

## 문제 설명

스마트폰 전화 키패드의 각 칸에 다음과 같이 숫자들이 적혀 있습니다.



이 전화 키패드에서 왼손과 오른손의 엄지손가락만을 이용해서 숫자만을 입력하려고 합니다.

맨 처음 왼손 엄지손가락은 [\*] 키패드에 오른손 엄지손가락은 [#] 키패드 위치에서 시작하며, 엄지손가락을 사용하는 규칙은 다음과 같습니다.

1. 엄지손가락은 상하좌우 4가지 방향으로만 이동할 수 있으며 키패드 이동 한 칸은 거리로 1에 해당합니다.
2. 왼쪽 열의 3개의 숫자 [1], [4], [7]을 입력할 때는 왼손 엄지손가락을 사용합니다.
3. 오른쪽 열의 3개의 숫자 [3], [6], [9]를 입력할 때는 오른손 엄지손가락을 사용합니다.

4. 가운데 열의 4개의 숫자 [2], [5], [8], [0]을 입력할 때는 두 엄지손가락의 현재 키패드의 위치에서 더 가까운 엄지손가락을 사용합니다.

4-1. 만약 두 엄지손가락의 거리가 같다면, 오른손잡이는 오른손 엄지손가락, 왼손잡이는 왼손 엄지손가락을 사용합니다.

순서대로 누를 번호가 담긴 배열 numbers, 왼손잡이인지 오른손잡이인 지를 나타내는 문자열 hand가 매개변수로 주어질 때, 각 번호를 누른 엄지손가락이 왼손인 지 오른손인 지를 나타내는 연속된 문자열 형태로 return 하도록 solution 함수를 완성해주세요.

#### [제한사항]

- numbers 배열의 크기는 1 이상 1,000 이하입니다.
- numbers 배열 원소의 값은 0 이상 9 이하인 정수입니다.
- hand는 "left" 또는 "right" 입니다.
  - "left"는 왼손잡이, "right"는 오른손잡이를 의미합니다.
- 왼손 엄지손가락을 사용한 경우는 L, 오른손 엄지손가락을 사용한 경우는 R을 순서대로 이어붙여 문자열 형태로 return 해주세요.

#### 입출력 예

numbers	hand	result
[1, 3, 4, 5, 8, 2, 1, 4, 5, 9, 5]	"right"	"LRLLLRLLRRL"
[7, 0, 8, 2, 8, 3, 1, 5, 7, 6, 2]	"left"	"LRLLRRLLRR"
[1, 2, 3, 4, 5, 6, 7, 8, 9, 0]	"right"	"LLRLLRLLRL"

#### 입출력 예에 대한 설명

#### 입출력 예 #1

순서대로 눌러야 할 번호가 [1, 3, 4, 5, 8, 2, 1, 4, 5, 9, 5]이고, 오른손잡이입니다.

왼손 위치	오른손 위치	눌러야 할 숫자	사용한 손	설명
*	#	1	L	1은 왼손으로 누릅니다.
1	#	3	R	3은 오른손으로 누릅니다.
1	3	4	L	4는 왼손으로 누릅니다.
4	3	5	L	왼손 거리는 1, 오른손 거리는 2이므로 왼손으로 5를 누릅니다.
5	3	8	L	왼손 거리는 1, 오른손 거리는 3이므로 왼손으로 8을 누릅니다.
8	3	2	R	왼손 거리는 2, 오른손 거리는 1이므로 오른손으로 2를 누릅니다.
8	2	1	L	1은 왼손으로 누릅니다.
1	2	4	L	4는 왼손으로 누릅니다.
4	2	5	R	왼손 거리와 오른손 거리가 1로 같으므로, 오른손으로 5를 누릅니다.
4	5	9	R	9는 오른손으로 누릅니다.
4	9	5	L	왼손 거리는 1, 오른손 거리는 2이므로 왼손으로 5를 누릅니다.
5	9	-	-	

따라서 "LRLLLRLLRRL"을 return 합니다.

#### 입출력 예 #2

왼손잡이가 [7, 0, 8, 2, 8, 3, 1, 5, 7, 6, 2]를 순서대로 누르면 사용한 손은 "LRLLRRLLRR"이 됩니다.

#### 입출력 예 #3

오른손잡이가 [1, 2, 3, 4, 5, 6, 7, 8, 9, 0]를 순서대로 누르면 사용한 손은 "LLRLLRLLRL"이 됩니다.

solution.cpp

```
1  #include <string>
2  #include <vector>
3
4  using namespace std;
5
6  string solution(vector<int> numbers, string hand) {
7      string answer = "";
8      int lx = 0, rx = 2, ly = 3, ry = 3;
9      int x, y;
10     for (int i = 0; i < numbers.size(); i++) {
11         if (numbers[i] % 3 == 1) {
12             answer += "L";
13             lx = 0;
14             ly = numbers[i] / 3;
15         }
16         else if (numbers[i] != 0 && numbers[i] % 3 == 0) {
17             answer += "R";
18             rx = 2;
19             ry = (numbers[i] / 3) - 1;
20         }
21         else if (numbers[i] % 3 == 2 || numbers[i] == 0) {
22             if (numbers[i] == 0) {
23                 x = 1;
24                 y = 3;
25             }
26             else {
27                 x = 1;
28                 y = numbers[i] / 3;
29             }
30
31             int d_r = abs(ry - y) + abs(rx - x);
32             int d_l = abs(ly - y) + abs(lx - x);
33
34             if (d_r < d_l) {
35                 answer += "R";
36                 rx = x;
```

실행 결과

테스트 1
입력값 > [1, 3, 4, 5, 8, 2, 1, 4, 5, 9, 5], "right"
기댓값 > "LRLLLRLLRRL"
실행 결과 > 테스트를 통과하였습니다.
테스트 2
입력값 > [7, 0, 8, 2, 8, 3, 1, 5, 7, 6, 2], "left"
기댓값 > "LRLLRRLRLR"
실행 결과 > 테스트를 통과하였습니다.
테스트 3
입력값 > [1, 2, 3, 4, 5, 6, 7, 8, 9, 0], "right"
기댓값 > "LLRLLRLLRL"
실행 결과 > 테스트를 통과하였습니다.

테스트 결과 (~ˇˇ~)~

3개 중 3개 성공

샘플 테스트 케이스를 통과했다는 의미로, 작성한 코드가 문제의 정답은 아닐 수 있습니다.  
(샘플 테스트 케이스는 [테스트 케이스 추가하기] 버튼을 통해 확인하실 수 있습니다.)

초기화

실행

코드 채점하고 제출