International Collegiate Programming Contest
ACM-ICPC South Western European Regional
Practice Session – SWERC 2008

November, 22 2008

Problem Overview:

Good luck and have fun!

Problem A
Alchemy

Alan is a well known alchemist. As any good alchemist, he also tries to find the philosopher's stone, the mystic substance to turn ordinary metal into gold. To reach this goal, he examines different materials and their properties. He believes that two materials are made from the same base element if they share the same property. Furthermore, he believes that every material has just one base element. Since only the base element seems to be crucial for making gold it is sufficient to try his magic with only one material of that base element. Other materials of that base element probably behave similarly and need not to be examined further.

In the past, he has pursued a long list of experiments to check if pairs of materials have the same property. But unfortunately due to his age, his eyesight is not the best anymore. Therefore, the king sends you to help him to process his list of results. Can you tell him for any pair of two different materials whether or not they are made of the same base element for sure.

Input

The first line specifies the number of test cases t that follow ($t \leq 15$). Each test case starts with three numbers $0 < m, e, q \leq 10000$ on a single line. All materials are assigned unique numbers between 0 and $m - 1$, while e denotes the number of material pairs for which Alan already determined identical properties.
Each of the following e lines hold two numbers $0 \leq a, b < m$, each of which represents the number of a material. Such a pair a and b states that the corresponding materials have the same property and, therefore, both are made of the same base element.
Subsequently, the test case has q additional lines that hold queries. On each of those lines there are two numbers $0 \leq c, d < m$ that again represent two materials. Using the earlier experiments, you must check if they are made of the same base element.

Output

For each query in each test case, you have to answer whether or not the two materials are made of the same base element. If you know for sure they consist of the same base element print on a single line"`Yes`", otherwhise "`No`". Omit the quotes.

Sample Input

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

Sample Output

```
Yes
Yes
No
Yes
```

Problem B
Bridges

In the Middle Ages, towns that were located close to rivers developed much faster than other settlements. Towns that had a bridge got even richer.
Thus, the king of our country asks his top researchers for the best places to build bridges. They found the following constraints:

- Bridges should not be placed in directly neighboring cities along the river.
- It should not be possible to add more bridges without violating the first condition.

It is your job now to count all possible constellations for the optimal placement of bridges. For example, consider five towns upon a river (numbered from 0 to 4 from east to west). Then there are *four* constellations that do not violate the constraints above, namely $\{0, 2, 4\}$, $\{1, 3\}$, $\{1, 4\}$, and $\{0, 3\}$.

Input

The first line contains the number of test cases that follow. Each test case is given in one line that holds the number of towns T that are close to a river in this kingdom ($0 < T < 77$).

Output

For each test case, output the number of possible constellations in a single line. Print a blank line after each test case.

Sample Input

```
4
1
3
5
76
```

Sample Output

```
1

2

4

1828587033
```

Problem C
Robin Hood

You are Robin Hood, one of the best archers around Nottingham. You want to participate in the championship of archery, which takes place today in the castle. Unfortunately, the local ruler does not want you to participate, because he knows that you would win. Wanting his son to win, he has locked you away in the catacombs of the castle.

With the help of your friends you obtained a key to unlock the prisoner's cell, but these catacombs are built like a maze. Now you do not know the shortest way, but are too excited about the upcoming championship to calm down and think clearly.

Since the championship will start in a few minutes, you have to hurry. What is the **longest** path you could take out of the catacombs to the arena, where the championship will take place? Will you be there in time even if you take this longest path?

Input

The input consists of several scenarios. Each scenario starts with two integers h and w in one line ($1 \leq h, w < 15$). h denotes the height and w the width of the following map of the castle's catacombs. The map consists only of ' ' (space) and '`#`' characters. The character ' ' denotes a free floorway cell and '`#`' is a blocked cell (by a wall, a closed door or something else).
After these h lines, there are 2 more lines describing your start position and target position respectively. The position is made up of 2 integers, the x and the y coordinate. The cell in the bottom left corner has coordinates 0 0.
The last line of each scenario contains the number of seconds s that are left before the championship starts. You can assume that you can run from a cell to a non-blocked adjacent cell (only using the 4 directions, not diagonally) in 5 seconds. There are never more than 1,234,567 valid paths from start position to target position, but there is at least one.
The input is terminated with two zeroes on one line.

Output

Assume you take the **longest** path out of the catacombs. But be aware that you are clever enough to never enter any cell twice.
If you can always leave the maze before the start of the championship (therefore *longest running time* $<$ *remaining time*), print "`:)`" otherwise print "`;(`" on one line.

(Sample Input and Output are provided on the next page)

Sample Input

```
6 10
##########
#       ##
#  #### ##
##      ##
# ## ##
##########
2 2
9 1
100
6 10
##########
#       ##
#  #### ##
##      ##
# ## ##
##########
2 2
9 1
70
6 10
##########
#       ##
#  #### ##
##      ##
# ## ##
##########
2 2
9 1
75
5 10
##########
#        #
#        #
#        #
##########
1 1
8 3
113
5 10
##########
#        #
#        #
#        #
##########
1 1
8 3
117
0 0
```

Sample Output

```
:)
;(
:)
;(
:)
```