

TEST 8

1.

카카오에 신입 개발자로 입사한 응경이는 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 8

‘(‘ 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 8

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 8

2.

스트리밍 사이트에서 장르 별로 가장 많이 재생된 노래를 두 개씩 모아 베스트 앨범을 출시하려 합니다. 노래는 고유 번호로 구분하며, 노래를 수록하는 기준은 다음과 같습니다.

- > 속한 노래가 많이 재생된 장르를 먼저 수록합니다.
- > 장르 내에서 많이 재생된 노래를 먼저 수록합니다.
- > 장르 내에서 재생 횟수가 같은 노래 중에서는 고유 번호가 낮은 노래를 먼저 수록합니다.

노래의 장르를 나타내는 문자열 배열 genres와 노래별 재생 횟수를 나타내는 정수 배열 plays가 주어질 때, 베스트 앨범에 들어갈 노래의 고유 번호를 순서대로 return 하도록 solution 함수를 완성하세요.

제한사항

- ▶ genres[i] 는 고유번호가 i 인 노래의 장르입니다.
- ▶ plays[i] 는 고유번호가 i 인 노래가 재생된 횟수입니다.
- ▶ genres와 plays의 길이는 같으며, 이는 1 이상 10,000 이하입니다.
- ▶ 장르 종류는 100개 미만이며, 장르에 속한 곡이 하나라면, 하나의 곡만 선택합니다.
- ▶ 모든 장르는 재생된 횟수가 다릅니다.

입출력 예

genres	plays	return
["classic", "pop", "classic", "classic", "pop"]	[500, 600, 150, 800, 2500]	[4, 1, 3, 0]

#1> - classic 장르는 1,450회 재생되었으며, classic 노래는 다음과 같습니다.

> 고유 번호 3: 800회 재생 / 고유 번호 0: 500회 재생 / 고유 번호 2: 150회 재생

- pop 장르는 3,100회 재생되었으며, pop 노래는 다음과 같습니다.

> 고유 번호 4: 2,500회 재생 / 고유 번호 1: 600회 재생

따라서 pop 장르의 [4, 1]번 노래를 먼저, classic 장르의 [3, 0]번 노래를 그 다음에 수록합니다.