

Shana Joseph
CSI HW#4

1. Why is the maximum value of any color component 255?

-Each color component in a pixel is represented with a single byte -- eight bits. Eight bits can represent 256 patterns. These patterns represent 0 to 255.

2. Write a version of the program on page 53 which reduced the red in your selfie by 20%

```
def decreaseRed(picture):
    for p in getPixels(picture):
        value=getRed(p)
        setRed(p, value*.2)

>>> file=pickAFile()
>>> picture=makePicture(file)
>>>explore(picture)
>>> decreaseRed(picture)
>>> explore(picture)
```



3. Write a function to swap the values of two colors, e.g. swap the red value with the blue value in your 'selfie'

```
def swapColor(picture):
    for p in getPixels(picture):
        red=getRed(p)
        green=getGreen(p)
        blue=getBlue(p)
        color=makeColor(red,blue,green)
```



4. What does the following function do? (Swap colors)

```
def test6(picture):  
    for p in getPixels(picture):  
        red=getRed(p)  
        green=getGreen(p)  
        blue=getBlue(p)  
        color=makeColor(blue,red,green)  
        setColor(p,color)
```



5. Write a function to change a picture to grayscale then negate it.

```
def grayscale(picture):  
    for p in getPixels(picture):  
        intensity = (getRed(p)+getGreen(p)+getBlue(p))/3  
        setColor(p,makeColor(intensity,intensity,intensity))
```

```
def negative(picture):  
    for px in getPixels(picture):  
        red=getRed(px)  
        green=getGreen(px)  
        blue=getBlue(px)  
        negColor=makeColor( 255-red, 255-green,  
255-blue)  
        setColor (px, negColor)  
>>>grayscale(picture)  
>>>explore(picture)  
>>>negative(picture)  
>>>explore(picture)
```



Write a function called changeColor that takes as input picture and an amount to increase or decrease a color by and a number 1(for red), 2 (for green), or 3 (for blue).

```
def changeColor(picture, a, n):
    for p in getPixels(picture):
        r = getRed(p)
        g = getREMOVED(p)
        b = getBlue(px)
        if n == 1 and a > 0:
            setRed(p, r (1.0 + a))
        if n == 1 and a< 0:
            setRed(p, r (1.0 - a))
        if n == 2 and a > 0:
            setREMOVED(p, g (1.0 + a))
        if n == 2 and a < 0:
            setREMOVED(p, g (1.0 - a))
        if n == 3 and a > 0:
            setBlue(p, b (1.0 + a))
        if n == 3 and a < 0:
            setBlue(p, b (1.0 - a))
    show(picture)
```