

Java-题目描述:

某长方形停车场，每个车位上方都有对应监控器，当且仅当在当前车位或者前后左右四个方向任意一个车位范围停车时，监控器才需要打开；

给出某时刻停车场的停车分布，请统计最少需要打开多少个监控器；

输入描述:

第一行输入 m, n 表示长宽，满足 $1 < m, n \leq 20$ ；后面输入 m 行，每行有 n 个 0 或 1 的整数，整数间使用一个空格隔开，表示该行已停车情况，其中 0 表示空位， 1 表示已停；

输出描述:

最少需要打开监控器的数量；

补充说明:

示例 1

输入:

```
3 3
0 0 0
0 1 0
0 0 0
```

输出:

```
5
```

说明:

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

```
// 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 sc = new Scanner(System.in);
        int m = sc.nextInt();
        int n = sc.nextInt();
        int[][] ints = new int[m][n];
```

```

// 注意 hasNext 和 hasNextLine 的区别
for (int i = 0; i < m; i++) {
    for (int j = 0; j < n; j++) {
        ints[i][j] = sc.nextInt();
    }
}
int res = 0;
for (int i = 0; i < m; i++) {
    for (int j = 0; j < n; j++) {
        if (ints[i][j] == 1) {
            res++;
        } else {
            if (i > 0 && ints[i - 1][j] == 1) {
                res++;
            } else if (j > 0 && ints[i][j - 1] == 1) {
                res++;
            } else if (j < n - 1 && ints[i][j + 1] == 1) {
                res++;
            } else if (i < m - 1 && ints[i + 1][j] == 1) {
                res++;
            } else {
            }
        }
    }
}
System.out.println(res);
}
}

```