CS222 Assignment 6 - Structure and makefile

Objective:

The purpose of this assignment is to learn and use structure type and makefile. You will use Figure 10.10 in the PPT notes as a template for implemetation.

Description:

You are to implement a program that will demonstrate complex number operations as in Figure 10.10. The program is divided into several files: "complex.h", "complex.c", and "main.c". You should have a Makefile that will allow the user to type "make" to generate the executable "complex".

Instructions:

Here are some basic components for your program design.

First, in the Makefile, you should compile "complex.c" and "main.c" into object files separately, and make sure the object file depends on "complex.h" as well. In other words, if you modify "complex.h", both "complex.c" and "main.c" will be compiled again if you type "make". The final executable "complex" depends on linking "main.o" and "complex.o".

Second, in general, global variables and constants,structure definitions, and function prototypes are defined in the header file for inclusions. In "complex.h", you should have structure type definition for "complex_t" and function prototypes for "add_complex", "subtract_complex", "multiply_complex", "abs_complex", and "print_complex". We are not implementing division.

Third, in general, related functions are grouped/implemented in the same file. In "complex.c", implement all the functions related to complex number operations as defined in the prototypes. Remember, in addition to the system include files, you need to add #include "complex.h" in order to have the defined structure and function prototypes.

Finally, in the main program, in addition to Fig 10.10's implementation, you also display the multiplication of two complex numbers.

As you have previously done, create a typescript that prints your program to the screen, confirms the system you are on, runs it using various input, and submit *both* the typescript and the source file containing your programs to Blackboard as HW 6.

Submission:

You will submit five files: (1) complex.h, complex.c, main.c, and Makefile; (2) a script file containing a sample run.