生日礼物

gift

【问题描述】

今天是CC姐姐的生日，小K给姐姐精心准备了一份生日礼物。不过为了刁难一下姐姐，他不想把礼物直接送给她，而是让她自己找到这份礼物。

CC姐姐的生日聚会上，小K神秘兮兮地搬出了 n 只小宝箱，它们的重量各不相同。小K事先公布了一部分宝箱之间的轻重关系，然后告诉CC姐姐生日礼物已经装在了次重的宝箱里，不过具体是哪个宝箱，让她自己去找。CC姐姐手边仅有的称量工具就是一架天平，每次可以比较两只宝箱的重量。她不仅想找到那只次重的宝箱，而且想用尽量少的称量次数找到它，因为小K告诉她，如果她的策略能保证在最坏情况下称量次数最少的话，她还会得到意外的惊喜！

【输入文件】

第一行是一个整数 n，表示宝箱的个数。

第二行是一个整数 m，表示事先公布的宝箱间的轻重关系数目。

接下来的 m 行，每行两个整数 x 和 y（$1 \le x, y \le n$）表示事先公布了宝箱 x 要重于宝箱 y。

【约定】

- $2 \le n \le 200000$
- $2 \le m \le 200000$
- 事先公布的关系没有自相矛盾

【交互方法】

本题是一道交互式题目，你的程序只可以访问输入文件 gift.in 以及和测试库进行交互。

输入文件格式如前所述。

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。
- compare(x, y)的作用是比较第 x 只宝箱和第 y 只宝箱的重量，$1 \le x, y \le n$。若第 x 只宝箱比第 y 只宝箱重，返回 1，否则返回-1。
- answer(x)的作用是向测试库报告结果，x 表示次重的宝箱的编号。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 gift_lib_p;
```

测试库使用的函数原型为：

```
procedure init;
function compare(x, y: longint): longint;
procedure answer(x: longint);
```

【对使用 C/C++选手的提示】

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

```
#include "gift_lib_c.h"
```

测试库使用的函数原型为：

```
void init();
int compare(int, int);
void answer(int);
```

【你如何测试自己的程序】

- 除了按照上述格式建立 gift.in 之外，请另外建立一个名为“gift.dat”的文件。该文件包含 *n* 个互不相同整数，绝对值不超过 $2*10^9$，依次表示每只宝箱的重量。
- 执行你的程序，此时测试库会产生输出文件 gift.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，gift.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，我们不保证 gift.log 中的内容有意义。
- **正式测试时，测试库会自行生成宝箱的重量，并使得每次回答尽量对你不利。**

【样例】

gift.in 内容如下

```
3
1
3 2
```

gift.dat 内容如下

```
10 1 100
```

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
init;	init();	初始化程序
compare(3,1);	compare(3,1);	返回 1
compare(2,1);	compare(2,1);	返回-1
answer(1);	answer(1);	1 确实是次重的宝箱，共称量 2 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：

- 自行终止；
- 非法调用库函数；
- 让测试库异常退出。

否则，该测试点的得分按如下公式计算：

$$YourScore=\begin{cases}0 & YourAnswer>n+\lceil\log_2 n\rceil\\ 1 & YourAnswer>BestAnswer+2\\ 4 & YourAnswer=BestAnswer+2\\ 7 & YourAnswer=BestAnswer+1\\ 10 & YourAnswer=BestAnswer\end{cases}$$

其中 *YourAnswer* 为你的程序的称量次数，*BestAnswer* 是最佳策略下的称量次数。

【提示】

输入量相当大，请自行选择高效的文件读入方式。