The first thing your program will do is print a menu of choices for the user. You may choose your own version of the wording or order of choices presented, but each choice given in the menu must match the following:

Menu Choice	**Valid User Input Choices**
Enter/Change Character	'C' or 'c'
Enter/Change Number	'N' or 'n'
Print Rectangle Type 1 (Border Only)	'1'
Print Rectangle Type 2 (Filled in)	'2'
Quit Program	'Q' or 'q'

A prompt is presented to the user to enter a choice from the menu. If the user enters a choice that is not a valid input, a message stating the choice is invalid is displayed and the menu is displayed again.

Your program must have at least five functions (not including main()) including:

- A function that prints the menu of choices for the user, prompts the user to enter a choice, and retrieves that choice. The return value of this function will be void. It will have one pass-by-reference parameter of type char. On the function's return, the parameter will contain the user's menu choice.

- A function that prompts the user to enter a single character. The return value of the function be a char and will return the character value entered by the user. This return value will be stored in a local variable, C, in main(). The initial default value of this character will be ' ' (blank or space character).

- A function that prompts the user to enter a numerical value between 1 and 15 (inclusive). If the user enters a value outside this range, the user is prompted to re-enter a value until a proper value is entered. The return value of the function be an int and will return the value entered by the user. This return value will be stored in a local variable, N, in main(). The initial default value of this character will be 0.

- Two "Print Rectangle" functions. Each function will take the current integer value N and character value C as input parameters. The return values of these functions will be void. The functions will print rectangles of N lines and columns using the input character C. The Border Only function will print the rectangle with the just the border. The Filled In function will print the rectangle as a solid rectangle. For example, if the integer value N = 6, and the character value C = '*' and the Filled In type is called, the following rectangle will be printed:

```
******
******
******
******
******
******
```

If the Border Only rectangle is to be printed, then the following rectangle is printed:

```
******
*    *
*    *
*    *
*    *
******
```

Your program must conform to the CS 262 Programming Style Guide. All variables must be declared within the body of a function, and if necessary, passed as parameters to other functions.

Suggested Steps to Complete the Assignment:

You are not required to use the following steps to write your program or even pay attention to them. However, following the steps will likely help you complete the lab faster and with fewer debugging issues. If you do use the suggested steps, you should test your program thoroughly to ensure each step works correctly before moving on to the next step.

1. Create a directory named "Lab3_<username>_<labsection> and make it your current working directory.

1. Copy your Makefile from Lab2 into this directory and modify it to work with the Lab3 executable
2. Create a source file with only your main() function and the standard #include files. Compile and run (it won't do anything).

1. Write a menu() function, called from main(), that prints the menu only (do not add any additional functionality or parameters at this point). Compile and test it to ensure it works properly.

1. Add code to the menu() function to prompt for retrieve user input (allow any character). Compile and test your code.
2. Add a pass-by-reference parameter to your menu() function that will retrieve the character input within that function and make it available for use within the main() function. This function must remain a void function and not return a value.

1. Enclose the menu() function within a loop that will exit the program when the Quit program choice is entered. Compile and test the logic of your code to ensure that the loop only exits on proper input.

1. Create four "stub functions" for the four other (non-Quit) choices. Put a print statement such as "This is function EnterChar()" or some other informative statement in the body of the function. For functions that return a value, return a specific character 'X' or number. This will be changed when the function is filled in.

1. Within the loop in main(), create the logic to call each of the four stub functions based on input from the menu choice (and handle incorrect input choices). Test this logic by observing the output of the stub function statements.

1. Fill in the logic and code for each function. Note that the Filled In Printing function is probably a little easier (logically) to write than the Border Only Printing function, so you may want to write it first. Once you have the Filled In function complete, think about how you would need to change it to print only the borders. This is the part of the lab (and the course) where you develop your skills to create algorithms that solve specific problems. These kinds of skills are not specific to C or any other language, but the methods used to implement these algorithms *are* language specific.

1. Test your program thoroughly.