

UCRPC F21 A - Senseless Census

Time Limit: 1.0s **Memory Limit:** 256M

Count all the Toads!

Look through the train car and count the Toads you see!

Senseless Census is a four-player mini-game in *Super Mario Party* and you can view how the game is actually played [here](#). Each of the four players must count the number of toads on the entire train. The one who counts the most precisely wins the game.



The difficulty of the game is that you can only see a segment of the train car on your screen at a time so you must walk around to see the entire train. At the same time, many of the Toads are annoyingly walking around back and forth. However, we can make this task easier if we can take a snapshot of the train car at different points at the same time. Given a snapshot of the train car, your task is to write a program to count the exact number of Toads.

Input Specification

The first line of the input contains two integers, n and m ($0 \leq n, m \leq 10^3$), which means that the train car can be described as an $n \times m$ matrix.

The next n lines each contain m characters, which are either `-`, `t` or `s`. They describe the snapshot of the train car. `-` means there's nothing, `t` means there's a Toad, and `s` means there's an empty seat or table.

Output Specification

The output only contains 1 integer, which is the number of Toads in the train car.

Sample Input

```
5 6
--t---
--stst
tttsss
----s-
--tsss
```

Sample Output

```
7
```

Explanation for Sample Output

Source of pictures and descriptions: https://www.mariowiki.com/Senseless_Census. You can also find more details about the game from this site.

Scoring

There are 20 test cases, 3 points each.

UCRPC F21 B - Lost in the Shuffle

Time Limit: 1.0s **Memory Limit:** 256M

Find the doll with the most mini-dolls inside!

Try not to blink during the shuffle!

Lost in the Shuffle is a four-player mini-game in *Super Mario Party*, and you can view how the game is played [here](#). In the game, Toad places five matryoshka dolls resembling them on the table. The middle one is a special one since it contains the most mini-dolls inside. Toad will then shuffle the five dolls by swapping pairs of them for several rounds, speeding up gradually. The players will then choose a matryoshka doll and the player that chooses the special one (the one that was originally in the middle) wins the game.



The last several rounds of shuffling go pretty fast, so it's very hard to tell by just staring at the screen. Given all the swaps that will be done in order, can you write a program to decide which one to choose to win the game?

Input Specification

The first line of the input contains one integer, n ($0 \leq n \leq 10^4$), which is the total number of swaps in the game.

In the next n lines, line i contains a pair a_i, b_i ($1 \leq a, b \leq 5$), which means that in the i -th swap, doll a_i will be swapped with doll b_i .

The five dolls are labeled 1 to 5. The middle one (the one with label 3) is our target.

Output Specification

The output only contains one integer, which is the current position of the special doll. Note that the output must be in the range of 1 to 5.

Sample Input

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

Sample Output

```
5
```

Explanation for Sample Output

In the first test case, the special doll was first swapped to position 1 in the first swap. Then in the 3rd swap, it was put at position 2. Then in the 4th swap, it will be moved to position 5. The last swap didn't change its position. Therefore the output is 5.

Source of pictures and descriptions: https://www.mariowiki.com/Lost_in_the_Shuffle. You can also find more details about the game from this site.

Scoring

There are 20 test cases, 4 points each.

UCRPC F21 C - Maths of Glory

Time Limit: 1.0s **Memory Limit:** 256M

Defeat your rivals with superior numbers!

Aim for the 6!

Maths of Glory is a 2v2. minigame in *Super Mario Party*, and you can view how the game is actually played [here](#). Each team has a castle that is made up of n jigsaw pieces, and a cannon that can fire cannonballs at the rival's castle. The goal is to destroy the castle of your rival.

In each round, each player will hit a slot with numbers ranging from 1 to 6. The numbers from the two players in the same team will be multiplied to be the points p earned in the current round. Then the cannon will fire at the rival's castle and destroy p jigsaw pieces of the castle. The first team to demolish the opposing team's castle entirely won the game. Note that if two teams both win at the same round, the game ends in a tie.



Given the numbers each player hits in each round, write a program to determine which team wins the game.

Input Specification

The first line of the input contains two integers, n ($0 < n \leq 10^6$), is the number of jigsaw pieces in each castle, and m , which means there will be m rounds of game shown below. The next m ($0 < m \leq 10^6$) lines each contain 4 numbers, which is the number hit by player 1, player 2, player 3 and player 4. Player 1 and 2 are in a team, and player 3 and 4 are in a team. The 4 numbers are guaranteed to be in the range of 1 to 6.

Output Specification

Output the winning team and when they win. For example, if team 1 wins at round 5, output `Team 1 wins at round 5!`.

If they both win at a certain round x , output `It's a tie at round x!`.

If the game ends with neither team winning, output `Oh no!`.

Sample Input 1

```
20 6
2 4 6 3
3 3 1 5
5 2 2 1
1 1 1 1
6 3 1 4
2 6 4 1
```

Sample Output 1

```
Team 2 wins at round 2!
```

Sample Input 2

```
30 6
2 4 6 3
3 3 1 5
5 2 2 1
1 1 1 1
6 3 1 4
2 6 4 1
```

Sample Output 2

```
It's a tie at round 5!
```

Sample Input 3

```
100 6
2 4 6 3
3 3 1 5
5 2 2 1
1 1 1 1
6 3 1 4
2 6 4 1
```

Sample Output 3

```
Oh no!
```

Explanation for Sample Outputs

For the first test case, after round 2, team one earns 17 points and team 2 earns 23 points. Since both castles only have 20 pieces each, team 2 will win.

For the second test case, there are 30 pieces in each castle and both teams reach 30 points or more at round 5 (46 vs 30). As a result, it's a tie.

For the third test case, after 6 rounds, team 1 earns 58 points and team 2 earns 32 points. Neither of them achieved 100 points to defeat the other team. As a result, you need to output `Oh no!`.

Source of pictures and descriptions: https://www.mariowiki.com/Maths_of_Glory. You can also find more details about the game from this site.

Scoring

There are 20 test cases, 3 points each.