CPS109 Lab 5

Most of the questions in this lab come from Chapter 5 (or earlier chapters) of the course text, Introduction to Computing and Programming in Python, by Guzdial and Ericson. Please put your answers (numbered) in a document and submit it on D2L as a PDF file. Other formats are not accepted.

The **learning objectives** for Chapter 5 include:

- to implement controlled color changes, like redeye removal and sepia
- to use background subtraction and replacement
- to use chromakey for putting in a new background
- to draw borders
- to use conditionals with if, elif, else statements

To do:

1) Write a function **changeColor(picture, amount, rgbNumber)**, where amount is a number from the value -0.99 to 0.99, and rgbNumber is 1 (for red), 2 (for green), or 3 (for blue). If amount is negative, you decrease the specified color component by that fraction, and if amount is positive, you increase the specified color component by that fraction. For example,
- changeColor(picture, -0.10, 1) decreases red by 10%.
- changeColor(picture, 0.30, 2) increases green by 30%.
- changeColor(picture, 0, 3) makes no change.

2) Imagine you a picture to copy, but you only have 8 mixtures of colors to use. Write a function **posterize8(picture)**, which changes the color of each pixel as follows: for red, green and blue, if the component is less than 100, make it 0, and otherwise make it 255.
3) Write a function **border(picture, borderwidth, bordercolor)** which puts a border on all four sides of the picture with width = borderwidth and color = bordercolor. Assume the borderwidth is an integer less than 50.
4) Write a function **drawXequalsYline(picture, color)**, which draws the x = y line with the given color on the picture. Draw that line from (0,0) to (m, m) where m = min(W, H), where W, H are the width and height of the picture.
5) Write a function **drawQuadrants(picture, color)**, which puts a vertical line down the middle of the given picture and a horizontal line across the middle, where the two lines have the given color.
6) Write a function **drawDiagonalLine2(picture, color)**, which draws a line on the picture from the lower left corner (x, y) = (0, H-1) to the top right corner (x, y) = (W-1, 0), where W, H are the width and height of the picture, respectively. The equation for that line is y = H' - (H'/W')x, where H' = H - 1 and W' = W - 1.
7) Write a function **posterize3(picture)** using if, elif and else to posterize as follows, where r, g, b are the red, green and blue components of a pixel. If r > 180, then set the pizel to red. If not, then check if b > 180, and if so, set the pixel to blue. If not, then check if g > 180, and if so set the pixel to green. If none of these components is greater than 180, then set the pizel to black.
8) Write a function that **pinkify(picture)** that pinkifies the white in a picture. For each pixel, if the three components are all over 100, then set the color of the pixel to pink, otherwise no change.
9) Write a function **downupRed(picture)**, which decreases the red by 50% on the left half of the picture and increases red by 200% on the right side.
10) Write a function **thirds(picture)** which makes the top third brighter (using makeLighter); for the middle third it decreases the red and green by 30%; and it negates the bottom third (makes it a negative).