

华为OD机试 - 座位调整 (Java & JS & Python)

原创

伏城之外

已于 2023-05-27 16:38:07 修改

1340

收藏 3

版权

分类专栏:

华为OD机试AB (Java & JS & Python)

文章标签:

算法

华为机试

Java

JavaScript

Python

OD

华为OD机试AB (Ja...)

同时被 2 个专栏收录

¥59.90

¥99.00

3381 订阅

371 篇文章

已订阅

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

备注

1 ≤ desk.length ≤ 2 * 10^4

用例

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

题目解析

我的解题思路如下：

将输入解析为一个整型数组desk，然后遍历每一个元素desk[i]：

- 如果 desk[i] == 0，则说明desk[i]是一个空位，此时只需要检查desk[i]的左右两边也是空位，则说明desk[i]可以坐人，此时将desk[i]更新为1，表示坐人了，而判断左右是否为空位的逻辑如下：
 - 如果 i == 0 || desk[i-1] == 0，则说明desk[i]左边是空位，i == 0 说明其左边没有其他座位了，也相当于空位。
 - 如果 i == desk.length - 1 || desk[i+1] == 0，则说明desk[i]右边也是空位，i == desk.length - 1说明其右边没有其他座位了，也就相当于空位
- 如果 desk[i] != 0，则说明desk[i]不是一个空位，坐不了人，且说明了下一个i+1座位也不能坐人

```
1 import java.util.Arrays;
2 import java.util.Scanner;
3
```

```
4 public class Main {
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7         int[] desk =
8             Arrays.stream(sc.nextLine().split(",")).mapToInt(Integer::parseInt).toArray();
9         System.out.println(getResult(desk));
10    }
11
12    public static int getResult(int[] desk) {
13        int ans = 0;
14
15        for (int i = 0; i < desk.length; i++) {
16            if (desk[i] == 0) {
17                boolean isLeftEmpty = i == 0 || desk[i - 1] == 0;
18                boolean isRightEmpty = i == desk.length - 1 || desk[i + 1] == 0;
19                if (isLeftEmpty && isRightEmpty) {
20                    ans++;
21                    desk[i] = 1;
22                    i++;
23                }
24            }
25        }
26
27        return ans;
28    }
29 }
```