

座位调整

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: $1 \leq \text{desk.length} \leq 2 * 10^4$

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.Scanner;
import java.io.*;

// 注意类名必须为 Main, 不要有任何 package xxx 信息
public class Main {
    public static void main(String[] args) throws IOException {
        BufferedReader bf = new BufferedReader(new InputStreamReader(System.in));
        String str;
        int count = 0;
        while ((str = bf.readLine()) != null) {
            String[] arr = str.split(",");
            int[] desks = new int[arr.length];
            int width = desks.length;
            for (int i = 0; i < width ; i++) {
                desks[i] = Integer.parseInt(arr[i]);
            }
            if (desks[0] == 0 && desks[1] == 0) {
                count++;
                desks[0] = 1;
            }
            if (desks[width - 1] == 0 && desks[width - 2] == 0) {
                count++;
                desks[width - 1] = 1;
            }
            for (int i = 1; i < width - 1; i++) {
                if (desks[i] == 0 && desks[i - 1] == 0 && desks[i + 1] == 0) {
                    count++;
                    desks[i] = 1;
                }
            }
            System.out.println(count);
        }
    }
}
```

}
}
}