

Baekjoon 10799 쇠막대기

📅 날짜	@2021년 8월 9일 → 2021년 8월 9일
🔗 링크	
📄 열	
🏷 태그	Baekjoon SWEA Silver3 stack 구현

$$\textcircled{6} \quad \vdots \quad + \quad | \quad 6$$
$$\textcircled{5} + 2 = 8$$
$$\begin{matrix} 7 \\ + \\ 3 \\ \hline 10 \end{matrix}$$
$$\textcircled{3} + 3 = 14$$

19 + 3 = 22

레이저가 가지는 결함은 없다

$\frac{1}{2} \times 100 = 50$

$\frac{2}{\sqrt{2}}$

1. 개관 연표

२-३॥१५६ प्रोगे, एन्डर्सन
७६४१-

⇒ 3. ~~정적~~ 외부에 대해

String 0123456789

$$\Rightarrow \underline{\text{Char}[A] \text{ 21 쌍}}$$
$$\frac{1}{\sqrt{2}} \frac{d}{dt} \left(\frac{1}{\sqrt{2}} \right)$$

1245 6/11 22/19 4-8/5/12 3/2

① ~~수많은 사용자를~~ 그놈 때문에 사용자들에게

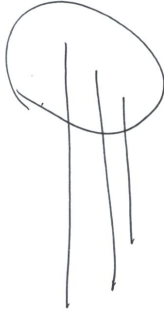
String 한글 입력 ⇒ Char[]로 바꾸기

('' 개수 세기 ('' 연속으로 있을 개수 될 수 있음)
 (count ⇒ 리버서 개수 및 위치 파악
 laserName
 laserIdx

laserName - count ⇒ 외막대기 처리 (반복)

('' 개수 증가할 때 마다 -

외막대기 수가 1 증가하게 된다.



(''')은 일단 정복

(count 1개 감소
⇒ 리버서 위치 파악)

⇒ count 만큼 외막대기 증가

이어서 인코딩 개수 외막대기 가산한다.
외막대기 개수는 laser의 외막대기 가산한다.

char 배열은 꼭 타입만 다.

count 1 '1' 총 개수
count 2 0 이하의 최빈대 크기

```
for (int idx = 0; idx < char.length; idx++)
```

```
if (tmp[idx] == '1')
```

```
count++;
```

```
if (tmp[idx] == '0') && (tmp[idx-1] == '1') {
```

```
count++;  
count--;
```

```
ans += count;
```

```
}
```

```
if (tmp[idx]
```

```
if (tmp[idx] == '1') {
```

```
layer--;
```

```
if (tmp[idx-1] == '1')
```

```
{ ans += layer;
```

```
{ count--;
```

```
{ layer (count++);
```

```
}
```

ans += count - baseCount;

```
import java.io.BufferedReader;  
import java.io.IOException;
```

```

import java.io.InputStreamReader;

//public class Main {
public class Baek10799_2 {
    public static void main(String[] args) throws NumberFormatException, IOException {
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        int layer = 0; // 현재 쇠막대기 층수
        int barCnt = 0; // 쇠막대기 개수
        int ans = 0; // 최종 쇠막대기 수
        char[] tmp = br.readLine().toCharArray();

        for(int idx = 0; idx<tmp.length;idx++) {
            if(tmp[idx] == '(') { // '('인 경우 쇠막대기 층과 쇠막대기 개수가 증가한다.
                ++layer;
                ++barCnt;
            }
            if(tmp[idx] == ')') {
                --layer; // 층은 확실하게 감소한다.
                if(tmp[idx-1] == '(') {
                    ans += layer; // 현재 layer 위에 레이저가 존재하므로 layer 만큼 증가한다.
                    --barCnt; // 레이저임을 파악하고 쇠막대기 수를 감소시킨다.
                }
            }
        }

        System.out.println(ans + barCnt);
    }
}

```