Submissions

Juice

3pt | Not attempted **97/97 users** correct (100%)

10pt | Not attempted **74/93 users** correct (80%)

Ping Pong Balls

4pt | Not attempted **92/97 users** correct (95%)

11pt | Not attempted **18/32 users** correct (56%)

Mine Layer

4pt | Not attempted **85/88 users** correct (97%)

13pt | Not attempted **14/33 users** correct (42%)

Bridge Builders

8pt | Not attempted **69/73 users** correct (95%)

17pt | Not attempted **20/31 users** correct (65%)

The Year of Code Jam

7pt | Not attempted **47/68 users** correct (69%)

23pt | Not attempted **6/9 users** correct (67%)

Top Scores

ACRush	89
Innovative.Cat	89
bmerry	87
pmnox	76
yuhch123	66
gawry	66
Eryx	60
mystic	60
ploh	60
blueblimp	59

Problem A. **Juice**

This contest is open for practice. You can try every problem as many times as you like, though we won't keep track of which problems you solve. Read the Quick-Start Guide to get started.

Small input 3 points	Solve A-small
Large input 10 points	Solve A-large

Problem

You are holding a party. In preparation, you are making a drink by mixing together three different types of fruit juice: Apple, Banana, and Carrot. Let's name the juices A, B and C.

You want to decide what fraction of the drink should be made from each type of juice, in such a way that the maximum possible number of people attending the party like it.

Each person has a minimum fraction of each of the 3 juices they would like to have in the drink. They will only like the drink if the fraction of each of the 3 juices in the drink is greater or equal to their minimum fraction for that juice.

Determine the maximum number of people that you can satisfy.

Input

- One line containing an integer **T**, the number of test cases in the input file.

For each test case, there will be:

- One line containing the integer **N**, the number of people going to the party.
- **N** lines, one for each person, each containing three space-separated numbers "A B C", indicating the minimum fraction of each juice that would like in the drink. A, B and C are integers between 0 and 10000 inclusive, indicating the fraction in parts-per-ten-thousand. A + B + C ≤ 10000.

Output

- **T** lines, one for each test case in the order they occur in the input file, each containing the string "Case #X: Y" where X is the number of the test case, starting from 1, and Y is the maximum number of people who will like your drink.

Limits

1 ≤ **T** ≤ 12

Small dataset

1 ≤ **N** ≤ 10

Large dataset

1 ≤ **N** ≤ 5000

Sample

Input

```
3
3
10000 0 0
0 10000 0
0 0 10000
3
5000 0 0
0 2000 0
0 0 4000
5
0 1250 0
3000 0 3000
1000 1000 1000
2000 1000 2000
1000 3000 2000
```

Output

```
Case #1: 1
Case #2: 2
Case #3: 5
```

In the first case, for each juice, we have one person that wants the drink to be made entirely out of that juice! Clearly we can only satisfy one of them.

In the second case, we can satisfy any two of the three preferences.

In the third case, all five people will like the drink if we make it using equal thirds of each juice.

All problem statements, input data and contest analyses are licensed under the Creative Commons Attribution License.

© 2008-2017 Google   Google Home - Terms and Conditions - Privacy Policies and Principles

Powered by

Google Cloud Platform

Submissions

Juice
3pt Not attempted **97/97 users** correct (100%)
10pt Not attempted **74/93 users** correct (80%)

Ping Pong Balls
4pt Not attempted **92/97 users** correct (95%)
11pt Not attempted **18/32 users** correct (56%)

Mine Layer
4pt Not attempted **85/88 users** correct (97%)
13pt Not attempted **14/33 users** correct (42%)

Bridge Builders
8pt Not attempted **69/73 users** correct (95%)
17pt Not attempted **20/31 users** correct (65%)

The Year of Code Jam
7pt Not attempted **47/68 users** correct (69%)
23pt Not attempted **6/9 users** correct (67%)

Top Scores

ACRush	89
Innovative.Cat	89
bmerry	87
pmnox	76
yuhch123	66
gawry	66
Eryx	60
mystic	60
ploh	60
blueblimp	59

Problem B. **Ping Pong Balls**

This contest is open for practice. You can try every problem as many times as you like, though we won't keep track of which problems you solve. Read the Quick-Start Guide to get started.

Small input 4 points	Solve B-small
Large input 11 points	Solve B-large

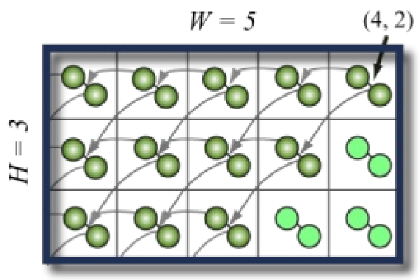
Problem

A large room is filled with mousetraps, arranged in a grid. Each mousetrap is loaded with two ping-pong balls, carefully placed so that when the mousetrap goes off they will be flung, land on other mousetraps and set them off. The walls of the room are sticky, so any balls that hit the walls of the room are effectively absorbed.

Every mousetrap that gets hit sends the two ping-pong balls in the same way: their movement is determined by a X and Y displacement relative to the launching mousetrap. You then decide to launch a single ping-pong ball into the room. It hits a mousetrap, setting it off, and launching its two balls. These two balls then set off two more mousetraps, and now four balls fly off... When the dust settles, many of the mousetraps have been set off, but some have been missed by all the flying balls.

You need to calculate how many mousetraps will be set off.

As an example (see the first sample test case), the picture below illustrates a room with width 5, height 3. The two directions for the ping-pong balls in each room are (-1, 0) and (-1, -1), respectively. The first ball you launch hits the mousetrap at the position (4, 2). In the end, 12 mousetraps are triggered.



Input

The first line of input gives the number of cases, **C**. **C** test cases follow. Each case contains four lines. The first line is the size of the grid of mousetraps (equal to the size of the room), given as its width **W** and height **H**. The next two lines give the destinations of the two ping-pong balls, as an X and Y displacement. For example, if the two lines were `0 1` and `1 1`, then triggering a mousetrap would launch two balls; one would hit the mousetrap just up from the triggered mousetrap, and the other would hit the mousetrap that is up and to the right of the triggered mousetrap. The final line has two integers specifying, respectively, the column and row of the mousetrap set off by the original ping-pong ball (where 0 0 would be the bottom left mousetrap).

Output

For each test case, output one line containing "Case #**A**: **B**", where **A** is 1-based number of the case and **B** is the number of mousetraps that are triggered (including the first one).

Limits

1 ≤ **C** ≤ 100
-20 ≤ any displacement ≤ 20
Neither vector will have zero length.

Small dataset

2 ≤ **W**, **H** ≤ 100

Large dataset

2 ≤ **W**, **H** ≤ 1000000

Sample

Input	Output
3 5 3 -1 0 -1 -1 4 2 50 50 0 1 1 1 10 10 6 2 2 0 3 0 0 0	Case #1: 12 Case #2: 820 Case #3: 5

All problem statements, input data and contest analyses are licensed under the Creative Commons Attribution License.

© 2008-2017 Google   Google Home - Terms and Conditions - Privacy Policies and Principles

Powered by

Google Cloud Platform

Submissions

Juice

3pt	Not attempted **97/97 users** correct (100%)
10pt	Not attempted **74/93 users** correct (80%)

Ping Pong Balls

4pt	Not attempted **92/97 users** correct (95%)
11pt	Not attempted **18/32 users** correct (56%)

Mine Layer

4pt	Not attempted **85/88 users** correct (97%)
13pt	Not attempted **14/33 users** correct (42%)

Bridge Builders

8pt	Not attempted **69/73 users** correct (95%)
17pt	Not attempted **20/31 users** correct (65%)

The Year of Code Jam

7pt	Not attempted **47/68 users** correct (69%)
23pt	Not attempted **6/9 users** correct (67%)

Top Scores

ACRush	89
Innovative.Cat	89
bmerry	87
pmnox	76
yuhch123	66
gawry	66
Eryx	60
mystic	60
ploh	60
blueblimp	59

Problem C. Mine Layer

This contest is open for practice. You can try every problem as many times as you like, though we won't keep track of which problems you solve. Read the Quick-Start Guide to get started.

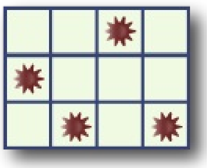
Small input 4 points	Solve C-small
Large input 13 points	Solve C-large

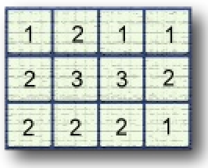
Problem

MineLayer is a MineSweeper-like puzzle game played on an **R** by **C** grid. Each square in the grid either has one mine or no mines at all. A MineLayer puzzle consists of a grid of numbers, each of which indicates the total number of mines in all adjacent squares and in the square underneath. The numbers will thus range from zero to nine.

The objective of MineLayer is to figure out a layout of the mines in the grid that matches the given clues.

Below is a typical 3 by 4 grid. The original layout is on the left, and the puzzle on the right.





Since there may be many solutions, your task is to write a program that outputs the maximum possible number of mines in the middle row. The number of rows will always be odd, and there will always be at least one solution to the puzzle.

Input

The first line of input gives the number of cases, **N**. **N** test cases follow.

The first line of each case contains two space-separated numbers: **R**, the number of rows, and **C**, the number of columns. **R** is always an odd integer. Each of the next **R** lines contains **C** space-separated numbers that denote the clues of that row.

Output

For each test case, output one line containing "Case #X: Y", where X is the 1-based case number, and Y is the maximum possible number of mines in the middle row of a grid that satisfies the given constraints.

Limits

1 ≤ **N** ≤ 50.
Each puzzle is guaranteed to have at least one solution.

Small dataset

R = 3 or **R** = 5.
3 ≤ **C** ≤ 5.

Large dataset

R is an odd number between 3 and 49, inclusive.
3 ≤ **C** ≤ 49.

Sample

Input	Output
2 3 3 2 2 1 3 4 3 2 3 2 3 4	Case #1: 1 Case #2: 1

```
1 2 1 1
2 3 3 2
2 2 2 1
```

All problem statements, input data and contest analyses are licensed under the Creative Commons Attribution License.

© 2008-2017 Google Google Home - Terms and Conditions - Privacy Policies and Principles

Powered by

Google Cloud Platform

World Finals 2008

A. Juice
B. Ping Pong Balls
C. Mine Layer
D. Bridge Builders
E. The Year of Code Jam

Contest Analysis
Questions asked

Submissions

Juice

3pt	Not attempted **97/97 users** correct (100%)
10pt	Not attempted **74/93 users** correct (80%)

Ping Pong Balls

4pt	Not attempted **92/97 users** correct (95%)
11pt	Not attempted **18/32 users** correct (56%)

Mine Layer

4pt	Not attempted **85/88 users** correct (97%)
13pt	Not attempted **14/33 users** correct (42%)

Bridge Builders

8pt	Not attempted **69/73 users** correct (95%)
17pt	Not attempted **20/31 users** correct (65%)

The Year of Code Jam

7pt	Not attempted **47/68 users** correct (69%)
23pt	Not attempted **6/9 users** correct (67%)

Top Scores

ACRush	89
Innovative.Cat	89
bmerry	87
pmnox	76
yuhch123	66
gawry	66
Eryx	60
mystic	60
ploh	60
blueblimp	59

Problem D. **Bridge Builders**

This contest is open for practice. You can try every problem as many times as you like, though we won't keep track of which problems you solve. Read the Quick-Start Guide to get started.

Small input 8 points	Solve D-small
Large input 17 points	Solve D-large

Problem

The king wants bridges built and he wants them built as quickly as possible. The king owns an **N** by **M** grid of land with each cell separated from its adjacent cells by a river running between them and he wants you to figure out how many man-hours of work it will take to build enough bridges to connect every island. Some cells are actually lakes and need not have a bridge built to them.

Some of the islands are forests where trees are abundant. Located in the top left corner is the *base camp*, which is always a forest.

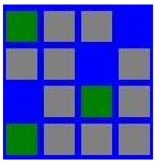
A bridge can only be built between two islands if they are vertically or horizontally adjacent, and one of the islands is accessible from the base camp through the bridges that are already built.

The number of man-hours it takes to build a bridge is the number of bridges the builders have to cross to get from the nearest forest to the island you're building to, including the bridge being built. Builders can only walk between two islands if there is already a bridge between them.

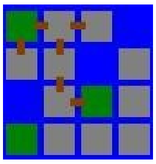
The king has already ensured that there is at least one way to connect all the islands.

Write a program that, given a map of the islands, will output the minimum number of man-hours required to connect all islands.

Consider this example. A green tile indicates a forest, gray indicates an empty island, and blue indicates water.

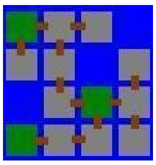


One optimal solution starts out by building the following bridges from the base camp forest.



This has a cost of 1 + 2 + 1 + 2 + 3 + 4 = 13

Now since the forest at row 3, column 3 is connected to base camp, we can build bridges from there. One optimal solution connects the rest of the islands with bridges built from this forest.



This has a cost of 2 + 1 + 2 + 1 + 2 + 3 = 11. This brings the total cost to 24 which is the optimal solution.

Input

The first line of the input contains an integer **T**, the number of test cases. **T** test cases follow. Each test case will begin with **N**, the number of rows, and **M**, the number of columns, on one line separated by a space. **N** rows follow that contain exactly **M** characters each. A 'T' indicates an island with a forest, a '#' indicates an island, and a '.' indicates water.

Output

A single line containing "Case #X: Y", where **X** is the 1-based case number, and **Y** is the minimum number of man-hours needed to connect all islands.

Limits

1 ≤ **T** ≤ 50
2 ≤ **N** ≤ 30
2 ≤ **M** ≤ 30
The top left cell will always be a 'T'
It will be possible to connect all islands through bridges

Small dataset

There will be at most 2 forests in the grid including the base camp.

Large dataset

There will be no limit on the number of forests in the grid.

Sample for small dataset

Input	Output

<pre>2
2 2
T.
T#
4 4
T##.
##.#
.#T#
####</pre>
	<pre>Case #1: 2
	Case #2: 24</pre>

Sample for large dataset

Input	Output

<pre>2
4 4
T##.
##.#
.#T#
T###
5 5
T#T.#
..#.#
#.###
###.#
T###T</pre>
	<pre>Case #1: 24
	Case #2: 49</pre>

All problem statements, input data and contest analyses are licensed under the Creative Commons Attribution License.

© 2008-2017 Google Google Home - Terms and Conditions - Privacy Policies and Principles

Powered by

Google Cloud Platform

World Finals 2008

Submissions

Juice

3pt	Not attempted **97/97 users** correct (100%)
10pt	Not attempted **74/93 users** correct (80%)

Ping Pong Balls

4pt	Not attempted **92/97 users** correct (95%)
11pt	Not attempted **18/32 users** correct (56%)

Mine Layer

4pt	Not attempted **85/88 users** correct (97%)
13pt	Not attempted **14/33 users** correct (42%)

Bridge Builders

8pt	Not attempted **69/73 users** correct (95%)
17pt	Not attempted **20/31 users** correct (65%)

The Year of Code Jam

7pt	Not attempted **47/68 users** correct (69%)
23pt	Not attempted **6/9 users** correct (67%)

Top Scores

ACRush	89
Innovative.Cat	89
bmerry	87
pmnox	76
yuhch123	66
gawry	66
Eryx	60
mystic	60
ploh	60
blueblimp	59

Problem E. The Year of Code Jam

This contest is open for practice. You can try every problem as many times as you like, though we won't keep track of which problems you solve. Read the Quick-Start Guide to get started.

Small input 7 points	Solve E-small
Large input 23 points	Solve E-large

Problem

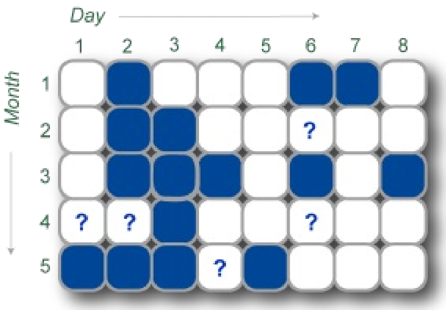
The year 2008 will be known as a year of change and transition, the start of a new era: we're talking, of course, about the new Google Code Jam format. The introduction of this contest has jammed so many great programming competitions together in a single year that people have started calling it *The Year of Code Jam*.
Sphinny, a passionate contestant, is looking at her calendar of the year and discovering that a great number of programming contests has been scheduled. She has marked every day of the year on the calendar in one of the three ways:

- White: She will not participate in a contest on this day. Either no contests are scheduled, or she has more important things to do (surely there are other good things in life!).
- Blue: She will definitely participate in a contest on this day.
- Question mark: There is a contest scheduled, but she has not decided yet whether she will participate.

Note: To simplify the problem, we'll assume that there is no concept of qualification: you don't have to participate in one contest to be eligible for another.

Being in a world that is somewhat different from ours, Sphinny's calendar has some features we must mention: It has **N** months, and each month has exactly **M** days.

The picture below depicts a calendar with 5 months, 8 days in each month, 15 blue days, and 5 question marks.



Looking at her beautiful calendar, Sphinny has decided that each day has up to 4 **neighbors** in the year: The previous day in the same month, the next day in the same month, the same day in the previous month, and the same day in the next month.

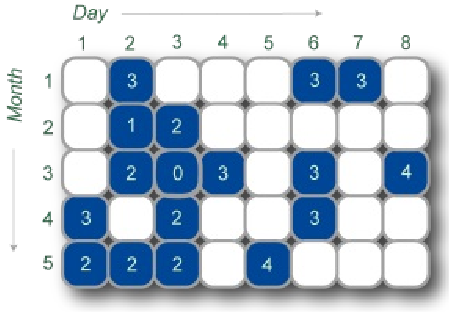
Sphinny wants to maximize her happiness from these contests, and she estimates the effect of the contests on her happiness as a summation of values for all the blue days. For each blue day, the value is computed as follows:

- The initial value is 4.
- For each blue neighbour the day has, decrease the value by 1.

You may think that Sphinny likes the contests, but participating on two consecutive days makes her a little tired. And for aesthetic reasons, participating on the same day in two consecutive months is also not so great.

Sphinny wants to plan her year now, and decide for every day with a question mark whether it should be white or blue. Her goal is simply to maximize the happiness value.

The following picture shows a solution for the example above. By changing two question marks to blue days, and the other three to white days, she can achieve a happiness value of 42.



Input

The first line in the input file contains the number of cases **T**. This is followed by **T** cases in the following format.
The first line is of the form "**N M**", where **N** and **M** are two numbers giving the number of months and the number of days per month.
The next **N** lines each contain a string of length **M**. The **j**-th character in the **i**-th string is one of {'#', '.', '?'}, which gives the status of the j-th day in the i-th month. '#' indicates a blue day, '.' indicates a white day, and '?' indicates a day with a question mark.

Output

For each input case, you should output a line in the format:

```
Case #X: Y
```

where **X** is the 1-based case number, and **Y** is the maximum happiness value.

Limits

1 ≤ **T** ≤ 100.

Small dataset

1 ≤ **M**, **N** ≤ 15.

Large dataset

1 ≤ **M**, **N** ≤ 50.

Sample

Input	Output

<pre>2
3 3
.?.
.?.
.#.
5 8
.#...##.
.##..?..
.###.#.#
??#..?..
###?#...</pre>
	<pre>Case #1: 8
	Case #2: 42</pre>

Note that the second sample is our example in the pictures above.

All problem statements, input data and contest analyses are licensed under the Creative Commons Attribution License.

© 2008-2017 Google Google Home - Terms and Conditions - Privacy Policies and Principles

Powered by

Google Cloud Platform