

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：

$1 \leq desk.length \leq 2 * 10^4$

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.*;
```

```
// 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) {
```

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

```
        // 注意 hasNext 和 hasNextLine 的区别
```

```
        String str=in.nextLine();
```

```
        StringBuffer stb=new StringBuffer(str);
```

```
        int sum=0;
```

```
        for(int i=0;i<stb.length();i+=2){
```

```
            if(stb.charAt(i)=='0'){
```

```
                if(i==0&&stb.charAt(i+2)=='0'){
```

```
                    stb.setCharAt(i,'1');
```

```
        sum++;
    }
    else if(stb.charAt(i+2)=='0'&&stb.charAt(i-2)=='0'){
        stb.setCharAt(i,'1');
        sum++;
    }
    else if(i==stb.length()-1&&stb.charAt(i+2)=='0'){
        stb.setCharAt(i,'1');
        sum++;
    }
}
System.out.println(sum);
}
```