South African Computer Olympiad
Camp 3, 2008
Day 1

Overview

Author	Charles and Ben	Max Rabkin	Marco Gallotta
Problem	**abe**	**election**	**flood**
Source	abe.c abe.cpp abe.pas	election.c election.cpp election.pas	flood.c flood.cpp flood.pas
Input file	stdin	stdin	stdin
Output file	stdout	stdout	stdout
Time limit	1 second	1 second	2 seconds
Number of tests	20	20	20
Points per test	5	5	5
Total points	**100**	**100**	**100**

The maximum total score is 300 points.

Abe The Obsessive-Compulsive Street Sweeper

Author

Charles and Ben

Introduction

Abe the obsessive compulsive street sweep has just taken a course on seamanship, and wants to show off his new found knowledge that Port (that is left) is red and that Starboard (i.e. right) is green. He wants to paint the flagstones on his street so that every one walking down it must put their left feet on red stones and their right feet on the green ones. This is made more difficult by the fact that Bruce's house is on the road, and hence normal laws of physics do not apply.

Task

Given the number of stones, N, and a list of E edges between adjacent stones, tell Abe whether he can paint the flagstones such that every red flagstone is adjacent to only green flag stones and every green flagstone is adjacent to only red flagstones, and if so how many stones will have each colour.

It is always possible to reach any stone from any other, but not necessarily using a path which alternates in colour.

Example

Assuming the following stones are adjacent: 0 and 1, 1 and 2, 2 and 3, 3 and 4, 1 and 4. If the stones 0, 2 and 4 are painted red and the stones 1 and 3 are painted green, then the street will satisfy Abe's flagstone-painting fetishes

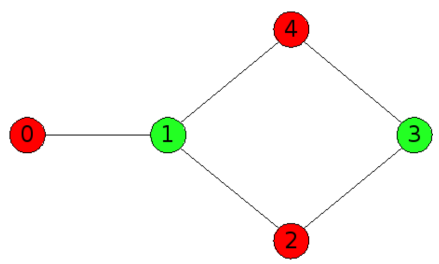

Figure 1: Example Street

Input (stdin)

The first line of the input contains two space-separated integers, N and E. The next E lines each contain 2 space-separated integers, S_1 and S_2, two adjacent flagstones.

Sample input

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

Output (stdout)

If Abe can paint the street as he wants, output the number of stones to be painted red or green (whichever is smaller), else output "BUMMER DUDE".

Sample output

```
2
```

Constraints

- $1 \leq N \leq 10000$
- $0 \leq E \leq 100000$

Time limit

1 second.

Scoring

For outputting any answer Abe will give you 100% however if he later learns you lied to him he will deduct all the marks he gave to you and you will be left with 0%

Bruce for President

Author

Max Rabkin

Introduction

The year is 2038, and tired of a world ruled by incompetents, Bruce has decided to run for the most powerful political office in the world: the Presidency of the United States of America.

The fact that he is not a natural-born citizen, as required, shall be no obstacle for him, for Bruce was not born but made.

Being a computer programmer, Bruce is lazy, and thus wants to campaign in as few states as possible.

Task

The American electoral system is rather complex, but it boils down to this: each state is worth a certain number of points (called electoral college votes) and the winner of the state's election wins that many points. If a candidate receives more than half the available points, he or she becomes president.

By 2038, a number of new states had been formed (for example, Canada has become the State of East Alaska and Mexico has become the State of New New Mexico) and there are now N states.

You must find out how many minimal winning sets of states there are. A set of states is winning if it contains over half the total number of electoral college votes. A winning set is minimal if no state can be removed from it and still be a winning set. As this may be a large number, give the answer modulo 1 000 000 007.

Example

The USA has three states: New New Mexico with four E.C. votes, East Alaska with three and Rhode Island with two. So a candidate needs five E.C. votes to win.

Here any set of two or three states is a winning set, but obviously the set of all three is not minimal, so there are three minimal winning sets (each pair of states).

Input (stdin)

The first line of the input contains an integer N. The next N lines each contain an integer V_i: the number of E.C. votes for the ith state.

Sample input

```
3
4
3
2
```

Output (stdout)

The output consists of one line containing a single integer, the number of minimal winning sets modulo 1 000 000 007.

Sample output

```
3
```

Constraints

- $1 \leq N \leq 1\,000$
- $1 \leq V_i \leq 100$

Additionally, in 50% of the test cases:

- $1 \leq N \leq 100$

Furthermore, in 20% of the test cases:

- $1 \leq N \leq 16$

Time limit

1 second.

Scoring

You know the drill. Right answers score, wrong answers don't.

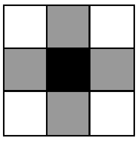

Figure 2: The four neighbours (grey) of the black cell.

Flood Zone

Author

Marco Gallotta

Introduction

It's all over the news – the Japanese are dropping water bombs all over the United Kingdom! Why water bombs you may ask? Well who cares, they're causing great floods and they're not far from the town of Cambridge where Bruce lives!

Unfortunately Bruce has become so stressed by the impending disaster that he's gone into mental melt-down. In you step to save the day, or will you?

Cambridge recently started upgrading their drainage system after hearing about the development of the water bomb. They installed a number of super-drains that suck in any mass of water that flow over them instantly, giving Bruce some hope of survival.

Task

The town can be represented by a 2D grid with each unit cell having a fixed height. With the height map and the location of the super-drains and where the bomb hits, your task is to determine the flood zone, which Bruce should avoid at all costs.

When the bomb hits its target it starts a flooding process which releases an unimaginable quantity of water onto a single cell. Each cell has four neighbours, illustrated in Figure 2. Water from a cell will always flood neighbouring cells that are *not* higher. When the water from the bomb can no longer spread outwards, the water level will increase until more flooding can occur.

Fortunately though, the super-drains instantly suck in all water that reaches the cells they are on such that no water can pass over them. As the water flows through the town most of it will be eventually drained out, while some will accumulate in cells which will never be drained. The flood zone is the cells which will retain water after all the water has stopped flowing out of the bomb. There is sufficient water in the bomb to flood any cells that are possible to flood.

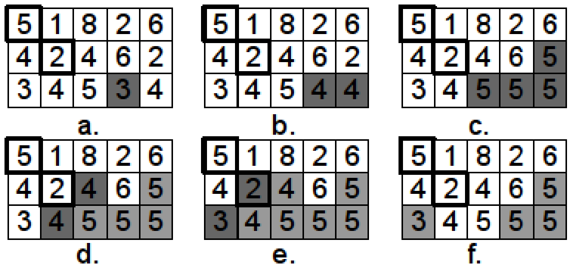

Figure 3: Example flooding. Outlined cells indicate super-drains, dark grey indicates newly flooded cells and light grey indicates flooded cells.

Example

Consider the height map and super-drains illustrated in Figure 3. The bomb hits the grey cell in state a in the figure. As the water has nowhere to go it accumulates where the bomb hit until it is able to spread in states b and c (at which point the water in all flooded cells is 5 units high).

The water then flows into the lower cells of height 4 in state d and finally into the two cells of height 2 and 3 in state e. As the water reaches the super-drain in state e it is immediately drained, stopping the water from passing it onto the cell of height 1. The water in the higher, surrounding cells flows into the drain. Once the water has stopped flowing from the bomb all water in these surrounding cells will be drained, leaving four flooded cells in state f. This is the flood zone.

Input (stdin)

The first line contains two integers N and M, the size of the town. The next N lines each contain M integers, which describes the height map of the town. The j^{th} integer on the i^{th} line is the height of cell i, j.

The next line contains a single integer S, the number of super-drains. The next S lines each describe the location of a super-drain with two integers. The last line describes the cell that the bomb hits.

Sample input

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

Output (stdout)

Output an $N \times M$ grid of boolean values, with a 1 indicating where water will accummulate and a 0 indicating a safe zone.

Sample output

```
0 0 0 0 0
0 0 0 0 1
0 0 0 1 1
```

Constraints

- $1 \leq N, M \leq 1,000$
- $1 \leq S \leq NM$
- $1 \leq$ height of all cells $\leq 1\,000\,000$

Additionally, in 50% of the test cases:

- $N \leq 100$

Time limit

2 seconds.

Scoring

A correct solution will score 100%, while an incorrect solution will score 0%.