

# Programming Assignment 1:

## Custom Mosaic Tile

Due Thursday, September 7th at 11:59 pm

### 1. Overview

The purpose of this assignment is to give you experience with the process of editing, compiling and executing a simple C program, to give you experience with declaring variables of several kinds, and to exercise your ability to use the **scanf** and **printf** functions for interactive input and output.

For this assignment, you will write a program that displays a brief description of its purpose, prompts the user for some input, and then confirms the values received. As the course moves along, we'll be doing some work with colors and images, so this program collects info about a color. A sample run of the program can be seen below. The highlighted portions are the user input.

```
etkraem@joey12:~/p1
```

```
Tiger Town Custom Mosaic Tile
*****
```

```
Please enter the specifications for your custom tile
single word color name: azure
single character code for the color: A
```

```
Specify the color using RGB values (0-255)
red component: 0
green component: 255
blue component: 255
```

```
Confirming:
color name: azure
color code: A
RGB: (0,255,255)
```

You'll need to create integer variables for red, green and blue values, a character for the color code, and a character array (a string) for the color name. You can assume that the longest possible color name is 19 characters -- so make your array of size 20 (*explanation to follow later in the course*).

You will create this program using "your favorite editor" and will compile it using the Makefile below. You can create the Makefile using a text editor just as you do to create your program. Type "make" at the command line and this makefile will run the command on the second line (gcc -o p1 p1.c), if you've made any changes to the file named on the first line. The yellow boxes indicate a tab. Makefiles are sensitive to the inclusion of tabs, so be sure to enter tabs rather than spaces.

```
p1: p1.c
gcc-9 -o p1 p1.c
```

## 2. Suggestions for how to proceed:

1. Create a Proj1 subdirectory inside a Projects directory inside your Mod1 directory,
  - which should be in your MSCS\_Ready directory
    - which should be in your home directory
2. Copy your hello.c program from the Aug\_24 directory
  - Which should be in your Exercises directory
    - Which should be in your Mod 1 directory
      - Which should be in your MSCS\_Ready directory

to p1.c in the Proj1 directory.

If you can't find it there, then poke around in your directories from last week. If you still can't find it, then look in the Week1 subdirectory under files and use one of the files you find in the Files/Lecture\_materials/Week 1/Sample\_code folder .

3. Create the Makefile (or copy and edit the one from the class exercise).
4. Run “make” to be sure that the makefile is working and that you don't have any compiler errors in p1.c.
5. Change the printf statements to produce the first few lines of the program.
6. Add code to prompt and receive input for the color name and to display it back to the user.
7. Compile, test and make changes until things work as expected.
8. Add code to prompt and receive input for the color code and to display it back to the user.
9. Compile, test, etc.
10. Add code to prompt and receive input for the three integer color values (red, blue and green) and display them back to the user.
11. Compile, test, etc.
12. Edit and re-arrange the statements to match the example output seen above. ... compile, test, etc. ..

You will submit via a link on Canvas and we'll use GradeScope, which provides some automated grading of your submissions. This feature is new for me and I'm not able to see the student view, so we'll work through it together, with guidance from Professor Wooster. The assignment is structured so that some points are awarded by the autograder and others are awarded manually (thanks, Mathew!). This gives us the opportunity to maximize partial credit and also to assign points to aspects of your submission that don't show up in the output (use of meaningful variable names, selection of appropriate data types, etc.). See the rubric on the next page for details.

### 3. Grading rubric:

#### Manual grading:

___ / 02	p1.c and Makefile submitted
___ / 02	no extraneous files submitted
___/ 02	integer variables defined for red, green and blue values
___/ 02	character variable defined for color code
___/ 03	character array of appropriate size defined for color name
___/ 03	meaningful variable names
___/ 02	correct output of header text (Tiger Town ... )
___/ 02	correct solicitation/collection of color name and code
___/ 02	correct solicitation of RGB values
___ / 02	correctly formatted confirmation block
-----	
___ / 22	

Your output will be checked against four test cases, links for which appear on the Project 1 assignment page. Each test case is worth 2 points, with a project total of 30 points.

### 4. Notes on Collaboration

You are required to work individually on this assignment. You can ask questions and get help in piazza, in office hours (scheduled or by appointment) and in class on Wednesday.