Palindromic Crossword (11pts, 17pts)

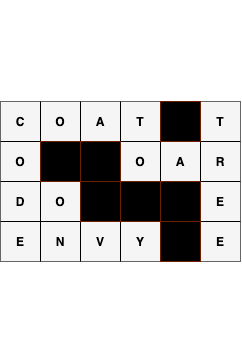
Competitive Submissions

You have not attempted this problem.

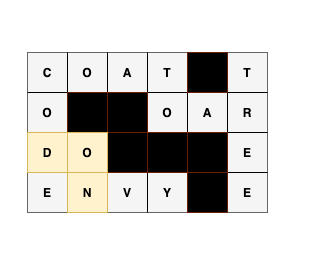
Last updated: Aug 22 2021, 09:00

Problem

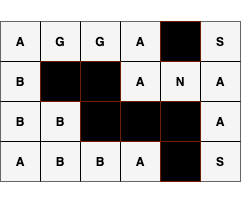
A [crossword puzzle](https://en.wikipedia.org/wiki/Crossword) is a rectangular grid of black cells and letters A-Z like the one shown below.



Words in the crossword are defined as maximal vertical or horizontal segments of characters. In the crossword below, DO and ON are examples of words.

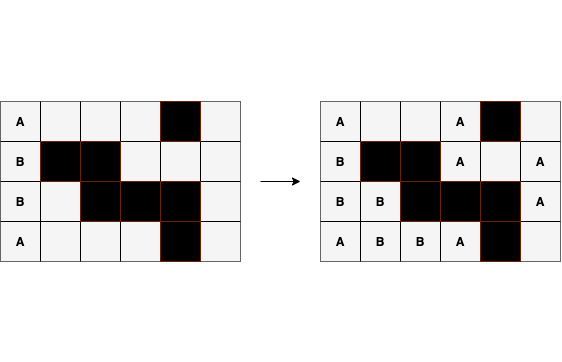


A palindromic crossword is one where every word is a [palindrome](https://en.wikipedia.org/wiki/Palindrome). Let Ri,jRi,j represent the character on the ii-th row and jj-th column, where ii and jj are 11-indexed. The top left corner is R1,1R1,1. In the example palindromic crossword below, the B in R3,2R3,2 is part of both the horizontal word starting at R3,1R3,1 and the vertical word ending at R4,2R4,2, and both are palindromes.



You have been gifted a palindromic crossword puzzle with NN rows and MM columns. You finished the crossword and throw away the clues, preparing to hang it on your wall. However, you accidentally erase some of the letters! You want to recover as much of the crossword as possible, but you do not have the clues anymore. Using only the knowledge that the crossword is palindromic, restore the maximum possible number of missing characters in the given crossword.

Missing letters are represented as empty white cells in the below diagram. The crossword on the left is the crossword you are given and the crossword on the right is the result after you recover as many letters as possible. The remaining cells cannot be filled because we do not have sufficient information to recover them.



Input

The first line of the input gives the number of test cases, TT. TT test cases follow.  
The first line of each test case contains two integers, NN and MM, representing the number of rows and columns in the crossword, respectively.  
The next NN lines represent the NN rows of the grid. The ii-th row consists of MM characters representing Ri,1Ri,1, Ri,2Ri,2, ……, Ri,MRi,M. Each character is one of the following:

* A capital letter of the alphabet (A-Z)
* A period (.) for a missing letter (empty white cell in the example crossword)
* A hash (#) for black cell

Output

For each test case, output one line containing Case #xx: yy where xx is the test case number (starting from 11) and yy is the number of empty white cells that were filled. Then, output NN more lines representing the final grid, with the missing characters (.) replaced by capital letters (A-Z) where possible.

Limits

Time limit: 60 seconds.  
Memory limit: 1 GB.  
1≤T≤1001≤T≤100.  
There exists at least one way to fill in the given input grid such that it is a palindromic crossword.  
All characters in the grid are in the set {{A-Z, #, .}}

Test Set 1

1≤N,M≤501≤N,M≤50.

Test Set 2

For at most 10 cases:  
1≤N,M≤10001≤N,M≤1000.  
  
For the remaining cases:  
1≤N,M≤501≤N,M≤50.

Sample

Sample Input

[save\_alt](https://codejam.googleapis.com/dashboard/get_file/AQj_6U3jG3prAiilZJkMuuERuVvtpKLRCGjyxv6W4bkdVTk_6XLdKlXA_BFJg3vapQyqz5uXTdBD3DtOltU3ZW0v6TnedSWEqzVVSC-xKA/palindromic_crossword_sample_ts1_input.txt)

content\_copy

2

2 2

A.

.#

4 6

A...#.

B##...

B.###.

A...#.

Sample Output

[save\_alt](https://codejam.googleapis.com/dashboard/get_file/AQj_6U02LhxsPxRrzdcORY8OMtyayoN6lXO6TE-vCMEYBAoG4wNVhLzQd9KoH5bHVyMVVtBldIzotdoVKKZhwvQrppj7RgcR0uI-ljKNu7Y/palindromic_crossword_sample_ts1_output.txt)

content\_copy

Case #1: 2

AA

A#

Case #2: 8

A..A#.

B##A.A

BB###A

ABBA#.

In Sample Case #2, we are able to fill in 88 of the blanks. We can fill in the missing letters as follows:

 row 11, column 44: We know this is A from character at row 11, column 11.

 row 22, column 4=4= A from row 11, column 44.

 row 22, column 6=6= A from row 22, column 44.

 row 33, column 6=6= A from row 22, column 66.

 row 33, column 2=2= B from row 33, column 11.

 row 44, column 2=2= B from row 33, column 22.

 row 44, column 3=3= B from row 44, column 22.

 row 44, column 4=4= A from row 44, column 11.

Increasing Sequence Card Game (9pts, 13pts, 18pts)

Competitive Submissions

You have not attempted this problem.

Last updated: Aug 22 2021, 09:00

Problem

You're playing a card game as a single player.

There are NN cards. The ii-th card has integer ii written on it.

You first shuffle NN cards randomly and put them in a pile. Take the card at the top of the pile to your hand. Then repeat the following process until the pile becomes empty:

1. Check the card on the top of the pile.
2. If the number on the card is larger than the number on the last card you took, take the card.
3. Otherwise, discard the card.

The score of the game is the number of cards in your hand at the end. With the given number of cards NN, what is the expected score of the game?

Input

The first line of the input contains the number of test cases, TT. TT lines follow. Each line contains a single integer NN, the number of cards in the pile.

Output

For each test case, output one line containing Case #x: y, where x is the test case number (starting from 1) and y is the expected score at the end of the game.

y will be considered correct if it is within an absolute or relative error of 10-6 of the correct answer. See the [FAQ](https://codingcompetitions.withgoogle.com/kickstart/faq#how-does-kick-start-handle-real-numbers) for an explanation of what that means, and what formats of real numbers we accept.

Limits

Time limit: 20 seconds.  
Memory limit: 1 GB.  
1≤T≤1001≤T≤100.

Test Set 1

1≤N≤101≤N≤10.

Test Set 2

1≤N≤1061≤N≤106.

Test Set 3

1≤N≤10181≤N≤1018.

Sample

Sample Input

[save\_alt](https://codejam.googleapis.com/dashboard/get_file/AQj_6U29o14TY3MB9Sicrc57sDtotCeFCjvTidotsLz0pB-tmWJX6bgpEwUC279MrMnMdcnbmsxLM3poSh9mFh4c-brQRZcogV8yevb6jxyvKjDcwOYH/increasing_sequence_card_game_sample_ts1_input.txt)

content\_copy

2

1

2

Sample Output

[save\_alt](https://codejam.googleapis.com/dashboard/get_file/AQj_6U0f3aNguXOh6nn6hniq6JJ2p7tK4gE2DGWrdXklwGuoWq5PD06-jg3ynhf6OkQa9Mr2IsUwey8OejwUEZfKn133sxNOGBfCkfJHoFg8RgbA16mGYg/increasing_sequence_card_game_sample_ts1_output.txt)

content\_copy

Case #1: 1.0

Case #2: 1.5

Birthday Cake (7pts, 13pts)

Competitive Submissions

You have not attempted this problem.

Last updated: Aug 22 2021, 09:00

Problem

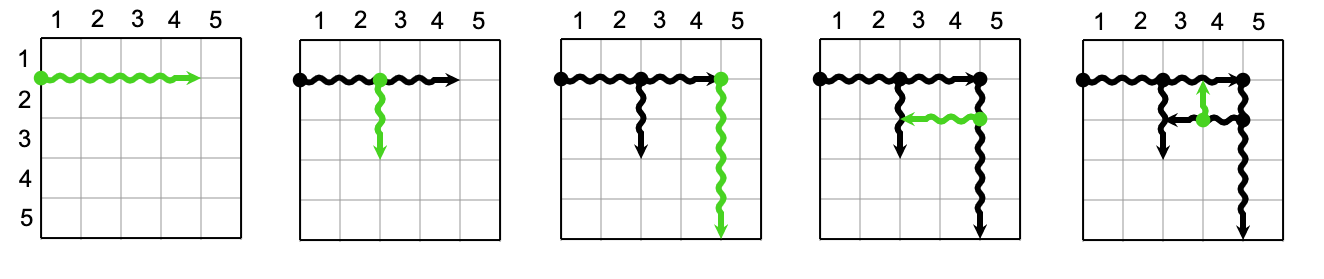
You are given a grid of RR rows and CC columns that corresponds to a birthday cake.  
The rows are numbered from 11 to RR starting from the top. The columns are numbered from 11 to CC starting from the left. Each cell in the grid is a square of size 1×11×1.  
You noticed that the most delicious part of the cake forms a single filled rectangle; that means all the cells inside this single rectangle will be delicious as well, but all the cells outside this rectangle are not delicious.  
You have a knife that is long enough to make straight-line cuts of length up to KK.

We want to make a series of cuts to extract each of the delicious cells separately, so that we can put candles on them, and enjoy the birthday party.  
To extract each of the delicious cells separately, they must be disconnected from any other cell.  
A cell is disconnected if no other cell is connected to it in any of the 44 directions (up, down, left, right).

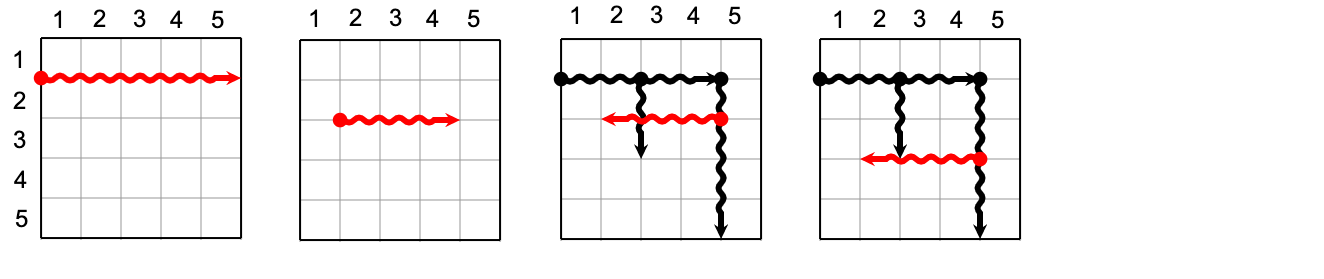
A cut is a directed line segment which is valid if the following conditions are met:

* The cut runs along one of the horizontal or vertical lines between the rows and columns of the grid.
* The length of the cut must not exceed KK.
* The starting and ending points of the cut must be grid points (i.e. a corner of a cell). In addition, the starting point must be already exposed, meaning that it lies on one of the 44 sides of the grid or on one of the previous cuts.
* The cut must not pass through any other exposed points. It may touch an exposed point, but if it does, it must end right there.

Suppose that K=4K=4. Below you can find five examples of valid cuts.



And here are four examples of invalid cuts



* In the first picture, the cut is too long (longer than 44).
* In the second picture, the cut starts from an unexposed point (neither one of the 44 sides of the grid nor a previous cut).
* In the third picture, the cut passes through an exposed point, it must stop once it touches the exposed point at length 22.
* The fourth picture is invalid because of the same reason as the third picture.

We need to find the minimum number of cuts needed to extract all the delicious cells.

Input

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

Each test case starts with a line containing three integers, RR, CC and KK.  
The next line contains four integers, r1r1, c1c1, r2r2, c2c2, representing the top-left and bottom-right cell of the delicious rectangle respectively.

Output

For each test case, output one line containing Case #xx: yy, where xx is the test case number (starting from 1) and yy is the minimum number of cuts.

Limits

Time limit: 10 seconds.  
Memory limit: 1 GB.  
1≤T≤1001≤T≤100.  
1≤r1≤r2≤R1≤r1≤r2≤R.  
1≤c1≤c2≤C1≤c1≤c2≤C.

Test Set 1

1≤R,C≤1001≤R,C≤100.  
K=1K=1.

Test Set 2

1≤R,C≤1051≤R,C≤105.  
1≤K≤1051≤K≤105.

Sample

*Note: there are additional samples that are not run on submissions down below.*

Sample Input

[save\_alt](https://codejam.googleapis.com/dashboard/get_file/AQj_6U0n4-b68c_ICXaC_1T2ykohQ6xmhjT9-kAg49BIA6iTCveMDbETxaVb4ynwqBizVTQqrJEx4JOb32rsYGqNeDpMupI/birthday_cake_sample_ts1_input.txt)

content\_copy

1

3 3 1

2 2 2 2

Sample Output

[save\_alt](https://codejam.googleapis.com/dashboard/get_file/AQj_6U2QToKU3nhHXawMP-3XohyG-vwjgP_14CfXxBokQx6xK2QAJ3rx1tf7luBv1k9ttatwzfobFE4NFw6OnPvNhTa3UOx-/birthday_cake_sample_ts1_output.txt)

content\_copy

Case #1: 5

Additional Sample - Test Set 2

*The following additional sample fits the limits of Test Set 2. It will not be run against your submitted solutions.*

Sample Input

[save\_alt](https://codejam.googleapis.com/dashboard/get_file/AQj_6U1cqNeLYvaSu21q5U2Eh_8Xgk6EyHlLUB5X2ovINc5HscIa2HUE6X07L-No0PPthLT-rjztK_4CPlN8Y20dzV33NBM/birthday_cake_sample_ts2_input.txt)

content\_copy

1

2 3 4

2 1 2 2

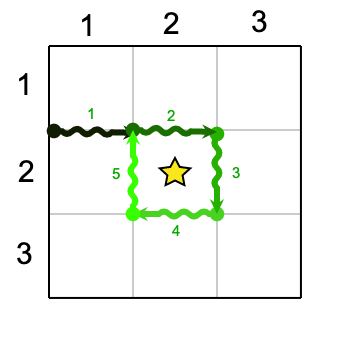
Sample Output

[save\_alt](https://codejam.googleapis.com/dashboard/get_file/AQj_6U0Z-OWLjmEpl4uZfO72yb-X06qscrFtoCjievHlh4O5h87Mjhy9NDMSZnzJWX7gNxcUvzTPmB8V_D7Ld4Yx9y_zf5QN/birthday_cake_sample_ts2_output.txt)

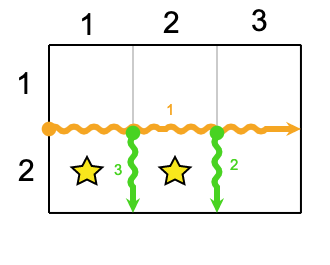
content\_copy

Case #1: 3

In the Sample Case, the minimum number of cuts is 55. One of the possible series of cuts is as follows:



In the Additional Sample Case, the minimum number of cuts is 33. One of the possible series of cuts is as follows:



Shuffled Anagrams (4pts, 8pts)

#### Competitive Submissions

You have not attempted this problem.

Last updated: Aug 22 2021, 09:00

### Problem

Let SS be a string containing only letters of the English alphabet. An anagram of SS is any string that contains exactly the same letters as SS (with the same number of occurrences for each letter), but in a different order. For example, the word kick has anagrams such as kcik and ckki.

Now, let S[i]S[i] be the ii-th letter in SS. We say that an anagram of SS, A, is *shuffled* if and only if for all ii, S[i]≠A[i]S[i]≠A[i]. So, for instance, kcik is not a shuffled anagram of kick as the first and fourth letters of both of them are the same. However, ckki would be considered a shuffled anagram of kick, as would ikkc.

Given an arbitrary string SS, your task is to output any one shuffled anagram of SS, or else print IMPOSSIBLE if this cannot be done.

### Input

The first line of the input gives the number of test cases, TT. TT test cases follow. Each test case consists of one line, a string of English letters.

### Output

For each test case, output one line containing Case #xx: yy, where xx is the test case number (starting from 1) and yy is a shuffled anagram of the string for that test case, or IMPOSSIBLE if no shuffled anagram exists for that string.

### Limits

Memory limit: 1 GB.  
1≤T≤1001≤T≤100.  
All input letters are lowercase English letters.

#### Test Set 1

Time limit: 20 seconds.  
1≤1≤ the length of SS ≤8≤8.

#### Test Set 2

Time limit: 40 seconds.  
1≤1≤ the length of SS ≤104≤104.

### Sample

Sample Input

[save\_alt](https://codejam.googleapis.com/dashboard/get_file/AQj_6U2V4occmnMb_iyeAI1F8Kjm0Fkb1ISo40L6xlIczX92zUZgkS7uYhiJDZxUEM4fqIdU8ECcfJgB9BFNDxhioopS9TLDNwb4/shuffled_anagrams_sample_ts1_input.txt)

content\_copy

2

start

jjj

Sample Output

[save\_alt](https://codejam.googleapis.com/dashboard/get_file/AQj_6U2BQB3Nhu--vfahjsEEPPp27W3E_o9-ft4zAn4lRE3ii1irQEngUJZKCbx-pcH_6q-BZUlNn8lwbNOkKRnVvwctblDZ3HJLkA/shuffled_anagrams_sample_ts1_output.txt)

content\_copy

Case #1: tarts

Case #2: IMPOSSIBLE

In test case #1, tarts is a shuffled anagram of start as none of the letters in each position of both strings match the other. Another possible solution is trsta (though you only need to provide one solution). However, in test case #2, there is no way of anagramming jjj to form a shuffled anagram, so IMPOSSIBLE is printed instead.