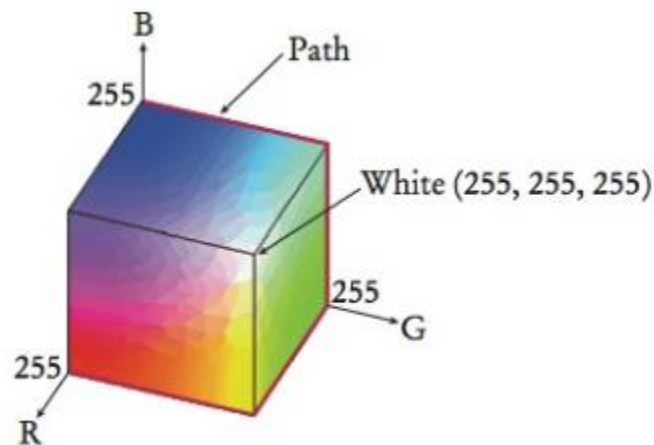


Homework 3 – Colors

GUI

Description

For this assignment we will emulate a smart thermostat. My smart thermostat will change color based on the temperature in the house. The color varies continuously from blue (when the temperature is low) to red (when the temperature is high). For this emulation we will consider a wide range of 0 to 100 degrees. Colors are coded in Red, Green, and Blue values called RGB in the format RGB(RedValue, GreenValue, BlueValue). Consider the image below.



The color can also be coded in hex (in the same order – red, green, blue) as #RRGGBB so for the color white – RGB(255, 255, 255) we can convert it to hex as #FFFFFF because 255 is hex FF. To make it a bit easier, we will only consider full saturated colors. That is, the color should be on the outside of the color cube, along the path that goes from blue through cyan, green, and yellow to red. In order to calculate the color, you will need to know how to interpolate between values. In general, if an output y should vary from c to d as an input x varies from a to b , then y is computed as follows:

$$z = (x - a) / (b - a)$$

$$y = d z + c (1 - z)$$

If the temperature is between 0 and 25 degrees, interpolate between blue and cyan, whose (red, green, blue) components are (0, 0, 255) and (0, 255, 255). For temperature values between 25 and 50, interpolate between (0, 255, 255) and (0, 255, 0), which represents green. For temperature values between 50 and 75, interpolate between (0, 255, 0) and (255, 255, 0) which represents yellow. Lastly, for temperatures greater than 75, interpolate between (255, 255, 0) and (255, 0, 0) which represents red.

Consider an example of 30 degrees. 30 is in the range of 25 to 50 so you have a low (0, 255, 255) and a high of (0 255, 0). Remember these colors are in the order (R, G, B). We must calculate our RGB. We start with Red. So, we interpolate. From the range above the Red low = 0 and the Red high = 0. Therefore, $z = (30 - 25) / (50 - 25) = 5/25 = .2$ and $y = (0 * .2 + 0 * (1 - .2)) = 0$. That means Red = 0. Now we move to Green. Green low = 255, Green high = 255. So, $z = (30 - 25) / (50 - 25) = .2$ and $y = (255 * .2 + 255 * (1 - .2)) = 51 + 204 = 255$. Therefore, green = 255. Lastly, Blue. Blue low = 255, Blue high = 0. $z = (30 - 25) / (50 - 25) = .2$ and $y = (0 * .2 + 255 * (1 - .2)) = 204$. So, our color is (0, 255, 204).

Now we calculate the color number using the formula - $65536 \times \text{red} + 256 \times \text{green} + \text{blue}$. Given our color of (0, 255, 204) = $0 + 255 * 256 + 204 = 65484$.

Now we calculate the hex. Remember hex is in the form of #RRGGBB so: Red – 0 in hex it is 00. Green - 255 in hex it is FF. Lastly, Blue – 204 in hex it is CC. Therefore, the hex code is #00FFCC

Specifications

You will need to complete the following:

1. Create a new Project called Colors.
2. **NOTE, DO NOT** forget to name all your controls correctly, including the form itself.
3. You need to show a GUI widget that shows the color. You also need a Rich Text Box to display the RGB, single integer value, and Hex of the color.
 - a. Consider an empty textbox of which the background color changes.
 - b. Use the method - Color.FromArgb(red, green, blue); to set the BackColor of a control.
4. You should allow the user to enter the temperature from 0 to 100 **AND** allow the user to use a numeric up down control to increment or decrement the temperature by 1 from 0 to 100.
 - a. The color and form should update each time the numeric up down is changed OR the user enters a new value in the textbox.
5. Make sure you validate the data.
6. Be sure to use appropriate helper methods to solve your task. Consider methods to interpolate values, create the integer color number, create the hex value, etc...
7. Make sure you do not forget to put all XML comments and a comment at the top of the code file which contains your name and the assignment.

Documentation

You will create a document (.docx, .rtf, .pdf) which contains the following:

- Your name and assignment.
- A screenshot of your form running with at least four test cases – one for each range.
- Your document should also contain a proof your calculations are correct. Work through each of your test cases by hand (like I did above) and show your code is working as expected.

- You will need to include your responses to the following questions – be detailed in your responses.
 - Explain the RGB color model and compare it to the CMYK color model.
 - What event did you use for the Numeric up down? Why?

What to Submit

You need to submit your entire solution folder (zipped) and your document. **DO NOT** zip your document. Make sure your document is in the correct format and all your files include your name and assignment.