paiza - C061: 문제

아이들은 덧셈을 할 때 손가락으로 세어 덧셈하기 때문에 반복 오름의 계산을 할 수 없습니다. 정수가 2개 입력 받아 아이들의 계산방법으로 계산한 결과를 출력하는 프로그램을 작성해주세요. 단, 입력 예2와 같이 선두의 수가 0이 된 경우에도 생략하지 않고 0 을 출력하세요. 예를 들어 예1에서는 아이들은 그림과 같이 계산합니다.

1의 자리는 8+ 5를 계산결과 13이 나와도 올림을 무시해 3, 10의 자리는 9+7을 계산하면 16이 나오지만 올림을 무시해 6이 됩니다. 따라서, 계산의 결과는 98 +75 = 63 이 됩니다. 따라서 결과를 출력하면 63이 됩니다.

- 입력되는 값 덧셈에 사용하는 정수 A, B가 공백으로 구분하여 입력 입력 한 줄이 되어, 끝에 줄 바꿈이 하나 들어갑니다. 각각의 값은 문자열로 표준 입력으로 전달됩니다.
- 기대하는 출력 반복 올림을 무시하고 A와 B를 더한 결과를 출력합니다. 출력은 A와 B의 큰 자릿수에 맞추어주세요. 끝에 줄바꿈을 넣고 불필요한 문자나 빈 행을 포함하지 마십시오.
- 조건: 1 ≤ A, B ≤ 999

	9	8
+	7	5
	X	3
X	6	
	6	3

98 75 출력 예 1 63 입력 예 2 274 840 출력 예 2 014 입력 예 3 624 58

출력 예 3

672

입력 예 1

paiza - C061 : 채점 결과

제출 코드 결과 상세

테스트 번호	입력 케이스 번호	심판 결과	실행 시간
테스트 1	케이스 1 (기본 데이터)	통과	0.07 초
테스트 2	케이스 1 (기본 데이터)	통과	0.08 초
테스트 3	케이스 1 (기본 데이터)	통과	0.08 초
테스트 4	케이스 1 (기본 데이터)	통과	0.08 초
테스트 5	케이스 1 (기본 데이터)	통과	0.08 초
테스트 6	케이스 1 (기본 데이터)	통과	0.08 초
테스트 7	케이스 1 (기본 데이터)	통과	0.08 초
테스트 8	케이스 1 (기본 데이터)	통과	0.08 초
테스트 9	케이스 1 (기본 데이터)	통과	0.08 초
테스트 10	케이스 1 (기본 데이터)	통과	0.08 초

paiza - C061 : 코드

```
limport java.util.ArrayList;
import java.util.List;
import java.util.Scanner;
public class C061 {
   public static List<Integer> resultSum(List<Integer> listA, List<Integer> listB) {
       List<Integer> resultSum = new ArrayList<>();
       for (int \underline{i} = 0; \underline{i} < listA.size(); \underline{i}++) { // aLength == bLength always
           resultSum.add(listA.get(i) + listB.get(i));
                                                         // 올림을 무시하므로 큰 자리수부터 계산해도 무방
                                                         // 덧셈 결과 올림이 존재하면
           if ((resultSum.get(i) / 10) == 1) {
                                                         // i 자리에 올림을 제외한 나머지만 덮어씌운다
               resultSum.set(i, resultSum.get(i) % 10);
                                                         // (올림을 무시)
       return resultSum;
   public static void main(String[] args) {
       Scanner sc = new Scanner(System.in);
       String a = sc.next(); // 98
       String b = sc.next(); // 75
       String[] strArrayA = a.split( regex: ""); // 98 -> 9 , 8
       String[] strArrayB = b.split( regex: ""); // 75 -> 7 , 5
       int aLength = strArrayA.length;
       int bLength = strArrayB.length;
       String compareLength = "a>b"; // aLength > bLength
       int diffLength = Math.abs(aLength - bLength);
```

paiza - C061 : 코드

```
String compareLength = "a>b"; // aLength > bLength
int diffLength = Math.abs(aLength - bLength);
if (aLength < bLength) { // aLength < bLength</pre>
    compareLength = "a<b";</pre>
List<Integer> intListA = new ArrayList<>();
List<Integer> intListB = new ArrayList<>();
if (compareLength.equals("a>b")) {
    // diffLength >= 1 이다
    for (int \underline{i} = 0; \underline{i} < diffLength; \underline{i}++) { // if diffLength == 1
                                             // for문은 i == 0인 경우 1바퀴 돈다
        intListB.add(0);
                                             // Length 차이만큼 앞 자리에 0을 넣는다.
} else if (compareLength.equals("a<b")) {</pre>
    for (int \underline{i} = 0; \underline{i} < \text{diffLength}; \underline{i} + +) {
        intListA.add(0);
for (String s : strArrayA) {
                                             // 문자열 배열에 들어있는 각 문자형 숫자 하나씩을
    intListA.add(Integer.parseInt(s));
                                             // int형으로 형변환해 리스트에 순차적으로 더해준다
for (String s : strArrayB) {
    intListB.add(Integer.parseInt(s));
// 올림 무시 덧셈 메서드 수행
List<Integer> resultSum = resultSum(intListA, intListB);
for (int i = 0; i < resultSum.size(); i++) {</pre>
```

paiza - C061 : 코드

```
for (String s : strArrayA) {
    intListA.add(Integer.parseInt(s));
                                             // 문자열 배열에 <u>들어있는</u> 각 문자형 숫자 <u>하나씩을</u>
                                             // int형으로 형변환해 리스트에 순차적으로 더해준다
for (String s : strArrayB) {
    intListB.add(Integer.parseInt(s));
// 올림 무시 덧셈 메서드 수행
List<Integer> resultSum = resultSum(intListA, intListB);
for (int \underline{i} = 0; \underline{i} < resultSum \cdot size(); \underline{i}++) {
    System.out.print(resultSum.get(<u>i</u>));
```