def zero(img):

for px in getPixels(img):

col = makeColor(0, 0, 0)

setColor(px, col)

The result is the entire image is black.

def zero(img):

for px in getPixels(img):

col = makeColor(255, 255, 255)

setColor(px, col)

The result is the entire image is white.

1. 1: Decreases the R value of each pixel by half

2: Increases the B value of each pixel by 50%

3: Sets the G value of each pixel to 0

4: Increases the R, G and B values by 10

5: Decreases the value of the R, G, and B of each pixel by 20, unless the decrease puts the value below 0, in which case the value is set to 0.

6: Literally nothing

7: Reduces the value of each component by half for each pixel.

8: Reduces the value of each component by a third for each pixel.

9: Doubles the value of each component for each pixel.

1. The majority of the image is white-washed.

def blue(img):

for px in getPixels(img):

if getBlue(px) > 100:

b = getBlue(px)+50

if b < 255:

b = 255

col = makeColor(getRed(px), getGreen(px), b)

setColor(px, col)



def blueify(img):

for px in getPixels(img):

r = getRed(px) / 2

g = getGreen(px) /2

b = getBlue(px) \* 2

col = makeColor(r, g, b)

setColor(px, col)

Before and after:  

1. def negGrey(img):

#Greyscale

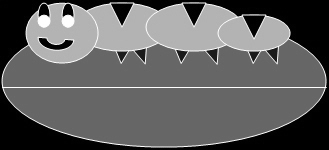
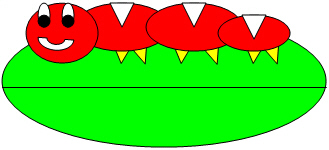
for px in getPixels(img):

intensity = getRed(px) \* 0.3 + getGreen(px) \* 0.6 + getBlue(px) \* 0.1

setColor(px, makeColor(intensity, intensity, intensity))

for px in getPixels(img):

setColor(px, makeColor(255-getRed(px), 255-getGreen(px), 255-getBlue(255)))

Before and after:

1. def negGrey(img):

greyscale(img)

negate(img)

def greyscale(img):

for px in getPixels(img):

intensity = getRed(px) \* 0.3 + getGreen(px) \* 0.6 + getBlue(px) \* 0.1

setColor(px, makeColor(intensity, intensity, intensity))

def negate(img):

for px in getPixels(img):

setColor(px, makeColor(255-getRed(px), 255-getGreen(px), 255-getBlue(255)))

The program in #7 is the most efficient, while the one in #8 is easier to understand.

9)

def lightGrey(img):

for px in getPixels(img):

r = getRed(px) + 75

g = getGreen(px) + 75

b = getBlue(px) + 75

intensity = r \* 0.3 + g \* 0.6 + b \* 0.1

setColor(px, makeColor(intensity, intensity, intensity))

10)

def lightGrey2(img):

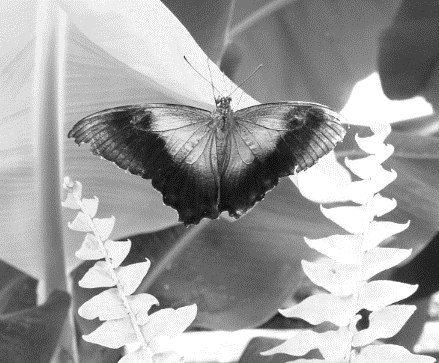
for px in getPixels(img):

setColor(px, makeLighter(makeColor(getRed(px), getGreen(px), getBlue(px))))

intensity = getRed(px) \* 0.3 + getGreen(px) \* 0.6 + getBlue(px) \* 0.1

setColor(px, makeColor(intensity, intensity, intensity))

Original, from question 9, from question 10





The makeLighter() function keeps the contrast level of the image the same, while simply adding 75 to each component does not.