

1 배열

주식가격

<문제 설명>

초 단위로 기록된 주식가격이 담긴 배열 prices가 매개변수로 주어질 때, 가격이 떨어지지 않은 기간은 몇 초인지를 return 하도록 solution 함수를 완성하세요.

<제한 조건>

- prices의 각 가격은 1 이상 10,000 이하인 자연수입니다.
- prices의 길이는 2 이상 100,000 이하입니다.

<입출력 예>

prices	return
[1, 2, 3, 2, 3]	[4, 3, 1, 1, 0]

입출력 예 설명

- 1초 시점의 ₩1은 끝까지 가격이 떨어지지 않았습니다.
- 2초 시점의 ₩2은 끝까지 가격이 떨어지지 않았습니다.
- 3초 시점의 ₩3은 1초 뒤에 가격이 떨어집니다. 따라서 1초간 가격이 떨어지지 않은 것으로 봅니다.
- 4초 시점의 ₩2은 1초간 가격이 떨어지지 않았습니다.
- 5초 시점의 ₩3은 0초간 가격이 떨어지지 않았습니다.

2 문자열 처리

최대값과 최소값

<문제 설명>

문자열 `s`에는 공백으로 구분된 숫자들이 저장되어 있습니다. `str`에 나타나는 숫자 중 최소값과 최대값을 찾아 이를 "(최소값) (최대값)"형태의 문자열을 반환하는 함수, `solution`을 완성하세요.

예를들어 `s`가 "1 2 3 4"라면 "1 4"를 리턴하고, "-1 -2 -3 -4"라면 "-4 -1"을 리턴하면 됩니다.

<제한 조건>

- `s`에는 둘 이상의 정수가 공백으로 구분되어 있습니다.

<입출력 예>

s	return
"1 2 3 4"	"1 4"
"-1 -2 -3 -4"	"-4 -1"
"-1 -1"	"-1 -1"

3 스택

올바른 괄호

<문제 설명>

괄호가 바르게 짝지어졌다는 것은 '(' 문자로 열렸으면 반드시 짝지어서 ')' 문자로 닫혀야 한다는 뜻입니다.

예를 들어

- "()()" 또는 "(()())" 는 올바른 괄호입니다.
- ")()(" 또는 "(()(" 는 올바르지 않은 괄호입니다.

'(' 또는 ')' 로만 이루어진 문자열 *s*가 주어졌을 때, 문자열 *s*가 올바른 괄호이면 true를 return 하고, 올바르지 않은 괄호이면 false를 return 하는 solution 함수를 완성해 주세요.

<제한 조건>

- 문자열 *s*의 길이 : 100,000 이하의 자연수
- 문자열 *s*는 '(' 또는 ')' 로만 이루어져 있습니다.

<입출력 예>

s	answer
"()()"	true
"(()())"	true
")()("	false
"(()("	false

4 정렬

가장 큰 수

<문제 설명>

0 또는 양의 정수가 주어졌을 때, 정수를 이어 붙여 만들 수 있는 가장 큰 수를 알아내 주세요.

예를 들어, 주어진 정수가 [6, 10, 2]라면 [6102, 6210, 1062, 1026, 2610, 2106]를 만들 수 있고, 이중 가장 큰 수는 6210입니다.

0 또는 양의 정수가 담긴 배열 numbers가 매개변수로 주어질 때, 순서를 재배치하여 만들 수 있는 가장 큰 수를 문자열로 바꾸어 return 하도록 solution 함수를 작성해주세요.

<제한 조건>

- numbers의 길이는 1 이상 100,000 이하입니다.
- numbers의 원소는 0 이상 1,000 이하입니다.
- 정답이 너무 클 수 있으니 문자열로 바꾸어 return 합니다.

<입출력 예>

numbers	return
[6, 10, 2]	"6210"
[3, 30, 34, 5, 9]	"9534330"