

2008 Canadian Computing Competition

Day 1, Question 1

Problem A – Moving Day

Input: from standard input

Output: to standard output

Source file: `moving.{c, cpp, pas}`

Every year on July 1st, the residents of Quebec experience Moving Day. It is customary for most apartment leases to begin and end on this day, making it a popular time for people to move. The streets are filled with moving trucks and anxious tenants packing and unpacking their belongings, waiting for their truck to become available, or waiting for a previous tenant to vacate their apartment. It is a busy but profitable time for moving companies.

You will write a program to help the residents of a small town plan the order in which they should move. Only one truck is available, so all of the people who are moving must share it. Therefore, the people must move one at a time. A person cannot move into a new house before the former tenant has moved out. Each person moves directly from their old house to their new house; a person may not move temporarily into another vacant house while waiting for their new house to be vacated.

You may assume that no two people are moving into the same house. You may assume that no two people are moving out of the same house. You may also assume that each house to which someone is moving is either vacant or being vacated on Moving Day.

Input description

The first line of input contains the number n ($1 \leq n \leq 100$) of people who are moving. Each of the following n lines of input contains a person's name followed by the address that the person is moving from and the address that the person is moving to. Because there is only one street in the town (called Main St.), each address is just a number between 1 and 100, inclusive. You can assume that no name will be longer than 100 characters, and that names contain only alphanumeric characters.

Output description

The program should print the names of the people who are moving in the order that they should move. If it is not possible to find an order that ensures that each person's new house is vacant by the time the person moves to it, the program should print only the word "Impossible". If there are several different orders in which the people could move, any valid order is acceptable.

Sample input

3
Pierre 51 43
Guy 28 83
Marie 43 28

Sample output

Guy
Marie
Pierre

Grading

All test cases will have $1 \leq n \leq 100$. Your solution must use at most 512 MB of memory and run in at most 3 seconds.

2008 Canadian Computing Competition

Day 1, Question 2

Problem B – King & Weber

Input: from standard input

Output: to standard output

Source file: `kingweber.{c, cpp, pas}`

It is easy to get lost in Kitchener-Waterloo. Many streets that are mostly parallel actually intersect, sometimes multiple times. The best-known example is King and Weber Streets. Other examples include Westmount and Fischer-Hallman, University and Erb, and Queen and Highland.

Navigation is easier in cities that respect the “Manhattan Assumption”: all streets are straight lines in a Euclidean plane, and any two streets are either parallel or perpendicular to each other. Visitors to Manhattan are cautioned that even Manhattan itself does not fully satisfy this assumption.

The input to your program will be a sequence of observations followed by a sequence of queries for a particular city. An observation asserts either that two streets are parallel, or that they intersect. A query asks whether two streets are parallel, or whether they intersect, provided the city satisfies the Manhattan Assumption.

Input description

The first line of input contains two integers m and n ($1 \leq m, n \leq 100000$). Each of the following m lines contains an observation. Each observation consists of three words separated by spaces: the two street names, and either the word `parallel` or the word `intersect`. Each street name is a sequence of no more than 100 uppercase or lowercase letters. The observations are followed by n queries, each on a separate line. A query consists of two street names separated by a space.

Output description

If it is impossible for the city to conform to both the Manhattan Assumption and the specified observations, output a single line containing the word `Waterloo`. Otherwise, output n lines containing the answers to the n queries. Each answer should be one of the following three words: `parallel`, `intersect`, `unknown`. If the two streets queried are parallel in every city satisfying the given observations and the Manhattan Assumption, the output should be `parallel`. If they are perpendicular in every such city, the output should be `intersect`. If they are parallel in some such city and perpendicular in another such city, the output should be `unknown`.

Sample input 1

```
3 3
fourthstreet fifthstreet parallel
fifthstreet sixthstreet parallel
fourthavenue fifthstreet intersect
sixthstreet fourthstreet
sixthstreet fourthavenue
sixthstreet King
```

Sample output 1

```
parallel
intersect
unknown
```

Sample input 2

```
2 1
King Weber parallel
King Weber intersect
King Weber
```

Sample output 2

```
Waterloo
```

Grading

You can assume that 20% of the test cases will have $1 \leq m, n \leq 100$. All test cases will have $1 \leq m, n \leq 100000$. Your solution must use at most 512 MB of memory and run in at most 3 seconds.

2008 Canadian Computing Competition

Day 1, Question 3

Problem C – Mobile

Input: from standard input

Output: to standard output

Source file: `mobile.{c, cpp, pas}`

Fred is a baby. Above Fred’s crib hangs a mobile. Fred is amused by this mobile. Fred has a twin sister, Mary. Above Mary’s crib hangs another mobile. Fred wonders whether the mobile above his crib and the mobile above Mary’s crib are the same. Help Fred.

A mobile is a collection of bars, strings, and decorative weights suspended from the ceiling. Each bar is suspended by a string tied to the exact centre of the bar. From each end of a bar hangs a string that is tied either to another bar or to a weight. The bars can rotate freely about their centres. Fred cannot tell two bars apart, even if they have different lengths. Fred also cannot tell two strings apart. Fred therefore considers two mobiles to be the same if the bars of one mobile can be rotated somehow to make the two mobiles appear identical.

Fred has even developed a notation for describing mobiles. He assigns each bar a distinct positive integer from 1 to the number of bars in the mobile, and he assigns the various objects negative integers. 1 always represents the bar suspended from the ceiling. (So, for example, a biplane might be represented by Fred as object -2 , a crescent-moon might be object -57 , and a star might be object -21 .) Fred can only count down to -9999 , so you can assume that he gave no objects lower numbers than -9999 .

Input description

The input contains two mobile descriptions. The first line of a mobile description contains a single nonnegative integer n ($1 \leq n \leq 100000$), indicating the number of bars in the mobile. On the next n lines, there are two numbers per line, with these two numbers representing the objects hanging from bar i .

Output description

Output is composed of one line. Write “Fred and Mary have different mobiles.” if Fred’s information is enough to distinguish the two mobiles; otherwise, “Fred and Mary might have the same mobile.”.

Sample input 1

```
5
2 3
4 5
```

```
-1 -2
-3 -4
-5 -6
5
2 5
-1 -2
-3 -4
-5 -6
3 4
```

Sample output 1

Fred and Mary might have the same mobile.

Sample input 2

```
5
2 3
4 5
-3 -4
-1 -2
-5 -6
5
2 5
-1 -2
-3 -4
-5 -6
3 4
```

Sample output 2

Fred and Mary have different mobiles.

Grading

Half of all marks will have $n \leq 200$. 75% of the marks will be from test cases with $n \leq 5000$. All test cases will have $n \leq 100000$.

Your solution must use at most 1 second of CPU time and at most 512MB of memory.