South African Computer Olympiad
Training Camp 1, 2005
Day 2

Overview

Author	**Bruce Merry**	**Carl Hultquist**	**Carl Hultquist**
Problem	**tomb**	**museum**	**area**
Source	tomb.pas tomb.c tomb.cpp tomb.java	museum.pas museum.c museum.cpp museum.java	area.pas area.c area.cpp area.java
Input file	tomb.in	museum.in	area.in
Output file	tomb.out	museum.out	area.out
Time limit	1 second	5 seconds	5 seconds
Number of tests	10	10	10
Points per test	10	10	10
Total points	**100**	**100**	**100**

The maximum total score is 300 points.

Tomb Raider

Author

Bruce Merry

Introduction

Lara Croft is trying to get to the Holy Grail ahead of the Union Aerospace Corporation, who would use it in dangerous secret experiments on Mars. She has made it to the inner temple, but faces one last lock to in order to reach it. Help her to open the lock and save the world.

Task

The lock consists of an $M \times N$ grid of touchstones and a great big lever (GBL). She must push in some of the touchstones, then pull the GBL. If she gets it right, the door will open and she can escape with the Holy Grail. If she pushes the wrong touchstones, she will be cut into tiny pieces and the universe will be doomed.

Fortunately, she has deciphered some ancient writings and found instructions regarding which touchstones to push. The correct touchstones are *connected*: for every pair of correct touchstones, there is a path between them (moving horizontally or vertically at each step) that covers only other correct touchstones.

There is also a number on each stone. The sum of the numbers must be as close to a power of two as possible, except that at least one stone must be pushed. Where there is a tie, the sum must be as large as possible.

Example

Suppose the grid of touchstones is as shown in figure 1. There is no way to get an exact power of two ($13 + 3$ is illegal because it is not connected). We can obtain many numbers one away from a power of two (3, 9, 15, $31 = 3 + 6 + 9 + 13$ and $33 = 3 + 9 + 6 + 15$), but 33 is the largest.

3	9	6
6	13	15

Figure 1: Example set of touchstones

Input (tomb.in)

The first line contains two positive integers, M and N, separated by a space. M is the number of rows in the grid and N the number of columns. The following M lines each contains N positive integers separated by spaces, which are the numbers on the touchstones in each row.

Sample input

```
2 3
3 9 6
6 13 15
```

Output (tomb.out)

The output consists of M lines of N integers, separated by spaces. Each integer is either a 0 or a 1, with a 1 indicating that Lara should push the corresponding touchstone. If there is more than one way to produce the correct sum, you may output any one of them.

Sample output

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

Constraints

- $1 \leq$ each touchstone ≤ 1000
- $1 \leq MN \leq 22$

50% constraints

- $1 \leq MN \leq 10$

Time limit

1 second.

Scoring

100% for a correct answer, 0% for an incorrect answer.

Museum Security

Author

Carl Hultquist

Introduction

Donald has just been appointed as the curator of the world's largest art museum, which houses many rare art works from around the world. At the moment, there is no security system installed, and Donald is worried that this could allow for visitors to the museum to steal some of the artwork. So, as his first major job, Donald wants to install an alarm system to monitor all of the work in the museum.

Each piece of artwork will be monitored by its own electronic alarm sensor, which will sound the alarm if the artwork is moved. All of the sensors have been installed, and all that remains is to connect power to all of the sensors. To do this, Donald will connect one sensor into the museum's power generator, and he then wants to connect pairs of sensors using wires so that there every sensor receives power through the connection network. That is, if sensor A has power and Donald connects it to sensor B, then sensor B will also have power.

To conserve costs, Donald wants to find the cheapest way of connecting up the sensors so that they all have power. Your task is to help Donald work out what the cheapest cost is, and which pairs of sensors should be connected.

Task

Due to the museum design, some pairs of sensors cannot be directly connected by a power wire. You must write a program that will read in the number of sensors, and then read in information about all the possible pairs of sensors that *could* be connected, as well as how much these connections would cost. Your program must then calculate the cheapest way to connect up all the sensors, and must output this cost followed by a list of which connections should be made. Donald will always connect the museum's power generator to the sensor guarding the first piece of artwork.

Example

When Donald first experimented with this idea, he tried it in a section of the museum that housed 5 pieces of artwork. There were 7 possible connections between the sensors, which are shown in Table 1.

Sensor number	Sensor number	Cost
1	2	3
1	3	2
1	4	5
1	5	10
3	4	4
3	5	7
4	5	1

Table 1: An example of the connections that could be made in a small part of the museum with 5 pieces of artwork

The cheapest way for all the sensors to receive power would be to connect the following pairs of sensors: 1 and 2, 1 and 3, 3 and 4, 4 and 5. This gives a total cost of 10.

Input (museum.in)

The first line of input will contain two space-separated integers, N and M, which are the number of sensors and the number of possible connections respectively. The sensors are numbered from 1 to N. The next M lines of input will each contain three space-separated integers, A, B and C, indicating that sensors A and B could be connected with a cost of C.

Sample input

```
5 7
1 2 3
1 3 2
1 4 5
1 5 10
3 4 4
3 5 7
4 5 1
```

Output (museum.out)

The first line of output must contain two space-separated integers, T and X (in this order), where T is the smallest total cost to connect up all the sensors, and X is the number of connections that were needed to achieve this cost. The next X lines of output will describe the connections needed to achieve this cost: each must contain two integers, A and B, indicating that sensors A and B must be connected.

Sample output

```
10 4
4 5
1 3
1 2
3 4
```

Constraints

- $1 \leq N \leq 20000$
- $N \leq M \leq 100000$
- $1 \leq C \leq 1000$ for any of the cost values C in the input

50% constraints

- $1 \leq N \leq 1000$
- $N \leq M \leq 50000$

Time limit

5 seconds.

Scoring

Your program will score 0% for a test-case if at least one of the following applies:

- Your output does not conform to the description specified above
- The set of connections in your output does not connect all of the sensors
- The actual total cost of the connections in your output does not match the total cost specified in your output

Your program will score 100% for a test-case if your total cost is the smallest cost possible to connect the sensors. If your total cost is more than the smallest cost possible, then your program will score $50\frac{S}{T}\%$ for the test-case, where S is the smallest cost possible and T is the total cost in your output.

Area

Author

Carl Hultquist

Introduction

Jack and Jill are playing a game, where Jill throws lots of rectangular pieces of paper onto the floor and Jack needs to work out the exact area covered by the pieces of paper. Luckily, the floor is also rectangular and all of the pieces of paper have their edges parallel to the edges of the floor.

Task

Your task is to help Jack work out the floor area covered by the pieces of paper. You will be given the position and size of every piece of paper, and the floor's lower-left corner is at position (0 ; 0), its upper- right corner is at position (10000 ; 10000).

Example

Suppose that Jill throws four pieces of paper onto the ground, which have the following positions for their lower-left and upper-right corners:

Lower-left	Upper-right
(1 ; 2)	(3 ; 4)
(1 ; 1)	(2 ; 2)
(0 ; 6)	(7 ; 8)
(2 ; 3)	(5 ; 7)

These pieces of paper have individual areas of 4, 1, 14 and 12. The actual floor area covered by the pieces of paper is 27.

Input (area.in)

The first line of input will contain a single integer, N, which specifies the number of pieces of paper that Jill has thrown onto the floor. The next N lines will each contain four space-separated integers, x_1, y_1, x_2 and y_2 which specify the bottom-left and upper-right co-ordinates of each piece of paper

Sample input

```
4
1 2 3 4
1 1 2 2
0 6 7 8
2 3 5 7
```

Output (area.out)

Your program must output a single integer, A, which is the area of floor covered by the pieces of paper.

Sample output

```
27
```

Constraints

- $0 \leq x_i ; y_i \leq 10000$ for any of the co-ordinate values x_i and y_i
- $2 \leq N \leq 10000$

50% constraints

- $0 \leq x_i ; y_i \leq 1000$ for any of the co-ordinate values x_i and y_i
- $2 \leq N \leq 1000$

Time limit

5 seconds.

Scoring

If your program outputs the correct answer for a test-case, then you will score 100% for that test-case. Otherwise, you will score 0% for that test-case.