

原创

若博豆 已于 2023-02-06 15:39:43 修改 2249 收藏 4

分类专栏: 华为OD机试2023 (JAVA) 文章标签: 算法 华为 java

华为OD机试2023 (...

专栏收录该内容

¥59.90

¥99.00

750 订阅

216 篇文章

已订阅

超级会员免费看

华为OD机试真题，2023年度机试题库全覆盖，刷题指南[点这里](#)

查找单入口空闲区域

时间限制：1s 空间限制：256MB 限定语言：不限

题目描述：

给定一个 $m \times n$ 的矩阵，由若干字符 'X' 和 'O' 构成，'X' 表示该处已被占据，'O' 表示该处空闲，请找到最大的单入口空闲区域。

解释：

空闲区域是由连通的 'O' 组成的区域，位于边界的 'O' 可以构成入口，单入口空闲区域即有且只有一个位于边界的 'O' 作为入口的由连通的 'O' 组成的区域。

如果两个元素在水平或垂直方向相邻，则称它们是“连通”的。

输入描述：

第一行输入为两个数字，第一个数字为行数 m ，第二个数字为列数 n ，两个数字以空格分隔， $1 \leq m, n \leq 200$ ，

剩余各行行为矩阵各行元素，元素为 'X' 或 'O'，各元素间以空格分隔。

输出描述：

若有唯一符合要求的最大单入口空闲区域，输出三个数字，第一个数字为入口行坐标（范围为 $0 \sim$ 行数-1），第二个数字为入口列坐标（范围为 $0 \sim$ 列数-1），第三个数字为区域大小，三个数字以空格分隔；

若有多个符合要求的最大单入口空闲区域，输出一个数字，代表区域的大小；

若没有，输出 NUL。

示例1

输入：

```
4 4
XXXX
XOOX
XOOX
XOOX
```

输出：

```
3 1 5
```

说明：

存在最大单入口区域，入口行坐标3，列坐标1，区域大小5

示例2

输入：

```
4 5
XXXXX
OOOOX
XOOOX
XOOXO
```

输出：

```
3 4 1
```

说明：

存在最大单入口区域，入口行坐标3，列坐标4，区域大小1

示例3

输入：

```
5 4
XXXXX
XOOO
XOOO
XOOX
XXXX
```

输出：

```
NUL
```

说明：

不存在最大单入口区域

示例4

输入：

```
5 4
XXXXX
XOOO
XXXXX
XOOO
XXXX
```

输出：

```
3
```

说明：

存在两个大小为3的最大单入口区域，两个入口横纵坐标分别为1,3和3,3

解题思路：

通过回溯法求出所有满足的区域

在回溯的同时记录其入口坐标

入口个数大于1则不符合要求；

入口个数等于1时，判断其区域大小；如果存在多个区域，且区域大小相同，则

只需记录其大小；其他情况则需要记录区域最大值和横纵坐标。

代码：

```
1 public class Main{
2
3     public static int m;
4     public static int n;
5     public static String[][] strings;
6     public static int[] rukou = new int[2]; //入口坐标
7     public static int count = 0; //入口个数
8
9
10    public static void main(String[] args) {
11
12        Scanner sc = new Scanner(System.in);
13
14        m = sc.nextInt();
15        n = sc.nextInt();
16
17        strings = new String[m][n];
18        sc.nextLine();
19
20        for(int i=0;i<m;i++){
21            String[] strInput = sc.nextLine().split(" ");
22            for(int j=0;j<n;j++){
23                strings[i][j] = strInput[j];
24            }
25        }
26
27        int max = 0; //最大的区域大小
28        List<int[]> quyu = new ArrayList<>(); //最大区域的入口坐标和其区域大小的集合
29        for(int i=0; i<m; i++){
30            for(int j=0; j<n; j++){
31                if(strings[i][j].equals("O")){
32                    strings[i][j] = "X"; //已经统计过的区域置为"X"
33                    List<int[]> zuobiao = new ArrayList<>(); //区域中的坐标集合
34                    zuobiao.add(new int[]{i,j});
35                    qiuquyu(i, j, zuobiao);
36                    if(count == 1){ //只有一个入口的区域
37                        if(max == zuobiao.size()){ //有大小相同的单入口空闲区域，只需要大小，无需
38                            quyu.clear();
39                        }else if(max < zuobiao.size()){
40                            quyu.clear();
41                            quyu.add(new int[]{rukou[0], rukou[1], zuobiao.size()});
42                            max = zuobiao.size();
43                        }
44                    }
45                    count = 0; //重置入口数量
46                    rukou = new int[2]; //重置入口坐标
47                }
48            }
49        }
50
51        if(quyu.size() == 1){
52            int[] res = quyu.get(0);
53            System.out.println(res[0] + " " + res[1] + " " + res[2]);
54        }else if(max != 0){
55            System.out.println(max);
56        }else {
57            System.out.println("NULL");
58        }
59
60    }
61
62    /**
63     *
64     * @param x 横坐标
65     * @param y 纵坐标
66     * @param list 区域内的坐标集合
67     */
68    public static void qiuquyu(int x,int y,List<int[]> list){
69
70        if(x==0 || x == m-1 || y ==0 || y == n-1){ //边界为入口坐标
71            count++; //入口的数量计数
72            rukou[0] = x;
73            rukou[1] = y;
74        }
75
76        if(x<m-1){
77            if(strings[x+1][y].equals("O")){
78                strings[x+1][y] = "X"; //已经统计过的区域需要置为"X"，避免统计重复
79                list.add(new int[]{x+1,y});
80                qiuquyu(x+1, y, list);
81            }
82        }
83        if(y<n-1){
84            if(strings[x][y+1].equals("O")){
85                strings[x][y+1] = "X"; //已经统计过的区域需要置为"X"，避免统计重复
86                list.add(new int[]{x,y+1});
87                qiuquyu(x, y+1, list);
88            }
89        }
90        if(y>0){
91            if(strings[x][y-1].equals("O")){
92                strings[x][y-1] = "X"; //已经统计过的区域需要置为"X"，避免统计重复
93                list.add(new int[]{x,y-1});
94                qiuquyu(x, y-1, list);
95            }
96        }
97    }
98
99 }
```

📖 文章知识点与官方知识档案匹配，可进一步学习相关知识

算法技能树 > 首页 > 概览 40101 人正在系统学习中

若博豆 已关注

1 4 8 专栏目录 已订阅

华为OD机试真题Java_2022-2023-题目0180-查找单入口空闲区域 amos_cloud 的博文 985

给定一个m*n的矩阵，由若干字符X和O构成，X表示该处已被占据，O表示该处空闲，请找到最大的单入口空闲区域。空闲区域是由连...

易语言139邮箱发短信 易语言139邮箱发短信,发送短信,用户登录,Ansi转Utf8,Ansi转Unicode,Unicode转Utf8,取十进制,HTTP读网页,COOKI...

评论 8条 > 写评论

华为OD机试 - 查找单入口空闲区域 (Java & JS & Python) qfc_1282202 的博文 1988

2022.Q4 已支持Java、JavaScript、Python，考察：深度优先搜索DFS

华为OD真题学习-查找单入口空闲区域 100 qq_30402163 的博文 246

查找单入口空闲区域 递归，回溯算法

一级建造师2022年重大调整，看看你是否符合报考条件？ 全国一级建造师报考条件

【华为OD机试 c++】 查找单入口空闲区域 【2023 Q1考试题目 A卷 | 100分】 goldmamour 的博文 188

【华为OD机试 c++】 查找单入口空闲区域 【2023 Q1考试题目 A卷 | 100分】

【华为OD机试真题 python】 查找单入口空闲区域 【2022 Q4 | 100分】 无厘de招的博文 821

华为OD机试 Python OD笔试 查找单入口空闲区域：给定一个 $m \times n$ 的矩阵，由若干字符 'X' 和 'O' 构成，'X' 表示该处已被占据，'O' 表示...

【华为OD机试真题 Java】 查找单入口空闲区域 qq_40642440 的博文 268

给定一个 $m \times n$ 的矩阵，由若干字符 'X' 和 'O' 构成，'X' 表示该处已被占据，'O' 表示该处空闲，请找到最大的单入口空闲区域。解...

有序矩阵元素查找LeetCode简单题 轩玉的博文 340

已知int一个有序矩阵mat，同时给定矩阵的大小n和m以及需要查找的元素x，且矩阵的行和列都是从小到大有序的。设计查找算法返回所...

【华为机试真题详解】 查找单入口空闲区域 【2022 Q4 | 100分】 不太灵光的程序员 2715

华为机试真题详解 Python实现【华为面试题】专栏、华为面经总结、华为OD机试真题。给定一个 $m \times n$ 的矩阵，由若干字符X和O...

华为面试宝典OD MISAYAONE 的博文 47万+

官方介绍：OD，全称(Outsourcing Dispatch)模式，目前华为和德科联合招聘的简称。目前华为社招大多数是OD招聘，17级以下都为OD...

华为OD机试真题 Java 实现【查找单入口空闲区域】 【2022.11 Q4 新题】 MISAYAONE 的博文 8259

华为OD机试真题 Java 实现【查找单入口空闲区域】 【2022.11 Q4 新题】 MISAYAONE 的博文 7880

给定一个 $m \times n$ 的矩阵，由若干字符 'X' 和 'O' 构成，'X' 表示该处已被占据，'O' 表示该处空闲，请找到最大的单入口空闲区域。

【华为OD机试真题 Python】 查找单入口空闲区域 A_D_L_D_A_S 的博文 68

给定一个 $m \times n$ 的矩阵，由若干字符 'X' 和 'O' 构成，'X' 表示该处已被占据，'O' 表示该处空闲，请找到最大的单入口空闲区域。解...

华为机试 (Python) 真题Od 热门推荐 MISAYAONE 的博文 4万+

2022.11 开始机试应该是换新题库了，优先更新最新的题目。

华为OD机试真题 Python 实现【查找单入口空闲区域】 【2022.11 Q4 新题】 MISAYAONE 的博文 8038

给定一个 $m \times n$ 的矩阵，由若干字符 'X' 和 'O' 构成，'X' 表示该处已被占据，'O' 表示该处空闲，请找到最大的单入口空闲区域。

华为机试 (C++) 真题Od MISAYAONE 的博文 4万+

2022.10.01 目标：在12月更新新到200题。每个季度出新题都会更新到这个专栏里。最后，好好刷题，好好生活~

解题思路-leetcode130题：被围绕的区域 PPlेमoin123 的博文 506

解题思路-leetcode130题：被围绕的区域 题目描述： 给你一个 $m \times n$ 的矩阵 board，由若干字符 'X' 和 'O'，找到所有被 'X' 围绕的区城...

【华为OD机试真题2023 JS】 查找单入口空闲区域 最新发布 weixin_40767375 的博文 55

查找单入口空闲区域 题目描述： 给定一个 $m \times n$ 的矩阵，由若干字符 'X' 和 'O' 构成，'X' 表示该处已被占据，'O' 表示该处空闲，请找到最...

2021-10-14：被围绕的区域。给你一个 $m \times n$ 的矩阵 board，由若干字符 'X' 和 'O'，找到... 福大大架构师每日一题 485

2021-10-14：被围绕的区域。给你一个 $m \times n$ 的矩阵 board，由若干字符 'X' 和 'O'，找到所有被 'X' 围绕的区域，并将这些区域里所有...

“相关推荐”对你有帮助么？

😞 非常没帮助 😊 没帮助 😐 一般 😄 有帮助 😁 非常有帮助

关于我们 招贤纳士 商务合作 寻求报道 400-660-0108 kefu@csdn.net 在线客服 工作时间 8:30-22:00

公安备案号11010502030143 京ICP备19004658号 京网文〔2020〕1039-165号 经营性网站备案信息 北京互联网违法和不良信息举报中心

家长监护 网络110报警服务 中国互联网举报中心 Chrome商店下载 账号管理规范 版权与免贵声明 版权申诉 出版物许可证 营业执照

©1999-2023北京创新乐知网络技术有限公司

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta

Beta