Cason Zelencik

Completed



Original



```
from mediaComp import *
import random

# Cason Zelencik
# 10/22/2026

#Main Function used for some base setup things
def collage():
    setMediaFolder()
    baseImage = makePicture(getMediaFolder("skeleton.png"))
    signature = makePicture(getMediaFolder("Signature.png"))
    generateAndCombineImages(baseImage, signature)

#This functions calls several helper functions to generate and combine the images together
def generateAndCombineImages(baseImage, signature):
    fullCollage = makeEmptyPicture(getWidth(baseImage) * 2, getHeight(baseImage) * 2, white)
    combine(fullCollage, baseImage, getWidth(baseImage), getHeight(baseImage))
    mirroredPic = mirrorVertical(makePicture(getMediaFolder("skeleton.png")))
    combine(fullCollage, mirroredPic, 0, 0)
    random = rainbify(mirroredPic)
    combine(fullCollage, random, 0, getHeight(random))
    lightened = lighten(lighten(lighten(makePicture(getMediaFolder("skeleton.png")))))
    combine(fullCollage, lightened, getWidth(lightened), 0)
    posterPic = posterize(scale(makePicture(getMediaFolder("skeleton.png")), .5))
    combine(fullCollage, posterPic, getWidth(baseImage), getHeight(baseImage))
    bluePic = mirrorVertical(blueTint(scale(makePicture(getMediaFolder("skeleton.png")), .5)))
    combine(fullCollage, bluePic, getWidth(baseImage) - getWidth(bluePic),getHeight(baseImage)-getHeight(bluePic))
    redPic = redTint(scale(makePicture(getMediaFolder("skeleton.png")), .5))
    combine(fullCollage, redPic, getWidth(baseImage), getHeight(baseImage) - getHeight(redPic))
    purplePic = mirrorVertical(purpleTint(scale(makePicture(getMediaFolder("skeleton.png")), .5)))
    combine(fullCollage, purplePic, getWidth(baseImage) - getWidth(purplePic), getHeight(baseImage))
    chromakey(signature, fullCollage)
    show(fullCollage)
```

```
#The function for combining the images into one collage
def combine(collage, imageAdding, startX, startY):
    for x in range(0, getWidth(imageAdding)):
        for y in range(0, getHeight(imageAdding)):
            currentPixel = getPixelAt(imageAdding, x, y)
            currentColor = getColor(currentPixel)
            addPixel = getPixelAt(collage, startX + x, startY + y,)
            setColor(addPixel, currentColor)

#Function used to scale images either bigger or smaller
def scale(source_picture, number):
    new_width = getWidth(source_picture) * number
    new_height = getHeight(source_picture) * number
    new_picture = makeEmptyPicture(new_width, new_height)
    source_x = 0
    for new_x in range(0,getWidth(new_picture)):
        source_y = 0
        for new_y in range(0,getHeight(new_picture)):
            color = getColor(getPixelAt(source_picture, int(source_x), int(source_y)))
            setColor(getPixelAt(new_picture, new_x, new_y), color)
            source_y = source_y + (1 / number)
        source_x = source_x + (1 / number)
    return new_picture

#Function used to flip an image vertically
def mirrorVertical(mirrored):
    width = getWidth(mirrored)
    mirror_point = width // 2
    for x in range(0, mirror_point):
        for y in range(0, getHeight(mirrored)):
            left_pixel = getPixelAt(mirrored, x, y)
            right_pixel = getPixelAt(mirrored, width - x - 1, y)
            leftColor = getColor(left_pixel)
            rightColor = getColor(right_pixel)
            setColor(right_pixel, leftColor)
            setColor(left_pixel, rightColor)
    return mirrored
```

```
#Function to lighten and image
def lighten(lightened):
    for pixel in getPixels(lightened):
        for i in range(2):
            color = getColor(pixel)
            color = makeLighter(color)
            setColor(pixel, color)
    return lightened

#A chromakey function used for the signature
def chromakey(source, background):
    for source_pixel in getPixels(source):
        x = getX(source_pixel)
        y = getY(source_pixel)
        if (getRed(source_pixel) < 1 and getGreen(source_pixel) < 1 and getBlue(source_pixel) < 1):
            background_pixel = getPixelAt(background, x, y)
            source_color = getColor(source_pixel)
            setColor(background_pixel, source_color)

#Function to posterize and image
def posterize(picture):
    postered = picture
    for pixel in getPixels(picture):
        red_value = getRed(pixel)
        green_value = getGreen(pixel)
        blue_value = getBlue(pixel)
        luminance = (red_value + green_value + blue_value) // 3
        if luminance < 50:
            setColor(pixel, black)
        elif 49 < luminance < 151:
            setColor(pixel, darkGray)
        elif 150 < luminance < 201:
            setColor(pixel, lightGray)
        else:
            setColor(pixel, white)
    return postered

#Function to greyscale an image
def grayScale(picture):
    for pixel in getPixels(picture):
        intensity = (getRed(pixel) + getGreen(pixel) + getBlue(pixel)) / 3
        setColor(pixel, makeColor(intensity, intensity, intensity))
```

```
#Function to give an image a blue tint
def blueTint(picture):
    pic = picture
    grayScale(pic)
    for pixel in getPixels(pic):
        red_value = getRed(pixel)
        green_value = getGreen(pixel)
        blue_value = getBlue(pixel)
        if blue_value < 63:
            blue_value = blue_value * 3
        elif 63 < blue_value < 192:
            blue_value = blue_value * 2.5
        else:
            blue_value = blue_value * 2
        setBlue(pixel, blue_value)
        setRed(pixel, red_value * .75)
        setGreen(pixel, green_value * .75)
    return pic

#Function to give an image a red tint
def redTint(picture):
    pic = picture
    grayScale(pic)
    for pixel in getPixels(pic):
        red_value = getRed(pixel)
        green_value = getGreen(pixel)
        blue_value = getBlue(pixel)
        if red_value < 63:
            red_value = red_value * 3
        elif 63 < red_value < 192:
            red_value = red_value * 2.5
        else:
            red_value = red_value * 2
        setBlue(pixel, blue_value * .75)
        setRed(pixel, red_value)
        setGreen(pixel, green_value * .75)
    return pic

#Function to give an image a purple tint
def purpleTint(picture):
    pic = picture
    grayScale(pic)
    for pixel in getPixels(pic):
        red_value = getRed(pixel)
        green_value = getGreen(pixel)
```

```
        blue_value = getBlue(pixel)
        if green_value < 63:
            red_value = red_value * 1.3
            blue_value = blue_value * 3
        elif 63 < green_value < 192:
            red_value = red_value * 1.2
            blue_value = blue_value * 2.5
        else:
            red_value = red_value
            blue_value = blue_value * 2
        setBlue(pixel, blue_value)
        setRed(pixel, red_value)
        setGreen(pixel, green_value * .75)
    return pic

#Function to make an image static, different static textures every time
def rainbify(rainbowed):
    for pixel in getPixels(rainbowed):
        redModify = random.randint(0, 200)
        blueModify = random.randint(0, 200)
        greenModify = random.randint(0, 200)
        red_value = getRed(pixel)
        green_value = getGreen(pixel)
        blue_value = getBlue(pixel)
        setBlue(pixel, blue_value + blueModify)
        setRed(pixel, red_value + redModify)
        setGreen(pixel, green_value + greenModify)
    return rainbowed
```