Exercise 1:

In this exercise you will write a matrix manipulation library that provides 3 functions to handle square matrices of size d×d where d is a constant.

1. Launch the terminal
2. Create a new directory with the name "Lab04" inside "CSC215"
3. Write the C file "mat.c" that defines the functions:
    a. `fill_matrix`: takes a square matrix of size d×d and reads its elements from the keyboard
    b. `transpose`: takes two matrices t and s of size d×d and fills t with the transpose of s
    c. `print_matrix`: takes a matrix of size d×d and prints it to the screen in rows and columns
    d. BONUS: `multiply`: takes two matrices m1 and m2 of size d×d and prints their matrix product
4. You can always test your code against compilation errors using gcc. 3 points

Exercise 2:

To expose the library functions so they can be used by other programs you will write a header file for it.

1. Write the file "mat.h" that:
    a. declares the matrix size as a preprocessor constant.
    b. declares prototypes for all of mat.c functions
2. Make sure that your header file is protected against repetitive and circular inclusion. 1 point

Exercise 3:

1. Write the program "test.c" that uses that library mat.c to:
    a. read a matrix and print it to the screen
    b. transpose the matrix and print the result to the screen.
2. Compile and run the program. 1 point

Exercise 4:

Write a suitable makefile for your program. 2 points

Lab assignment: 3 points

Write a C program, assignment.c, that calculates the average of a group of 4 test scores, where the lowest score in the group is dropped.
Your program should use the following functions:

- Function `Avg` that calculates and displays the average of the three highest scores.
  This function should be called just once by `main`, and should be passed the four scores.
- Function `findLowest` that finds and returns the lowest of the four scores passed to it.
  It should be called by `Avg`, which uses the function to determine which of the four scores to drop.