Homework 3

Files to submit: **anagram.c, mat_add.c, pascal.c, plural.c, ReadMe.txt**

- All programs must compile without warnings when using the -Wall option
- If you are working in a group **ALL** members must submit the assignment on SmartSite
- Submit only the files requested
  - Do **NOT** submit folders or compressed files such as .zip, .rar, .tar, .targz, etc
- All output must match the provided solution in order to receive credit
  - We use a program to test your code so it must match exactly to receive credit
- All input will be valid unless stated otherwise
- The examples provided in the prompts do not represent all possible input you can receive. Please see the Tests folder for each problem for more adequate testing
- You may assume all inputs are valid unless otherwise specified
- All inputs in the examples in the prompt are underlined
- If you have questions please post them to Piazza

Restrictions

- No global variables are allowed
- Your main function may only declare variables and call other functions.

1. plural.c (20 mins) Write a program called plural.c that asks the user for a noun and prints out its plural form. We will be using the following rules for making a word plural

Ends With	Plural Form
ch, sh, s, x, or z	Add es
vowel and y	Add s
a consonant and y	Y becomes ies
f or fe	Becomes ves
All others	Add s

    1. Name your executable **plural.out**
    2. The user will only enter alphabetical characters
    3. The case of each character should be preserved
    4. No word will be longer than 10 characters
    5. You may only use the %s format specifier to display your string
    6. Hint: don't forget to restore the null character at the end of the string
    7. Examples
        1. Please enter a word: boy
           The plural from of boy is boys.
        2. Please enter a word: loaf
           The plural from of loaf is loaves.
        3. Please enter a word: moose
           The plural from of moose is mooses.
2. pascal.c (20 mins) Write a program called pascal.c that prints out Pascal's Triangle up to a user entered level.
    1. Name your executable **pascal.out**
    2. The maximum level that you will have to display is 28
    3. You do not have to make your output appear as a triangle
    4. You may find the following formula helpful $\sum_{i=0}^{N} i = \frac{N*(N+1)}{2}$
    5. Examples
        1. Please enter how many levels of Pascal's Triangle you would like to see: 2
           1
           1 1
        2. Please enter how many levels of Pascal's Triangle you would like to see: 4
           1
           1 1
           1 2 1
           1 3 3 1

3. (10 mins) Write a program called mat_add.c that asks the user for 2 matrices A, and B, and displays their sum, C.
    1. Name your executable **mat_add.out**
    2. All numbers entered will be integers
    3. All matrices will be valid
    4. The maximum dimension of each matrix is 100
    5. Each row of the matrix will be entered 1 line at a time
    6. The formula for calculating C[i][j] is $C[i][j] = A[i][j] + B[i][j]$
        1. For more on how to compute the sum of two matrices see here: http://www.purplemath.com/modules/mtrxadd.htm
    7. Examples:
        1. Please enter the number of rows: 2
           Please enter the number of columns: 2
           Enter Matrix A
           1 2
           3 4
           Enter Matrix B
           100 200
           200 400
           A + B =
           101 202
           203 404
        2. Please enter the number of rows: 2
           Please enter the number of columns: 3
           Enter Matrix A
           10 20 -30
           1 2 7
           Enter Matrix B
           1 2 30
           -3 4 5
           A + B =
           11 22 0
           -2 6 12

4. (15 mins) Write a program called anagram.c that asks the user for 2 words and tells the user if those 2 words are anagrams. A word is an anagram of another word if the letters in that word can be rearranged to form the other word. For example *Mary* and *army* are anagrams.
    1. Name your executable **anagram.out**
    2. The user will only enter alphabetical characters
    3. The maximum length of each word is 20 characters
    4. The check for an anagram should be case insensitive. For example MARY and army are still anagrams
    5. Examples
        1. Please enter the first word: MaRy
           Please enter the second word: arMY
           MaRy is an anagram of arMY
        2. Please enter the first word: dog
           Please enter the second word: god
           dog is an anagram of god
        3. Please enter the first word: bob
           Please enter the second word: bobs
           bob is NOT an anagram of bobs