Old Mutual - CSSA Computer Olympiad DAY 1

Overview

Problem	Flood	Order	Letters
Author	Graham Poulter	Bruce Merry	Bruce Merry
Program name	flood.exe	order.exe	letters.exe
Source name	flood.pas flood.cpp flood.c flood.java	order.pas order.cpp order.c order.java	letters.pas letters.cpp letters.c letters.java
Input file	flood.in	order.in	letters.in
Output file	flood.out	order.out	letters.out
Time limit per test	2 seconds	1 second	none
Number of tests	10	10	10
Points per test	10	10	10
Total points	**100**	**100**	**100**

The maximum total score for Day1 is 300 points.

Old Mutual - CSSA Computer Olympiad DAY 1

Flood

Author

Graham Poulter

Introduction

It was a time of drought and the Guji tribe called on their wise man to do his rain dance. He danced the entire day, and in the evening it began to rain. The wise man is very good, and can even calculate the exact volume of rain that he is calling down. Unfortunately he may have been a bit overzealous — the Guji live on a hill within a valley, which is now rapidly filling with water all around them. The chief has called on you to find the height to which the water will rise so he knows whether he can sleep in tomorrow or will have to call an emergency evacuation of the village.

Task

You are given a height map, which is an M by N array of values representing a grid across the Guji tribal land. The value of an array element is the height of the land in that grid block. The terrain can be thought of as being assembled out of unit cubes, and the height in a grid cell is the number of cubes stacked on top of each other in that cell.

Now imagine a box around this terrain, and that you pour an integer volume of water, V (representing V unit cubes of water) into it, so that land becomes submerged from the bottom up. The volume of water will always fill a level (the data sets won't leave a level only part full with a step in it). You are required to write a program that calculates the height of the water level, and the number of terrain cubes whose volume lies below the water level (that is, the total volume of land below the plane marking the top of the water).

Input (flood.in)

The first line contains three integers: M, N and V, representing the dimensions of the terrain to follow, and the volume of water being poured into it. The next M lines each contain N integers H[i, j] (row i, column j), representing the height map of the terrain.

Sample input:

```
4 5 33
2 2 2 2 2
1 3 4 3 2
2 3 5 3 2
2 4 1 1 2
```

This corresponds to a 4 row by 5 column height map into which 33 unit cubes of water are to be poured.

Output (flood.out)

The first line contains two integers: the height of the water level, L, and the number of terrain blocks below the water level, B.

Sample output:

```
4 47
```

The value L=4 corresponds to a water level that is level with the top of the block in row 2, column 3 in the input, and B=47 is the number of blocks with their volume below the water level (that is, the lone block on top of row 3, column 3 is the only one whose volume is above water).

Constraints

1 • M, N • 400
1 • H[i, j] • 10000 where 1 • i • M, 1 • j • N
0 • V, L, B • 1000000000

Time limit

2 seconds.

Scoring

There are 10 test cases of 10 marks each. 5 marks is awarded for the correct water level, L, and 5 marks for the correct number of blocks whose volume is below the water level, B.

Old Mutual - CSSA Computer Olympiad DAY 1

Order

Author

Bruce Merry

Introduction

The children in the Guji tribe have a game they like to play. One child sets up a number of objects in a row. One of the other children has to guess what order they are in, by asking only questions of the form "is A before B?"

After watching them for a while, you realise that some of them are not very good at it. They tend to ask questions for which they could work out the answers. For example, if they know that A is before B, and B is before C, then they could deduce that A is before C and there is no need to ask the question.

Task

You will be given a list of facts of the form "A is before B". Some of these facts can be deduced from the other facts in the list (those appearing both before and after the given fact). Such facts are called *redundant* facts. You must find all the non-redundant facts in the given information.

The given facts will never contradict each other. In other words, there is at least one arrangement of the items that makes all the facts true (you do *not* need to find such an arrangement, however).

Example

Suppose there are 5 items, numbered 1 to 5. You are given that $3 < 5$, $4 < 2$, $5 < 2$, $2 < 1$, $3 < 1$ and $4 < 1$ (here "<" means "comes before"). The fact $3 < 1$ is redundant since $3 < 5$, $5 < 2$, $2 < 1$. Similarly $4 < 1$ is redundant because $4 < 2$, $2 < 1$. The remaining facts are all non-redundant.

Input (order.in)

The first line of input contain two space-separated integers, N and F. N is the number of items (which are numbered 1 to N) and F is the number of facts that will be given. Each of the next F lines contains two space-separated integers X and Y. This indicates that item X comes before item Y. No facts will be repeated in the input file.

Sample Input:

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

Output (order.out)

The first line of output contains U, the number of non-redundant facts. The next U lines each describe one of these facts, in the same format as the input file. You may output the facts in any order but may not repeat them.

Sample output:

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

Constraints

$1 \bullet N \bullet 1500$
$0 \bullet F \bullet 10000$

Time limit

1 second.

Scoring

Let V be the true number of non-redundant facts, and U be the number you claim in your output file. You will score 0 if

- Your output file is incorrectly formatted
- You list any facts not present in the input
- You omit any non-redundant facts
- You duplicate any facts in the output
- $F - V < 5$ and you list any redundant facts

If none of the above applies, then your score (in percent) will be $100 - 20(U - V)$, or 0 if this is negative.

Old Mutual - CSSA Computer Olympiad DAY 1

Letters

Author

Bruce Merry

Introduction

The Guji tribe has a rather curious writing system. They use symbols to represent their letters. However, a single symbol can correspond to several possible letters! This makes their writing rather ambiguous, as several different words could be written in the same way. To reduce the number of ambiguities, they redesign the writing system every few years by changing the assignment of letters to symbols.

The Guji language has L letters. For the purposes of this problem, they will be A, B, C, …, up to the Lth letter of the English alphabet. They use N symbols, which are numbered 1, 2, …, N. One part of the writing system they never change is that if two letters are in alphabetical order, then their symbols are in numerical order. For example the symbols for C and Q could not be 5 and 3 respectively, because C, Q are in alphabetical order while 5, 3 are not in numerical order.

Task

You have been given a Guji dictionary, and asked to help the Guji find the best way to assign symbols to the letters. The best way is defined as the one where the most dictionary words are unambiguous when written down (i.e. no other word in the dictionary is written in the same way). There may be several ways that are equally good; if so you can choose any one of them.

Input

The first line contains N and L, separated by a single space. The second line contains D, the number of words in the Guji dictionary. The next D lines list the words in the dictionary, one per line. The dictionary is in alphabetical order with no words repeated. Words are given in uppercase and contain only the letters A to X, where X is the Lth letter of the English alphabet.

Sample input

```
3 13
11
ALL
BALL
BELL
CALK
CALL
CELL
DILL
FILL
FILM
ILL
MILK
```

Output

The first line of output is U, the number of words in the dictionary that correspond to unique symbol sequences with your chosen symbol assignment. The remaining N lines specify your symbol assignment. The Kth of these lines contains the letters assigned to symbol number K, in alphabetical order and uppercase, with no spaces in between.

Sample output

```
7
AB
CDEFGHIJK
LM
```

The words CELL, DILL, FILL and FILM are all written 2233, so are not unique. The remaining 7 words are unique.

Constraints

- 1 • D • 50000
- No word contains more than 10 letters.

Scoring

You will score zero if your symbol assignment is illegal or if your value of U does not correspond to your assignment. Otherwise let V be the optimal value for U. Your score in percent will be 500U/V – 400, or 0 if this is negative. Fractions are rounded downwards.