

diou 模拟赛

时间：2024 年 9 月 19 日 08:00 ~ 12:30

题目名称	跑步	网络	游走	移除
题目类型	传统型	传统型	传统型	传统型
可执行文件名	run.exe	net.exe	walk.exe	move.exe
输入文件名	run.in	net.in	walk.in	move.in
输出文件名	run.out	net.out	walk.out	move.out
每个测试点时限	1.0 秒	1.0 秒	2.0 秒	1.0 秒
内存限制	512 MiB	512 MiB	512 MiB	512 MiB
测试点数目	10	5	20	10
测试点是否等分	是	是	是	是

提交源程序文件名

对于 C++ 语言	run.cpp	net.cpp	walk.cpp	move.cpp
-----------	---------	---------	----------	----------

编译选项

对于 C++ 语言	-O2 -std=c++14 -Wl,--stack=536870912
-----------	--------------------------------------

跑步 (run)

【题目描述】

He_ren 每天沿玄武湖跑步。玄武湖可以抽象成一个 $n \times m$ 的矩阵，每个格子可能是空地，也可能是水。He_ren 每天会选择一个空地格子作为起点，找一个以它为右下角且周长最大的空矩阵，然后沿着这个空矩阵的边界跑一圈，也就是说这一天的跑步里程等于空矩阵的周长。

若干天后，He_ren 已经以每个空地格子作为起点恰好一次，你需要统计出，对于每个 $i \in [1, n + m]$ ，有多少天的跑步里程恰好为 $2i$ 。

【输入格式】

从文件 *run.in* 中读入数据。

第一行包含两个正整数 n, m ，分别表示矩阵的行数和列数。

接下来 n 行，每行一个长度为 m 的字符串，每个字符为 `.` 或者 `#`，分别表示空地格子和水格子。

【输出格式】

输出到文件 *run.out* 中。

一行包含 $n + m$ 个整数，第 i 个整数表示有多少天的跑步里程恰好为 $2i$ 。

【样例 1 输入】

```
1 6 6
2 ..#.#
3 .#....
4 #..##.
5 ...#..
6 #.....
7 #..#..
```

【样例 1 输出】

```
1 0 6 6 6 4 3 0 0 0 0 0 0
```

【样例 2 输入】

```
1 6 6
2 #.#..#
3 .#...#.
4 #...#..
5 ..#...#
6 .#...#.
7 #...#.#
```

【样例 2 输出】

```
1 0 13 9 0 0 0 0 0 0 0 0 0
```

【样例 3 输入】

```
1 5 6
2 #.#..#
3 .#...#.
4 #.....
5 .....
6 .....
```

【样例 3 输出】

```
1 0 7 5 3 3 3 2 1 0 0 0
```

【子任务】

对于全部数据， $1 \leq n, m \leq 5000$ 。

- 对于 20% 的数据： $n, m \leq 50$ 。
- 对于 40% 的数据： $n, m \leq 500$ 。
- 对于 80% 的数据： $n, m \leq 2000$ 。

网络 (net)

【题目描述】

某国有 n 个城市，现在准备建立连接全国的广域网。经过考察，一些城市之间可以直接建造光缆，但为了节约成本，只能建造其中的 $n - 1$ 条光缆来使全国连通，也就是选一棵生成树。另外，还需要最小化城市两两之间信息传递时间的最大值，定义两个城市之间的信息传递时间为树上路径的 **节点数**。

【输入格式】

从文件 *net.in* 中读入数据。

第一行包含一个正整数 n ，表示城市的数量。

接下来 n 行，每行包含一个长度为 n 的 01 串，用 (i, j) 表示第 i 行的第 j 个字符。 $(i, j) = 1$ 说明城市 i, j 之间可以直接建造光缆， $(i, j) = 0$ 说明不能直接建造光缆，保证 $(i, j) = (j, i)$ 并且 $(i, i) = 0$ 。

【输出格式】

输出到文件 *net.out* 中。

共 $n - 1$ 行，每行包含两个正整数 u, v ，表示 u 和 v 之间直接建造光缆。

输出任意一组最优解即可。

【样例 1 输入】

```
1 4
2 0100
3 1011
4 0101
5 0110
```

【样例 1 输出】

```
1 1 2
2 2 3
3 2 4
```

【样例 2 输入】

```
1 5
2 01001
3 10100
4 01010
5 00101
6 10010
```

【样例 2 输出】

```
1 1 2
2 1 5
3 2 3
4 4 5
```

【样例 3 输入】

```
1 8
2 01101100
3 10011010
4 10010101
5 01100011
6 11000000
7 10100000
8 01010000
9 00110000
```

【样例 3 输出】

```
1 1 2
2 1 3
3 1 5
4 1 6
5 2 4
6 2 7
7 3 8
```

【子任务】

本题采用子任务捆绑和依赖。

对于全部数据 $1 \leq n \leq 2500$ 。

保证建造完所有可以直接建造的光缆后所有城市连通。

- 子任务 1 (20 分): $n \leq 6$ 。
- 子任务 2 (20 分): $n \leq 50$, 依赖于子任务 1。
- 子任务 3 (20 分): $n \leq 500$, 保证最优解中信息传递时间的最大值为 奇数。
- 子任务 4 (20 分): $n \leq 500$, 依赖于子任务 23。
- 子任务 5 (20 分): 无特殊限制, 依赖于子任务 4。

游走 (walk)

【题目描述】

有 n 个黑衣 He_ren 和 m 个红衣 He_ren 在走廊上游走，走廊的长度为 L ，可以抽象成一个区间 $[0, L]$ 。每个 He_ren 的游走速度恒定，每一个单位时间移动一个单位长度，而方向可能是向左或向右。当两个 He_ren 相遇即游走到同一个点时，它们会同时反向并保持速度不变。当一个 He_ren 游走到走廊的尽头（坐标为 0 和 L 的位置）时，也会反向并保持速度不变。

有 q 个蒟蒻在一旁膜拜 He_ren，第 i 个蒟蒻的观测区间为 $[l_i, r_i]$ ，保证这些观测区间两两不相交。你需要回答，从时刻 0 到时刻 T （包括 T ），每个蒟蒻分别观测到以下事件各多少次：

- 两个黑衣 He_ren 相遇。
- 两个红衣 He_ren 相遇。
- 一个黑衣 He_ren 和一个红衣 He_ren 相遇。

【输入格式】

从文件 *walk.in* 中读入数据。

第一行包含五个非负整数 n, m, q, L, T ，分别表示黑衣 He_ren 数量、红衣 He_ren 数量、蒟蒻数量、走廊长度和时间。

接下来 n 行，每行包含两个整数 x, p ，分别表示第 i 个黑衣 He_ren 的位置和游走方向， $p = 1$ 表示向右游走， $p = -1$ 表示向左游走。

接下来 m 行，每行包含两个整数 x, p ，分别表示第 i 个红衣 He_ren 的位置和游走方向。

接下来 q 行，每行包含两个非负整数 l_i, r_i ，表示第 i 个蒟蒻的观测范围。

【输出格式】

输出到文件 *walk.out* 中。

共 q 行，第 i 行包含三个整数，分别表示第 i 个蒟蒻观测到的两个黑衣 He_ren 相遇、两个红衣 He_ren 相遇、一个黑衣 He_ren 和一个红衣 He_ren 相遇的次数。

【样例 1 输入】

```
1 2 1 1 8 7
2 2 1
3 4 -1
4 7 1
```

5

3 6

【样例 1 输出】

1

1 0 1

【样例 1 解释】

$T = 1$ 时，第一个 He_ren 和第二个 He_ren 在位置 3 相遇。

$T = 1$ 时，第三个 He_ren 走到走廊尽头 L 。

$T = 3.5$ 时，第二个 He_ren 和第三个 He_ren 在位置 5.5 相遇。

$T = 4$ 时，第一个 He_ren 走到走廊尽头 0。

$T = 6$ 时，第三个 He_ren 走到走廊尽头 L 。

$T = 6.5$ 时，第一个 He_ren 和第二个 He_ren 相遇于位置 2.5，注意这不在观测范围内。

【子任务】

对于全部数据 $n, m \geq 0, 2 \leq n + m \leq 10^6, 1 \leq q \leq 10, 2 \leq L \leq 10^8, 0 \leq T \leq 10^9, 0 < x < L, p \in \{1, -1\}, 0 \leq l_i < r_i \leq L$

输入数据保证：

- 开始时所有 He_ren 的位置两两不同。
- 所有 $[l_i, r_i]$ 两两不交。

测试点编号	$n + m$	其他约定
1, 2	≤ 10	$q \leq 3, L \leq 1000, T \leq 10^5$
3, 4	$\leq 10^2$	$m = 0, L \leq 10^6$
5, 6	$\leq 10^3$	$m = 0$
7, 8		$q = 1, l_1 = 0, r_1 = L$
9, 10	$\leq 10^5$	
11, 12		
13 ~ 20	$\leq 10^6$	

移除 (move)

【题目描述】

在二维平面上，有 n 个 He_ren 平躺在上面，可以抽象成 n 条互不相交的线段，你的任务是通过如下的操作将所有 He_ren 移除：

- 选择一个 He_ren，按水平或者垂直方向将他移除（也就是平移到无穷远处），前提是移动过程中不被任何尚未移除的 He_ren 所阻碍。如果会被阻碍，那么这次移除就是非法的。

你需要解决两个问题：

1. 给定一个移除方案，求出最早的一次非法移除。
2. 给出一个合法的移除方案。

两个问题各占一半的分数。

注意：在线段移动时仅端点接触不会造成阻碍。

【输入格式】

从文件 `move.in` 中读入数据。

第一行包含一个正整数 n ，表示 He_ren 的数量。

接下来 n 行，每行包含 4 个整数 a_i, b_i, c_i, d_i ，表示编号为 i 的 He_ren 所抽象成的线段的端点为 (a_i, b_i) 和 (c_i, d_i) 。

接下来 n 行，每行 2 个整数 p_i, q_i ，表示第 i 轮移动的 He_ren 编号为 p_i ，方向为 q_i 。其中 q_i 为一个 0 到 3 之间的整数，0 表示向左平移（即 x 轴负方向），1 表示向上平移（即 y 轴正方向），2 表示向右平移，3 表示向下平移。

【输出格式】

输出到文件 `move.out` 中。

第一行包含一个正整数，表示最早出现非法移动的是哪一轮。

接下来 n 行，每行包含两个整数，内容同输入格式所述，描述一个合法的移除方案。请确保输出格式正确，即使只解决了一个问题，也要输出完整，否则判作 0 分。

【样例 1 输入】

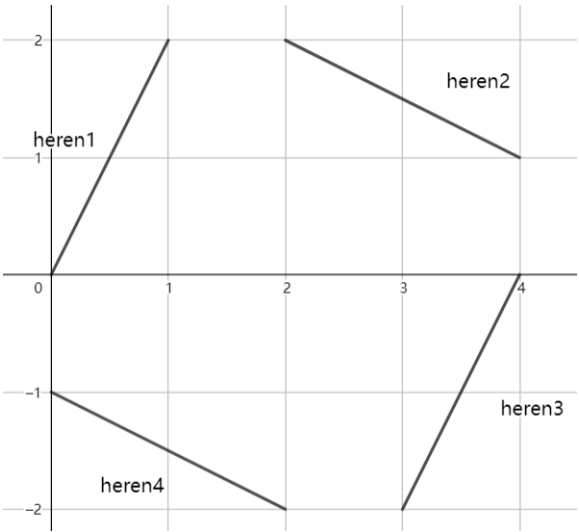
```
1 4
2 0 0 1 2
3 2 2 4 1
4 4 0 3 -2
5 2 -2 0 -1
```

```
6 1 0
7 2 1
8 3 0
9 4 3
```

【样例 1 输出】

```
1 3
2 1 0
3 2 1
4 3 2
5 4 3
```

【样例 1 解释】



【样例 2 输入】

```
1 6
2 0 0 1 1
3 1 0 2 1
4 2 0 3 1
5 0 1 1 2
6 3 2 4 1
7 2 2 4 3
```

```

8 2 1
9 6 0
10 5 1
11 3 0
12 4 1
13 1 3

```

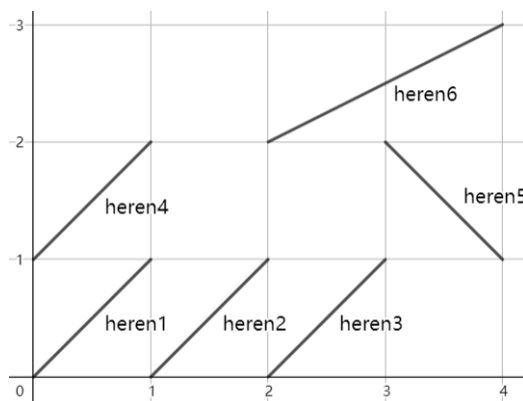
【样例 2 输出】

```

1 4
2 2 1
3 6 0
4 5 1
5 3 2
6 4 1
7 1 3

```

【样例 2 解释】



【子任务】

对于全部数据 $1 \leq n \leq 10^5$, $|a_i|, |b_i|, |c_i|, |d_i| \leq 10^9$ 。

输入数据保证：

- 所有线段长度为正，两两之间没有公共点（包括端点），且不存在垂直或者水平的线段。
- p_1 到 p_n 恰好组成一个 1 到 n 的排列。
- 给定的移除方案中一定存在非法移动。

- n 轮均合法的移除方案总是存在的。

对于一个测试点：两个问题各占 5 分。

测试数据编号	n	其他约定	
1	$=3$	对任意 $2 \leq i \leq n$	对任意 $1 \leq i \leq n$ 有 $q_i = 1$ $ a_i , b_i , c_i , d_i \leq 10^4$
2	≤ 8	有 $b_{i-1} < d_{i-1} < b_i < d_i$	
3	≤ 100	对任意 $1 \leq i \leq n$	
4	≤ 2000	有 $a_i < c_i, d_i - b_i = 1$	
5	≤ 2000	/	对任意 $1 \leq i \leq n$ 有 $0 \leq q_i \leq 3$ $ a_i , b_i , c_i , d_i \leq 10^9$
6	≤ 20000	/	
7	≤ 30000	/	
8	≤ 50000	/	
9	≤ 80000	/	
10	≤ 100000	/	