CMPS 11 - Assignment 2

Submission deadline: February 2nd, 2018 at 11:59 pm

Problem 1 - Series (25 points)

Euler's number or e is an irrational number approximately equal to 2.7182. In order to calculate e^w for $w \in \mathbb{R}$, we can use the following series representation:

$$e^w = 1 + w + \frac{w^2}{2!} + \frac{w^3}{3!} + \frac{w^4}{4!} + \ldots$$

Write a program that asks user to input two integers. The first integer is the power w, and the second is a (positive) number of terms, say i. Then calculate the approximation for e^w by evaluating the first i terms of the above series. Round the result to 2 decimal places.

Please note that although w can be any real number, your code only needs to work for $w \in \mathbb{Z}$ (integers). You can learn more about e here.

Example 1:

$$e^3 \approx 1 + 3 = 4$$

Input	Output
3	4.00
2	

Example 2:

$$e^7 \approx 1$$

Input	Output
7	1.00
1	

Example 3:

$$e^5 \approx 1 + 5 + \frac{5^2}{2\,!} + \frac{5^3}{3\,!} + \frac{5^4}{4\,!} = 65.38$$

Input	Output
5	65.38
5	

Example 4:

$$e^2 \approx 1 + 2 + \frac{2^2}{2\,!} = 5$$

Input	Output
2	5.00
3	

Example 5:

$$e^2 \approx 1 + 2 + \frac{2^2}{2\,!} + \frac{2^3}{3\,!} = 6.33$$

Input	Output
2	6.33
4	

Problem 2 - Matrix (40 points)

Write a program that receives “m” and “n” as the number of rows and columns of a matrix, respectively. Then it follows by “m” lines of input from the user, each referring to “n” **integers** in each row of the matrix.

The program then continuously reads commands from user and performs different operations based on user’s command. User’s command will be one of the following:

- **T**: Transpose - rows become columns (and vice versa) and prints the result matrix.
- **R**: RowMultiply - multiplies all values in each of the rows and prints the results in one line, separated by space.
- **C**: ColumnMin - finds the smallest value in each of the columns and prints the results in one line, separated by space.
- **Q**: Quit - stops reading commands from the user.

If user enters one of the first three commands, the operation should be applied to the original matrix. The result of each operation is either printing a new matrix (if the command is “T”) or printing the result of calculating RowMultiply or ColumnMin in one line (if the command is “R” or “C”).

Please note that each of the “T”, “R”, and “C” commands should be applied to the original matrix (not on the matrix that results from the most recent operation).
If user selects the quit option by entering “Q”, it terminates the program execution.

It will be a good practice if you implement “R”, “C”, and “T” operations using separate methods.

Example 1:

Input	Output
2 3	-126 56
-3 6 7	
1 8 7	
R	
Q	

Example 2:

Input	Output
2 3	126 56
3 6 7	1 6 7
1 8 7	
R	
C	
Q	

Example 3:

Input	Output
1 5	0
0 1 8 10 22	1
T	8
R	10
C	22
Q	0
	0 1 8 10 22

Example 4:

Input	Output
3 3	1 4 7
1 2 3	2 5 8
4 5 6	3 6 9
7 8 9	1 2 3
T	
C	
Q	

Problem 3 - Words (35 points)

We want to find out how many words in our list of words do not have any repeated characters. All words in our list are only made of characters ‘a’ to ‘z’, all in lowercase. There is no uppercase or special character in any of them.

- **Input:** An integer ’n’ indicating the number of words in our list. Then, it follows with ’n’ words/strings, each in one line.
- **Output:** An integer representing the number of words in our list without any repeated characters.

Hint: You can use an array of size 26 to keep track of the number of times each character appeared in a word.

Example 1:

Input	Output
3	1
hi	
hello	
apple	

Example 2:

Input	Output
1	0
all	

Example 3:

Input	Output
2	2
python	
bay	

Bonus Problem - ConnectFour (Up to 10 extra points)

John and Lise love Java so much that they decided to write a Connect Four game in Java. Their table has "m" rows and "n" columns. John and Lise alternate in throwing a marble in the table with specific color assigned to each one of them. In each step, the player chooses a column of the table and throws a marble into it. The marble goes down until it reaches the bottom of the table or reaches another marble, whichever comes first. The winner is the player who can fill 4 adjacent cells horizontally, vertically, or diagonally. If all cells of the top row become full and no one wins, it is a tie. Please help John and Lise with building this game.

Ladies first! Lise always starts the game.

- **Input:** First line of the input has "m"(number of rows) and "n"(number of columns). It follows with the columns that players choose in their turn.
- **Output:** Depending on the input, one of the options below should be printed:
  - Lise Wins!
  - John Wins!
  - It's a tie.

Example 1:

Input	Output
4 4	Lise Wins!
1	
0	
1	
1	
0	
2	
2	
3	
3	

Example 2:

Input	Output
3 4	It’s a tie.
0	
1	
1	
2	
2	
0	
2	
3	
0	
1	
3	
3	