Noah Scheick

Completed



Original



```
#--------------------
# Project 2
# Noah Scheick
# October 20, 2026
#--------------------

from mediaComp import *
setMediaFolder()

def collage():
    canvas = makeEmptyPicture(1000, 736)
    source_img_path = getMediaFolder("j love heart.jpg")
    sig_path = getMediaFolder("signature.jpg")
    img = makePicture(source_img_path)
    sig_img = makePicture(sig_path)
    panel1 = resizeImage(img, 500, 368)
    panel1 = applyNegative(panel1)
    copyInto(panel1, canvas, 0, 0)
    panel2 = resizeImage(img, 500, 368)
    panel2 = posterizeAndShift(panel2)
    copyInto(panel2, canvas, 500, 0)
    panel3 = resizeImage(img, 500, 368)
    panel3 = applyGrayscaleAndLighten(panel3)
    copyInto(panel3, canvas, 0, 368)
    panel4 = resizeImage(img, 500, 368)
    panel4 = applySepiaTone(panel4)
    copyInto(panel4, canvas, 500, 368)
    addRectFilled(canvas, 344, 252, 312, 232, makeColor(255, 255, 255))
    centerImg = resizeImage(img, 300, 220)
    copyInto(centerImg, canvas, 350, 258)
    applyRotatedSignature(canvas, sig_img, 830, 620)
    pictureTool(canvas)
    return canvas


def resizeImage(picture, targetWidth, targetHeight):
    target = makeEmptyPicture(targetWidth, targetHeight)
    srcWidth = getWidth(picture)
    srcHeight = getHeight(picture)
    for x in range(targetWidth):
        for y in range(targetHeight):
            srcX = int(x * float(srcWidth) / targetWidth)
            srcY = int(y * float(srcHeight) / targetHeight)
            px = getPixelAt(picture, srcX, srcY)
            tPx = getPixelAt(target, x, y)
            setColor(tPx, getColor(px))
    return target
```

```
def applyNegative(picture):
    for p in getPixels(picture):
        r = 255 - getRed(p)
        g = 255 - getGreen(p)
        b = 255 - getBlue(p)
        setColor(p, makeColor(r, g, b))
    return picture

def posterizeAndShift(picture):
    for p in getPixels(picture):
        r = getRed(p)
        g = getGreen(p)
        b = getBlue(p)
        if r < 100:
            r = 0
        else:
            r = 255
        if g < 100:
            g = 120
        else:
            g = 255
        if b < 100:
            b = 50
        else:
            b = 255
        setRed(p, r)
        setGreen(p, g)
        setBlue(p, b)
    return picture

def applyGrayscaleAndLighten(picture):
    for p in getPixels(picture):
        r = getRed(p)
        g = getGreen(p)
        b = getBlue(p)
        luminance = (r * 0.3) + (g * 0.59) + (b * 0.11)
        lightened = int(luminance + 40)
        if lightened > 255:
            lightened = 255
        setColor(p, makeColor(lightened, lightened, lightened))
    return picture

def applySepiaTone(picture):
    for p in getPixels(picture):
        r = getRed(p)
        g = getGreen(p)
        b = getBlue(p)
        luminance = (r * 0.3) + (g * 0.59) + (b * 0.11)
        newR = int(luminance * 1.35)
        newG = int(luminance * 1.15)
        newB = int(luminance * 0.85)
        if newR > 255: newR = 255
        if newG > 255: newG = 255
        if newB > 255: newB = 255
        setRed(p, newR)
        setGreen(p, newG)
        setBlue(p, newB)
    return picture
```

```
def applyRotatedSignature(canvas, sigImg, startX, startY):
    sigWidth = getWidth(sigImg)
    sigHeight = getHeight(sigImg)
    signatureColor = makeColor(255, 255, 255)
    targetWidth = 140
    targetHeight = 175
    for x in range(targetWidth):
        for y in range(targetHeight - 15):
            srcX = int(x * float(sigWidth) / targetWidth)
            srcY = int(y * float(sigHeight) / targetHeight)
            if srcX < sigWidth and srcY < sigHeight:
                sigPx = getPixelAt(sigImg, srcX, srcY)
                r = getRed(sigPx)
                g = getGreen(sigPx)
                b = getBlue(sigPx)
                if r < 120 and g < 120 and b < 120:
                    rotatedX = y
                    rotatedY = targetWidth - 1 - x
                    targetX = startX + rotatedX
                    targetY = startY + rotatedY
                    if targetX < getWidth(canvas) and targetY < getHeight(canvas):
                        canvasPx = getPixelAt(canvas, targetX, targetY)
                        setColor(canvasPx, signatureColor)
```