
[ “AN”, “CF”, “MJ”, “RT”, “NA” ]	[ 5, 3, 2, 7, 5 ]	“TCMA”
[ “TR”, “RT”, “TR” ]	[ 7, 1, 3 ]	“RCJA”