

Java-座位调整-疫情期间课堂的座位进行了特殊的调整

题目描述:

疫情期间课堂的座位进行了特殊的调整, 不能出现两个同学紧挨着, 必须隔至少一个空位。

给你一个整数数组 `desk` 表示当前座位的占座情况, 由若干 `0` 和 `1` 组成, 其中 `0` 表示没有占位, `1` 表示占位。在不改变原有座位秩序情况下, 还能安排坐几个人?

输入描述:

第一行是个子数组表示作为占座情况, 由若干 `0` 和 `1` 组成, 其中 `0` 表示没有占位, `1` 表示占位

输出描述:

输出数值表示还能坐几个人

补充说明:

`1 <= desk.length <= 2 * 104`

示例1

输入: `1, 0, 0, 0, 1`

输出: `1`

说明: 只有`desk[2]`的位置可以坐一个人

```
import java.util.Arrays;
```

```
import java.util.List;
```

```
import java.util.Scanner;
```

```
import java.util.stream.Collectors;
```

```
public class Main {
```

```
    public static void main(String[] args) {
```

```
        Scanner sc = new Scanner(System.in);
```

```
        String line = sc.nextLine();
```

```
        String[] seatArray = line.split(",");
```

```
        List<Integer> collect = Arrays.stream(seatArray).map(Integer::parseInt).collect(Collectors.toList());
```

```
        int sum = 0;
```

```
        for (int i = 0; i < collect.size(); i++) {
```

```
            Integer seat = collect.get(i);
```

```
            // 位置为 1 过滤
```

```
            if (seat == 1) {
```

```
                continue;
```

```
            }
```

```
            // 位置为 0 判断
```

```
            // 开始为 0
```

```
            if (i == 0 && i + 1 < collect.size() && collect.get(i + 1) == 0) {
```

```
                sum++;
```

```
        collect.set(i, 1);
        continue;
    }
    // 最后为 0
    if (i == collect.size() - 1 && i != 0 && collect.get(i - 1) == 0) {
        sum++;
        collect.set(i, 1);
        continue;
    }
    // 中间值
    if (collect.get(i - 1) == 0 && collect.get(i + 1) == 0) {
        sum++;
        collect.set(i, 1);
    }
}
System.out.println(sum);
}
}
```