

<codeClash>2019

Code Clash Official Problem #0 'Introductions'

Problem statement

As with any social situation, it is important to properly introduce yourself before you get to any nitty gritty work. In life, you will likely have a tough time if you do not have great social skills. You will certainly have a tough time at Code Clash if you do not know how to properly process input and output, so hopefully you can get this easy point.

A single line will be provided as input containing a person's name. Address this person with response: "Hi, <name>! Nice to meet you!" Print this message exactly, substituting "<name>" for the string provided as input. There is one space between the sentences.

Sample test case

Sample input and output for this problem:

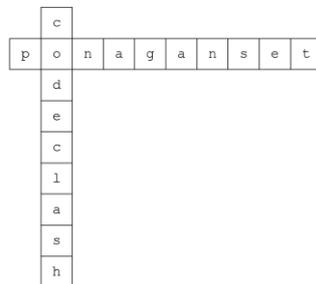
Input	Output
Carol	Hi, Carol! Nice to meet you!

<codeclash>2019

Code Clash Official Problem #1 'Characters'

Problem statement

To make crossword puzzles, it is necessary to find a common character between two words. Provided with two words from a crossword puzzle, your program should determine the first letter that the words have in common. Consider the longest of the two words provided to be the "primary word." The first letter from the primary word that is also contained by the secondary word is the letter that should be selected. If both words are the same length, the first word provided is the primary word. If neither of the words have any letters in common, print "NO COMMON CHARACTER" to your output.



The first line of input will contain an integer with the number of test cases to follow. For each test case, a subsequent line will follow with two words separated by a space. On each line of output, your program should print a single capital letter that represents the first letter that the words have in common, or print "NO COMMON CHARACTER".

Sample test case

Sample input and output for this problem:

Input	Output
4	
codeclash ponaganset	O
computer science	C
abc xyz	NO COMMON CHARACTER
apples laundry	L

<codeclash>2019

Code Clash Official Problem #2 'Sudoku'

Problem statement

The puzzle game Sudoku tasks the player with filling in each of the cells in a grid with a digit from 1 to 9 so that every row, column, and 3x3 square contains every digit exactly once. Sudoku is played on a 9x9 grid composed of nine smaller 3x3 sections. The same digit may not be repeated within these sections, nor may it be repeated within the same row or column. Your program must judge whether a provided Sudoku solution is valid or not.

4	3	1	6	7	9	5	2	8
9	6	7	2	5	8	3	4	1
5	8	2	1	4	3	9	6	7
6	5	9	8	1	7	2	3	4
3	2	8	5	6	4	1	7	9
7	1	4	9	3	2	8	5	6
8	7	3	4	2	1	6	9	5
1	4	5	3	9	6	7	8	2
2	9	6	7	8	5	4	1	3

A completed sudoku grid will be provided as input, with nine rows and nine columns that are populated with numerical digits. Your program will output "CORRECT" if the provided grid is validated as an accurate sudoku solution, or "INCORRECT" if the grid is invalid for any reason.

Sample test case

Sample input and output for this problem:

Input	Output
431679528 967258341 582143967 659817234 328564179 714932856 873421695 145396782 296785413	CORRECT

<codeClash>2019

Code Clash Official Problem #3 'Handshakes'

Problem statement

You are attending a large party with your family and there are many people at the party that you are meeting for the first time. Everyone at the party shakes hands with every other person that isn't in their family exactly once. Every family at the party has the same number of members. Given the size of each family and the number of families present, calculate the total number of handshakes that must have taken place.

The first line of input will contain an integer with the number of test cases to follow. For each test case, a subsequent line will follow with two integers: the first representing the size of each family and the second representing the number of families. All numbers will be non-negative and less than 2^{16} . Print each of your answers on their own line in the order they were given.

Sample test case

Sample input and output for this problem:

Input	Output
3	
2 4	24
4 10	720
9 15	8505

<codeClash>2019

Code Clash Official Problem #4 'Redacted'

Problem statement

[REDACTED]

The problem statement is hidden! You must analyze the sample test case provided below to figure out the problem and create a solution! The first line contains the number of test cases to follow.

Sample test case

Sample input and output for this problem:

Input	Output
3	
100	13
64	11
4	5

<codeClash>2019

Code Clash Official Problem #5 'Cheer Scrabble'

Problem statement

You are attending a meet for your favorite high school sport to support your friends that are competing. You decide to create signs that spell out your friends' names to cheer them on. Each sign will display one letter, and you will be able to combine several signs together to spell out any of your friends' names. To conserve signs, you want to make as few signs as possible while still being able to spell everyone's name.

The first line of input will contain an integer with the number of test cases to follow. For each test case, a subsequent line will follow with several names in all caps separated by spaces. Your answer output should contain each of the capitalized letters in alphabetical order, with each letter separated by a space.

Sample test case

Sample input and output for this problem:

Input	Output
3 NANCY AUSTIN TINA MIKE GEORGE MACIE ELIZABETH	A C I N N S T U Y E E G G I K M O R A B C E E H I L M T Z

<codeClash>2019

Code Clash Official Problem #6 'Mines'

Problem statement

You may be familiar with the classic desktop game called Minesweeper. Minesweeper is played on a large grid filled with mines and empty spaces. The empty cells of the grid adjacent to a mine have a number printed on them warning of the nearby danger. The number on the cell represents the number of adjacent mines (including diagonal mines). Mines are represented by an asterisk (*) and empty cells with no adjacent mines are represented by a hyphen (-). You are provided with a grid of mines and you must perform a transformation on this grid to include the numbers printed on adjacent cells.

The first line of input will contain two integers separated by a space. The first integer will be the number of columns and the second integer will be the number of rows in the playing board. The grid will follow, filled only with hyphens and asterisks. Your program should output a transformation of that same grid with a transformation. Each empty cell that is adjacent to a mine must be populated with a digit from 1-9 representing the number of adjacent mines. Other cells will remain unchanged.

Sample test case

Sample input and output for this problem:

Input	Output
30 12 -----*-----**-----*---* -----*-----*-----*-----*----- ---*-----*-----*----- -----*-----*-----*----- --*-----*---*-----*-----*---* -----*-----*-----*----- -*-----*-----*-----*-----*---* -----*-----*-----*----- -----*-----**-*-----*---* -----*-----**-*-----*----- -----*-----*-----*----- -----*-----*-----*-----*-----	-----1*1-111-1**1-111--12*11* --111-111-1*1-1221-1*1-12*2111 --1*1-----111-----11112*21111 -1221-111111-----111-1*2112*2 -1*1--1*11*1-----1*1-111-1*4* 1221--111222-----111----123*3 1*1-----1*1-----111---1*22* 111--111-111---12222*1---22222 -----1*211-----13**3*31---1*11* -----112*1-----1**66*3----11111 -----1221-----13****2-----11 -----1*1-----123321-----1*

<codeClash>2019

Code Clash Official Problem #7 'Artifacts'

Problem statement

A museum has a database of ancient documents and artifacts that they have copies of throughout the displays. Many of these documents and artifacts will have numbers written on them, many of them are written in Roman numerals. It is difficult for the computer to do processing on these Roman numerals without first converting them to standard base 10 numbers, so it is necessary to create a program that can convert to and from Roman numerals.

The first line of input will contain an integer with the number of test cases to follow. Each subsequent line will contain a number in either roman numeral format or a base 10 integer. Output your result as the opposite. All numbers will be non-negative and no greater than 3999.

Principles of Roman numerals: Write numerals left to right, with the largest numeral first. The largest numeral possible is used at each stage. No more than three instances of same adjacent numeral. A smaller numeral such as I or X placed before a larger one has the effect of subtraction - thus IV is one less than five, or four. Only one numeral can be placed to the left. The small numeral must be a power of ten: I, X or C. In Roman numerals, I = 1, V = 5, X = 10, L = 50, C = 100, D = 500, and M = 1000.

Sample test case

Sample input and output for this problem:

Input	Output
5	19
XIX	XXIX
29	147
CXLVII	MMCDLXXVI
2476	MX
1010	

<codeClash>2019

Code Clash Official Problem #8 'Search'

Problem statement

Word searches are puzzles in which the player is tasked with finding a list of words hidden within the grid of letters. Word searches are typically played on a rectangular board, but this word search will be played on a diamond-shaped board for added difficulty. In this diamond-shaped puzzle, words can be spelled forwards and backwards, horizontally and diagonally (but not vertically). Multiple words could share the same letter in the puzzle. Given a list of words and a diamond word search puzzle, transform the puzzle to reveal all the words hidden within.

The first line of input will contain two integers separated by a space. The first integer will be the number of search words and the second integer will be the total height and number of rows in the puzzle. The word search puzzle will follow. The first and last rows of the puzzle will only have one letter, and the number of letters per row increases as you approach the middle row. There will only be puzzles with an odd number of rows. A space separates letters that are on the same line, but there are no leading spaces to center/align the columns. Your program should output the solution to the puzzle, showing the search words in their original position and replacing all letters that are not part of a search word with hyphens (-).

Sample test case

Sample input and output for this problem:

Input	Output
3 9 CODE CLASH PHS A B H D S F G A I J K L S H P C Q R S O U V D X E	- - H - S - - A - - - L S H P C - - - O - - D - E

<codeClash>2019

Code Clash Official Problem #9 'Bacteria'

Problem statement

There are many types of unique bacteria that are found across the world. These bacteria reproduce over and die over time, so their populations are always fluctuating. Your mission is to estimate the population of several bacteria in an ecosystem after a given number of months. Every type of bacteria will have a starting population, a reproduction rate (each individual bacterium will duplicate every given number of months), and a lifespan (each individual bacterium will die after a given number of months). A bacterium does not give birth in the month that it dies. Assume that each bacterium can reproduce (duplicate) independently and the clock until it duplicates begins as soon as it is born. Assume that the entire population is at the beginning of its lifespan at the start of the population. For each simulation, calculate and print the final population of each bacteria species after a specific number of months.

The first line of input will contain two integers separated by a space. The first integer will be the number of bacteria to follow, and the second integer will be the number of months to run the simulation. A line will follow for each bacteria species, containing three integers separated by a space: the starting population, reproduction rate, and lifespan. All numbers will be non-negative and less than 2^{32} . Print each of your output numbers on its own line in the same order it was provided.

Sample test case

Sample input and output for this problem:

Input	Output
4 24	
10 1 5	78398880
50 2 10	149000
100 3 12	16200
500000 12 50	2000000

<codeclash>2019

Code Clash Official Problem #10 'Delivery'

Problem statement

A delivery truck's route changes every day depending on what packages need to be delivered. It is important for business that the delivery truck takes the most efficient route between its destination points so that no time is wasted. Your program must determine the shortest route to follow between each of the necessary stops. The truck will start at the warehouse, visit all the stops in the list of clients, and finally return to the warehouse. The warehouse is located at the origin, with coordinates 0, 0. The warehouse and all the stops are laid out as points on a coordinate plane. The truck must travel in a straight line from point to point. There are two solution paths with equal distance (one for each direction of travel, one going A-Z and another going Z-A). Print the solution path with the first stop that is closest to the warehouse.

Each line of input will contain an x and y coordinate separated by a comma and a space. Your program's output must be the list of the stops' coordinates formatted in the same fashion as the input, but arranged in an order that will allow the delivery truck to travel most efficiently. The last point will always be the warehouse at 0, 0. Do not print the warehouse coordinates as a part of your output. All numbers will be non-negative integers and less than 1000.

Sample test case

Sample input and output for this problem:

Input	Output
39, 40	39, 40
266, 366	48, 223
123, 14	266, 366
48, 223	356, 228
329, 10	329, 10
356, 228	123, 14
0, 0	