

IOI '94 P2 - The Castle

Time limit: 1.0s Memory limit: 8M

IOI '94 - Haninge, Sweden

```
  1  2  3  4  5  6  7
#####
1 #  |  #  |  #  |  |  #
   #####---#####---#---#####---#
2 #  #  |  #  #  #  #  #
   #---#####---#####---#####---#
3 #  |  |  #  #  #  #  #
   #---#####---#####---#---#
4 # ->#  |  |  |  |  #  #
   ##### (Figure 1)
```

= Wall -,| = No wall
-> = It points to the wall to remove to
 make the largest possible new room

Figure 1 shows the map of a castle. Write a program that calculates:

1. How many rooms the castle has,
2. How big the largest room is,
3. The size of the largest room creatable by removing one wall, and
4. Which wall to remove from the castle to make as large a room as possible.

The castle is divided into $N \times M$ ($1 \leq N, M \leq 50$) square modules. Each such module can have between zero and four walls. The castle always has at least two rooms and a wall that can be removed.

Input Specification

The map is stored in the form of numbers, one number for each module, M numbers on each of N lines to describe the floorplan. The input order corresponds to the numbering in the example diagram above.

- The input starts with two space separated integers M and N .
- The following N lines each contains M integers from 0 to 15 that each describe a single module of the castle.

Each module number indicates which of the four walls exist, and is the sum of up to four integers:

- 1: wall to the west
- 2: wall to the north
- 4: wall to the east

- 8: wall to the south

Output Specification

The output should contain 4 lines.

- Line 1: The number of rooms the castle has
- Line 2: The size of the largest room
- Line 3: The size of the largest room creatable by removing one wall
- Line 4: The single wall to remove to make the largest room possible

Choose the optimal wall to remove from the set of optimal walls by choosing the module farthest to the west (and then, if still tied, farthest to the south). If still tied, choose **N** before **E**. Name that wall by naming the module that borders it on either the west or south, along with a direction of **N** or **E** giving the location of the wall with respect to the module.

Sample Input

```
7 4
11 6 11 6 3 10 6
7 9 6 13 5 15 5
1 10 12 7 13 7 5
13 11 10 8 10 12 13
```

Sample Output

```
5
9
16
4 1 E
```