

PHOTOSHOP ASCIIFY PROJECT

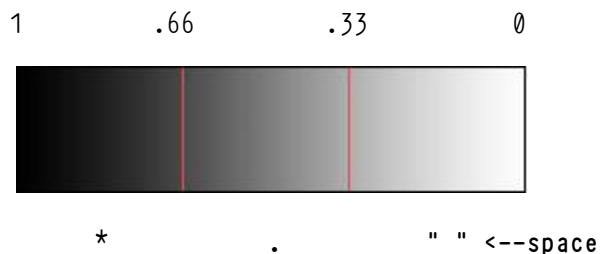
The Photoshop team leader Christianna Taylor wants to implement a new image filter called Asciify. The Asciify filter takes an image and renders it as ascii characters. Christianna wants a prototype developed in C# first. The general process involves.

1. Looking at every pixel in the image. Image files can be loaded in C# via the BitmapMaker class.
2. For each pixel convert the RGB color values to a normalized greyscale value (0 to 1). There are various ways to convert RGB to greyscale, find a method and implement in your program.
3. Used the normalized greyscale values to determine which ascii character to draw.

The process used to draw the correct ascii character involves mapping normalized ranges of the greyscale to certain characters. Normalized means all values fall between 0 and 1. So a normalized greyscale ramp looks like.



A mapping might look like this, notice how the darker areas of the ramp are assigned to characters that use more "ink".



Note: your filter should use six greyscale ranges to map to ascii characters.

The program should implement the below methods

`string Asciiitize(Bitmap)`

This method should accept a Bitmap and return a string containing the ascii text version of the picture

`double AverageColor(Color)`

This method should accept a Color and return a normalized value (0-1) of the grey value calculated from the Color's RGB values

`string GrayToString(double)`

This method should accept a normalized value (0-1) and return a string containing the ascii character mapped to the range the value falls in.

Before coding come up with a design for the GUI and for the process of converting the image to ASCII and get it approved. There is an example of what the program's output should look like on the next page.



