



山岳救助隊 (Mountain Rescue Team)

IOI 国の政府は IOI 山で遭難事故が絶えないことから IOI 山専門の山岳救助隊を設置した。設置から数日が経ったある日、山岳救助隊のもとに遭難事故を知らせる通報が届いた。通報によれば遭難している人々はいま高度が X のところに留まっているという。

IOI 山は縦 R 行、横 C 列のマス目で表され、各マス目に対して高度の値が決まっている。上から数えて r 行目、左から数えて c 列目のマスを (r, c) で表すことにする。山の形が次のようになっていることは分かっている。

- 高度が同じであるようなふたつの異なるマスは存在しない。
- どのマスの高度も 1 以上 1 000 000 000 以下の整数である。
- マス (R_S, C_S) が山の頂点である。すなわち、最も高度が大きいマスが (R_S, C_S) である。
- ふたつのマス (r_1, c_1) と (r_2, c_2) の距離を $|r_1 - r_2| + |c_1 - c_2|$ と定める。このとき、ふたつの隣接するマスについて、頂点からの距離が遠い方が高度が小さい。ただしふたつのマスが隣接しているとは、それらのマスが辺を共有していることを言う。

山岳救助隊は設置されて間もないため IOI 山の調査はよく済んでおらず、頂点も含め各マスの高度は分かっていない。専門の測定機器を用いて位置を指定したひとつのマスの高度を調べることができるが、調べるのには一定の時間がかかる。この測定機器を用いて遭難者のいる高度 X のマスの位置を特定したいが、そのための測定機器の使用回数は 1 000 回以内におさえない。

課題

IOI 山を表すマス目の大きさを示す整数 R, C 、頂点の位置 R_S, C_S および遭難者のいるマスの高度 X が与えられたとき、測定機器を最大でも 1 000 回用いることで遭難者のいるマスの位置を特定するためのプログラムを作成せよ。

実装の詳細

あなたは測定機器を使用する方法を実装した 1 個のプログラムを書かねばならない。プログラムは、以下のルーチンを実装しなければならない。

- `void Rescue(int R, int C, int RS, int CS, int X)`

このルーチンは、各テストケースに対し 1 回だけ呼び出される。引数 R, C はそれぞれ山を表すマス目の縦の大きさと横の大きさ R, C である。引数 RS, CS はそれぞれ山の頂点の行番号と列番号 R_S, C_S である。引数 X は遭難者のいるマスの高度 X である。



また、プログラム中では以下のルーチンを呼び出すことができる。

- `int Measure(int RM, int CM)`

このルーチンは、測定機器を使用する際に呼び出す。引数 RM, CM はそれぞれ測定機器によって高度を調べるマスの行番号と列番号 R_M, C_M である。 $1 \leq R_M \leq R, 1 \leq C_M \leq C$ を満たす。このルーチンの戻り値はマス (R_M, C_M) の高度を表す 1 以上 1 000 000 000 以下の整数である。

$1 \leq R_M \leq R, 1 \leq C_M \leq C$ という条件を満たさないような値でこのルーチンを呼び出した場合は、**不正解 [1]** と判定され、プログラムの実行は終了される。

また、このルーチンが 1 000 回呼び出されたあとに再び呼び出された場合は **不正解 [2]** と判定され、プログラムの実行は終了される。

- `void Pinpoint(int RP, int CP)`

このルーチンは、遭難者のいるマスが特定できた際に 1 回だけ呼び出す。引数 RP, CP はそれぞれ遭難者がいると特定したマスの行番号と列番号 R_P, C_P である。 $1 \leq R_P \leq R, 1 \leq C_P \leq C$ を満たす。このルーチンは戻り値を持たない。

$1 \leq R_P \leq R, 1 \leq C_P \leq C$ という条件を満たさないような値でこのルーチンを呼び出した場合は、**不正解 [3]** と判定され、プログラムの実行は終了される。

プログラムがこのルーチンを呼び出したとき、マス (R_P, C_P) の高度が X に等しければ **正解**、そうでなければ **不正解 [4]** と判定され、プログラムの実行は終了される。

Rescue ルーチンがこのルーチンを 1 回も呼び出さずに終了した場合は、**不正解 [5]** と判定され、プログラムの実行は終了される。

コンパイル・実行の方法

作成したプログラムをテストするための採点プログラムのサンプルが、コンテストサイトからダウンロードできるアーカイブの中に含まれている。このアーカイブには、提出しなければならないファイルのサンプルも含まれている。

採点プログラムのサンプルは 1 つのファイルからなる。そのファイルは `grader.c` または `grader.cpp` である。作成したプログラムをテストするには、次のようにコマンドを実行する。

- C の場合

```
gcc -O2 -lm grader.c mountain.c -o grader
```

- C++ の場合

```
g++ -O2 grader.cpp mountain.cpp -o grader
```

コンパイルが成功すれば、`grader` という実行ファイルが生成される。

実際の採点プログラムは、採点プログラムのサンプルとは異なることに注意すること。このプログラムは、標準入力から入力を読み込み、標準出力に結果を出力する。



入力

採点プログラムのサンプルは標準入力から以下の入力を読み込む。

- 1 行目には整数 R, C, R_S, C_S, X が空白を区切りとして書かれており、山を表すマス目の縦の大きさが R 、横の大きさが C 、頂点がマス (R_S, C_S) 、遭難者のいるマスの高度が X であることを表す。
- 続く R 行には山の高度の情報が書かれている。そのうちの i 行目 ($1 \leq i \leq R$) には C 個の整数が書かれており、そのうちの j 番目 ($1 \leq j \leq C$) の整数はマス (i, j) の高度を表す。

出力

プログラムの実行が正常に終了した場合、採点プログラムのサンプルは標準出力へ以下の情報を 1 行で出力する。

- 正解の場合、"Accepted"と出力される。(引用符は実際には出力されない。以下同様である。)
- 不正解の場合、不正解の種類が「実装の詳細」の節に書かれた番号によって"Wrong Answer[1]"のように出力される。さらに、不正解 [4] については、正しい遭難者の位置 (R_X, C_X) および Pinpoint ルーチンが呼び出された際に引数によって指定されたマス (R_P, C_P) の情報が "Wrong Answer[4] : $(R_X, C_X) = (3, 2), (R_P, C_P) = (4, 1)$ " のように出力される。

制限

すべての入力データは以下の条件を満たす。

- $1 \leq R \leq 200$
- $1 \leq C \leq 200$
- $1 \leq R_S \leq R$
- $1 \leq C_S \leq C$
- $1 \leq X \leq 1\,000\,000\,000$

小課題

小課題 1 [20 点]

以下の条件を満たす。

- $R \leq 50$
- $C \leq 50$



小課題 2 [80 点]

追加の制限はない.

やりとりの例

採点プログラムのサンプルが読み込み入力の例と、それに対応するルーチンの呼び出しの例を以下に示す.

入力
5 5 3 3 76
14 59 84 62 28
15 73 92 76 35
68 97 100 89 75
58 67 86 79 55
17 25 71 10 5

呼び出し	返り値
Measure(1, 1)	14
Measure(3, 5)	75
Measure(2, 4)	76
Pinpoint(2, 4)	なし

この例は、必ずしも意味のあるルーチン呼び出しをしているとは限らないことに注意せよ.