CS 118 — Programming Fundamentals
Assignment #2

Due Date: Wednesday, September 11th at 11:55pm on Google Classroom.

Instructions: Assignments are to be done individually. **No late assignments will be accepted.** You must complete this assignment by yourself. You cannot work with anyone else in the class or with someone outside of the class. The code your write must be your own. You are encouraged to get help from the instructional staff. You may post general questions on Piazza. Do not post more than one line of code when using Piazza.

You must **submit a single zip file** containing your code and documentation on Google Classroom named ⟨*your_student_id*⟩*.zip* where ⟨*your_student_id*⟩ is something like *i19-XXXX*. This means that you must submit only **one file named *i19-XXXX.zip* containing only your source files**. Each file that you submit **must contain your name, student-id, and assignment#** on top of the file in comments. You submission must NOT contain multiple main() functions, otherwise it will not compile for grading. Test your program on a lab machine before submission.

Follow the instructions. Assignments not following the instructions will be awarded zero points.

Assignment Statement: ASCII Art is a graphics design technique that uses keyboard characters to create drawings and images. In this assignment you will create a program that produces an ASCII art representation of the FAST Main Building. The output will vary based on two inputs that the user will provide; *window style* and *grass style*. By simply changing the input characters the output will change (somewhat) dramatically. In the end your program must match the outputs shown for the give characters for *window* and *grass* style EXACTLY.

Here is the program output with the with an input character `U` for *window style* and `,` for *grass style*, (also provided as output1.txt).

```
FAST Main Building in ASCII Art
Enter window and grass style: U ,
                          !
                        (***)
                       |0|0|0|
   /\_________________---------_________________/\
  | || U U U U || U || U U U  || U || U U U U || |
  ==================== ______ ====================
  | || U U U U || U || |    | || U || U U U U || |
  | || U U U U || U || |    | || U || U U U U || |
  | || U U U U || U ||_|____|_|| U || U U U U || |
 /                   /   __   \                   \
/    , , , ,        /   /__\   \        , , , ,    \
```

Your program should take in both user inputs using a single `scanf()`. The inputs cannot be empty or newlines. Our assumption is that they will be valid ASCII characters.

When the input characters are changed to `@` and `^`, the following output is produced:

```
FAST Main Building in ASCII Art
Enter window and grass style: @ ^
                          !
                        (***)
                       |0|0|0|
   /\_________________---------_________________/\
  | || @ @ @ @ || @ || @ @ @  || @ || @ @ @ @ || |
  ==================== ______ ====================
  | || @ @ @ @ || @ || |    | || @ || @ @ @ @ || |
  | || @ @ @ @ || @ || |    | || @ || @ @ @ @ || |
  | || @ @ @ @ || @ ||_|____|_|| @ || @ @ @ @ || |
 /                    /   __   \                    \
/    ^ ^ ^ ^        /   /__\   \        ^ ^ ^ ^    \
```

This assignment is meant to give you practice with input and output functions from chapters 2 and 3 of the textbook, (`printf` and `scanf` statements, and program constants). This will require you to write printf() and scanf() statements that uses constant and local variables. You may not use anything besides the constructs from chapters 2 and 3. You may not use any libraries or functions other than `printf()` and `scanf()` statements.

You should use methods to structure your solution. Try not to write your complete solution in `main()`. You can pass a parameter into a function using the round brackets after the function name. For example, if you have a function called *printLastLine*, you can pass in the style of grass into the function by declaring it as

```
void printLastLine(char grass) {
 ...
}
```

where `grass` can be used inside the function as a variable. The function call in main would look like

```
printLastLine(gs);
```

where `gs` would be a variable declared in `main`.

You are required to properly indent your source code and will lose points if your indenting is not readable and consistent. You should localize variables whenever possible.

Note that the window and grass styles will determine how the drawing is rendered. Also, note how the *window style* is used to determine the shape of the windows in the floors of the main building. The inputs to your program will always be two valid ASCII characters.

As a reference my solution consists of approximately 50 lines including blank lines and comments. The problem is broken up into 3 - 5 functions including `main()`. Include a comment for each function you write explaining the purpose of that method.

On any given execution your program will produce just one version of the building, but it should be possible to change the styles of windows and grass during different runs. Given are two output files for inputs `U ,` and `@ ^`. Your output must match these exactly for a given input or you will lose points for correctness. Use a `diff` tool to ensure your program produces the correct output.

Honor Policy

This assignment is a individual learning opportunity that will be evaluated based on your ability to think independently, work through a problem in a logical manner solve the problems on your own. You may however discuss verbally or via email the general nature of the conceptual problem to be solved with your classmates or the course instructor, but you are to complete the actual assignment without resorting to help from any other person or other resources that are not authorized as part of this course. If in doubt, ask the course instructor. You may not use the Internet to search for solutions to the problem.