

# Kelsey Wolfert

Completed



Original



```
from mediaComp import *
#written by Kelsey Wolfert 3/10/2026
def collage():
    origin_picture = makePicture(getMediaFolder("Red Clover Field.jpg"))
    origin_picture = scale_change(origin_picture, 0.6) #had to shrink the picture so it fit the size requirements
    height = getHeight(origin_picture)
    width = getWidth(origin_picture)
    width_half = width // 2
    height_half = height // 2
    canvas = makeEmptyPicture(width, height)
    #unchanged section of original picture
    copyPicSection(origin_picture, canvas, 0, width_half - (width // 5), 0, height_half, 0, 0)
    invertColors(origin_picture, width_half - (width // 5), width_half, 0, height_half)
    #inverted section of picture
    copyPicSection(origin_picture, canvas, width_half-(width//5),width_half,0,height_half,width_half-(width//5),0)
    #unchanged section
    copyPicSection(origin_picture, canvas, width_half, width, 0, height_half - (height // 4), width_half, 0)
    colorSwap(origin_picture, width_half, width, height_half - (height // 4), height_half)
    #color swapped section of picture
    copyPicSection(origin_picture, canvas, width_half, width, height_half - (height // 4), height_half,
        width_half, height_half - (height // 4))
    colorSwapAlt(origin_picture, 0, width_half, height_half, height_half + (height // 4))
    #alternate color swapped section of picture
    copyPicSection(origin_picture, canvas, 0, width_half, height_half, height_half + (height // 4), 0,height_half)
    #unchanged section
    copyPicSection(origin_picture, canvas, 0, width_half,height_half+(height//4),height,0,height_half+(height//4))
    grayscale(origin_picture, width_half, width_half + (width // 5), height_half, height)
    #I added this because I wanted the grayscale part to be a bit brighter
    lightenDouble(origin_picture, width_half, width_half + (width // 5), height_half, height)
    #grayscale section of picture
    copyPicSection(origin_picture, canvas, width_half, width_half + (width // 5), height_half, height,
        width_half, height_half)
    #unchanged section
    copyPicSection(origin_picture, canvas, width_half + (width // 5), width, height_half, height,
```

```

        width_half + (width // 5), height_half)
signature = makePicture(getMediaFolder("Kelsey W. signature.jpg"))
signature = scale_change(signature, 0.05) #shrinking the signature to fit on the canvas appropriately
#coordinates for putting signature in bottom right of canvas
reverseChromaKey(signature, canvas, width - 200, height - 125)
show(canvas)

def scale_change(source_picture, scale_factor):
    new_width = int(getWidth(source_picture) * scale_factor)
    new_height = int(getHeight(source_picture) * scale_factor)
    new_picture = makeEmptyPicture(new_width, new_height)
    source_x = 0
    for new_x in range(0, new_width):
        source_y = 0
        for new_y in range(0, new_height):
            pixel = getPixelAt(source_picture, int(source_x), int(source_y))
            color = getColor(pixel)
            new_pixel = getPixelAt(new_picture, new_x, new_y)
            setColor(new_pixel, color)
            source_y = source_y + 1 / scale_factor
            source_x = source_x + 1 / scale_factor
    return new_picture

#copy but only for a specific slice
def copyPicSection(picture, new_picture, start_x, end_x, start_y, end_y, target_x, target_y):
    new_x = target_x
    for x in range(start_x, end_x):
        new_y = target_y
        for y in range(start_y, end_y):
            pixel = getPixelAt(picture, x, y)
            color = getColor(pixel)
            new_pixel = getPixelAt(new_picture, new_x, new_y)
            setColor(new_pixel, color)
            new_y = new_y + 1
            new_x = new_x + 1
    return new_picture

def invertColors(picture, start_x, end_x, start_y, end_y):
    for x in range(start_x, end_x):
        for y in range(start_y, end_y):
            pixel = getPixelAt(picture, x, y)
            red_value = getRed(pixel)
            green_value = getGreen(pixel)
            blue_value = getBlue(pixel)
            setRed(pixel, 255 - red_value)

```

```

        setGreen(pixel, 255 - green_value)
        setBlue(pixel, 255 - blue_value)
    return picture

def colorSwap(picture, start_x, end_x, start_y, end_y):
    for x in range(start_x, end_x):
        for y in range(start_y, end_y):
            pixel = getPixelAt(picture, x, y)
            red_value = getRed(pixel)
            green_value = getGreen(pixel)
            blue_value = getBlue(pixel)
            setRed(pixel, blue_value)
            setGreen(pixel, red_value)
            setBlue(pixel, green_value)
    return picture

def colorSwapAlt(picture, start_x, end_x, start_y, end_y):
    for x in range(start_x, end_x):
        for y in range(start_y, end_y):
            pixel = getPixelAt(picture, x, y)
            red_value = getRed(pixel)
            blue_value = getBlue(pixel)
            green_value = getGreen(pixel)
            setRed(pixel, green_value)
            setGreen(pixel, blue_value)
            setBlue(pixel, red_value)

def grayscale(picture, start_x, end_x, start_y, end_y):
    for x in range(start_x, end_x):
        for y in range(start_y, end_y):
            pixel = getPixelAt(picture, x, y)
            new_red = getRed(pixel) * 0.299
            new_green = getGreen(pixel) * 0.587
            new_blue = getBlue(pixel) * 0.114
            intensity = int(new_red + new_green + new_blue)
            setColor(pixel, makeColor(intensity, intensity, intensity))
    return picture

def lightenDouble(picture, start_x, end_x, start_y, end_y):
    for x in range(start_x, end_x):
        for y in range(start_y, end_y):
            pixel = getPixelAt(picture, x, y)
            color = getColor(pixel)
            color = makeLighter(color)
            color = makeLighter(color)

```

```
        setColor(pixel, color)
return picture

def reverseChromakey(foreground, background, target_x, target_y):
    for front_pixel in getPixels(foreground):
        x = getX(front_pixel)
        y = getY(front_pixel)
        if getRed(front_pixel) < 110 and getGreen(front_pixel) < 110 and getBlue(front_pixel) < 110:
            background_pixel = getPixelAt(background, target_x + x, target_y + y)
            setColor(background_pixel, pink)
```