

寻找盆地

basin

【问题描述】

五一节到了，为了摆脱紧张且繁忙的学习生活，寻找久违的刺激，小强打算和朋友们去冒险城探险。冒险城是一个边长为 n 的正方形区域，被人为地划分成了 $n*n$ 块小区域，每块小区域都是一个单位正方形。为了方便起见，我们定义第 i 行第 j 列的小区域坐标为 (i, j) 。我们称两块小区域 (x_1, y_1) 和 (x_2, y_2) 相邻，当且仅当它们有一条公共边，即 $|x_1 - x_2| + |y_1 - y_2| = 1$ 。每块小区域都有一个海拔高度（用实数表示），不同的小区域高度不同。

小强想在这 $n*n$ 块小区域中找到一块盆地，即一块比相邻小区域海拔都低的小区域。冒险城被一圈很高的围墙包围，所以盆地可以出现在边上，此时它只需比相邻的三块小区域海拔低；盆地甚至可以出现在角上，此时它只需比相邻的两块小区域海拔低。

冒险城入口处有一个地形查询系统，每次查询可以输入两块不同的小区域，系统会告诉你谁的海拔高。由于查询速度很慢，小强想用尽量少的询问次数找到一块盆地，你能帮他设计一种方案吗？

【交互方法】

本题是一道交互式题目，你的程序应当和测试库进行交互，而不得访问任何文件（包括临时文件）。测试库提供了若干函数，它们的用法和作用如下：

- `init` 必须先调用，但只能调用一次，用作初始化测试库。
- `get_n` 的作用是返回冒险城的边长 n 。
- `query(x1, y1, x2, y2)` 的作用是询问小区域 (x_1, y_1) 和 (x_2, y_2) 的相对高低。如果 (x_1, y_1) 比 (x_2, y_2) 高，则返回 1，否则返回 -1。
- `answer(x, y)` 的作用是向测试库报告结果， (x, y) 表示你找到的盆地的坐标。当这个函数被调用后，测试库会自动中止你的程序。

【对使用 Pascal 选手的提示】

你的程序应当使用如下的语句引用测试库。

```
uses basin_lib_p;
```

测试库使用的函数原型为：

```
procedure init;  
function get_n: longint;  
function query(x1, y1, x2, y2: longint): longint;  
procedure answer(x, y: longint);
```

【对使用 C/C++ 选手的提示】

你应当建立一个工程，把文件 `basin_lib_c.o` 包含进来，然后在程序头加一行：

```
#include "basin_lib_c.h"
```

测试库使用的函数原型为：

```
void init();
```

```
int get_n();  
int query(int, int, int, int);  
void answer(int, int);
```

【你如何测试自己的程序】

- 在工作目录下建立一个文件叫做 `basin.in`，文件的第一行包括一个整数 n 为冒险城的边长。接下来的 n 行，每行 n 个实数，表示每个小区域的海拔高度。
- 执行你的程序，此时测试库会产生输出文件 `basin.log`，该文件中包括了你程序和库交互的记录和最后的结果。
- 如果程序正常结束，`basin.log` 的最后一行包含一个整数，为你查询的次数。
- 如果程序非法退出，则我们不保证 `basin.log` 中的内容有意义。
- 正式测试时，测试库会自行生成地形，并使得每次回答尽量对你不利。

【约定】

- $2 \leq n \leq 63$

【样例】

`basin.in` 内容如下

```
3  
9.1 1.2 7.3  
5.4 2.5 3.6  
6.7 8.8 4.9
```

一种可能得满分的调用方案如下：

Pascal 选手的调用方法	C/C++选手的调用方法	说明
<code>init;</code>	<code>init();</code>	初始化程序
<code>query(3,3,2,3);</code>	<code>query(3,3,2,3);</code>	返回 1
<code>query(3,3,3,2);</code>	<code>query(3,3,3,2);</code>	返回-1
<code>query(2,3,2,2);</code>	<code>query(2,3,2,2);</code>	返回 1
<code>query(1,3,2,3);</code>	<code>query(1,3,2,3);</code>	返回 1
<code>query(1,2,2,2);</code>	<code>query(1,2,2,2);</code>	返回-1
<code>query(2,1,2,2);</code>	<code>query(2,1,2,2);</code>	返回 1
<code>query(1,2,1,1);</code>	<code>query(1,2,1,1);</code>	返回-1
<code>query(1,3,1,2);</code>	<code>query(1,3,1,2);</code>	返回 1
<code>answer(1,2);</code>	<code>answer(1,2);</code>	(1,2) 确实是盆地，共查询 8 次

注意，该例子只是对库函数的使用说明，并没有算法上的意义。

【评分方法】

如果你的程序有下列情况之一，该测试点 0 分：

- 访问了任何文件(包括临时文件)或者自行终止；
- 非法调用库函数；
- 让测试库异常退出。
- 查询次数超过 200 次。

否则，该测试点满分。