EGR111 – Introduction to Computer Science (C Language)

RGB Converter Project Rev 1.0

Overview:

In this activity, you will create a color conversion solution. The user will enter an RGB color using a hexadecimal format. Your solution will produce alternative color format codes including RGB Arithmetic, RGB Percentage, RGB 8-bit, (C,M,Y,K), and a color name when appropriate.

Prerequisites:

Prior to beginning the instruction provided in this lesson you must have completed the following:

1. Variables and Data Types
2. Simple Input and Output
3. Basic Operators
4. Selection and iteration

Performance Outcomes:

1. Input hexadecimal values using scanf() function.
2. Validate user input using while() structure.
3. Convert and cast data values.
4. Use the ternary operator.
5. Implement switch expression.
6. Format output.
7. Use VS Code to create, edit, run, and debug a solution.

Resources:

1. RGB Color Code 🎨 Color Picker, Codes, Converters
2. RGB Color Codes Chart 🎨 (rapidtables.com)
3. RGB to CMYK conversion | color conversion (rapidtables.com)
4. C Ternary Operator (With Examples) (programiz.com)
5. Switch statement in c | Why do we need break statement in switch (log2base2.com)

Directions:

1. Research RGB color codes along with other color formats using resources listed above.

2. Create a color_converter.c solution that satisfies the following requirements.

    a. Includes comment lines at the top of the file describing solution and providing your name and date.

    b. Includes required header files.

c. Displays solution title to user and prompt them to enter an RGB color value using a pound or hash sign prefix as shown in the following sample.

```
Enter the RGB color code to convert: #FF0000
```

d. Uses the scanf() function to parse the user's input to r, g, and b values using hexadecimal format specifiers.

e. (optional) Validates the user input by looking for # character. Uses a *while* structure. Pseudocode is below. There will likely be issues. We'll discuss.

```
Prompt user for hex color input
Parse user input using scanf()
While user input not valid
     Prompt user for hex color input
     Parse user input using scanf()
```

f. Outputs the following alternative formats for the color specified. Example output is shown for an input of #FF0000. See additional information below.

```
RGB Value: 0xFF0000 (16711680)
RGB Arithmetic: (1.0, 0.0, 0.0)
RGB Percentage: (100%, 0%, 0%)
RGB 8-bit: (255, 0, 0)
(C,M,Y,K): (0,1,1,0)
Color Name: Red
```

g. RBG value is calculated from r, g, and b using the following. See resources for additional information.

```
RGB Value = (R*65536)+(G*256)+B
```

h. See resources for conversion to Cyan, Magenta, Yellow, Black (C,M,Y,K). Use ternary operator structure to determine required max value.

i. Use a switch structure to print the Color Name. You are switching on the RBG Value. Only the most basic RGB color names are required (Black, White, Red, Green, Blue, Cyan, Yellow, and Magenta). Print "Color Name: N/A" if the value is not a basic color.

3. Run your solution and test. Verify your results using the color tools included in the Web references.

4. Push your completed solution to your remote repository.