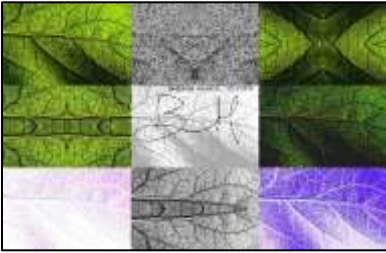


# Benjamin Hummel

Completed



Originals



```
def main():
    #first thing I do is set the media path to have the right location of my images
    pic = setMediaPath()

    #this section of code gets my signature properly formatted and scaled down,
    # and even rotated for display.
    sig = getMediaPath("newSignature.jpg")
    sig = makePicture(sig)
    sig = scaleDown(sig, 2.5)
    new = makeEmptyPicture(int(getWidth(sig)*3), int(getHeight(sig)*3))
    sig = rotateRight(sig, new)

    #This leaf image is available under the creative commons license
    pic = getMediaPath("leaf.jpg")

    #And right here is where all the pictures are made, and edited
    pic1 = makePicture(pic)
    pic1 = scaleDown(pic1, 5)
    pic = makePicture(pic)
    pic = scaleDown(pic, 5)
    pic2 = mirrorBotTop(mirrorRightToLeft(pic))
    pic = getMediaPath("leaf.jpg")
    pic = makePicture(pic)
    pic = scaleDown(pic, 5)
    pic3 = mirrorHorizontal(grayScale(pic))
    pic = getMediaPath("leaf.jpg")
    pic = makePicture(pic)
    pic = scaleDown(pic, 5)
    pic4 = negative(pic)
    pic = getMediaPath("leaf.jpg")
    pic = makePicture(pic)
    pic = scaleDown(pic, 5)
    pic5 = negative(lighten(sepiaTint(mirrorVertical(edgeDetect(pic)))))
    pic = getMediaPath("leaf.jpg")
    pic = makePicture(pic)
    pic = scaleDown(pic, 5)
    pic6 = posterize(mirrorVertical(mirrorHorizontal(pic)))
    pic = getMediaPath("leaf.jpg")
    pic = makePicture(pic)
    pic = scaleDown(pic, 5)
    pic7 = lighten(lighten(posterize(negative(pic))))
    pic = getMediaPath("leaf.jpg")
    pic = makePicture(pic)
    pic = scaleDown(pic, 5)
```

```

pic8 = posterize(darken(pic))
pic = getMediaPath("leaf.jpg")
pic = makePicture(pic)
pic = scaleDown(pic, 5)
pic9 = lighten(posterize(lighten(grayPosterize(pic))))

#this line assigns a canvas that is an exact 3x3 square my images can fit in
canvas = makeEmptyPicture(int(getWidth(pic) *3), int(getHeight(pic) *3), green)

#this is where the images are placed
copy(pic1, canvas, 0, 0)
copy(pic2, canvas, (getWidth(pic) *2), 0)
copy(pic3, canvas, getWidth(pic), (getHeight(pic)*2))
copy(pic4, canvas, (getWidth(pic) *2), (getHeight(pic)*2))
copy(pic5, canvas, getWidth(pic), 0)
copy(pic6, canvas, 0, getHeight(pic))
copy(pic7, canvas, 0, (getHeight(pic)*2))
copy(pic8, canvas, (getWidth(pic) * 2), getHeight(pic))
copy(pic9, canvas, getWidth(pic), getHeight(pic))

#adding my signature
chromaSig(sig, canvas, int(getWidth(pic) ), int(getHeight(pic)))
#adding the comment
canvas = addComment(canvas)
#finally saving and showing the picture
explore(canvas)
writePictureTo(canvas, "Fin.jpg")

#everything else below is a function used to manipulate pictures in some way

def addComment(picture):
    str = "Benjamin Hummel, 10/22/2021"
    addText(picture, 330, 175, str, black)
    return picture

def lighten(pic):
    for i in getPixels(pic):
        r = getGreen(i)
        g = getBlue(i)
        b = getRed(i)
        r = r + 75
        b = b + 75
        g = g + 75
        setGreen(i,r)
        setRed(i,b)
        setBlue(i,g)
    return pic

def darken(pic):
    for i in getPixels(pic):
        r = getGreen(i)
        g = getBlue(i)
        b = getRed(i)
        r = r - 75
        b = b - 75
        g = g - 75

```

```
    setGreen(i,r)
    setRed(i,b)
    setBlue(i,g)
return pic
```

```
def negative(pic):
    for i in getPixels(pic):
        r = getGreen(i)
        g = getBlue(i)
        b = getRed(i)
        r = 255 -r
        b = 255 -b
        g = 255 -g
        setGreen(i,r)
        setRed(i,b)
        setBlue(i,g)
    return pic
```

```
def grayScale(pic):
    for i in getPixels(pic):
        g = getGreen(i)
        b = getBlue(i)
        r = getRed(i)

        if g > r and g > b:
            x = g
        elif r > g and r > b:
            x = r
        else:
            x = b

        setGreen(i,x)
        setRed(i,x)
        setBlue(i,x)
    return pic
```

```
def grayScaleNew(pic):
    for i in getPixels(pic):
        r = getRed(i) * 0.299
        g = getGreen(i) * 0.587
        b = getBlue(i) * 0.114
        luminance = r+g+b
        setColor(i,makeColor(luminance,luminance,luminance))
```

```
def sepiaTint(pic):
    grayScaleNew(pic)
    for i in getPixels(pic):
        r = getRed(i)
        b = getBlue(i)

        if (r < 63):
            r = r*1.1
            b = b*0.9

        if (r > 62 and r < 192):
            r = r*1.15
            b = b*0.85

        if (r > 191):
            r = r*1.08
```

```

    if (r > 255):
        r = 255
    b = b*0.93
    setBlue(i, b)
    setRed(i,r)
return pic

def mirrorVertical(pic):
    mirrorPoint = getWidth(pic)/2
    width = getWidth(pic)
    for y in range(0,getHeight(pic)):
        for x in range(0,mirrorPoint):
            leftPixel = getPixel(pic,x,y)
            rightPixel = getPixel(pic,width -x -1,y)
            color = getColor(leftPixel)
            setColor(rightPixel, color)
    return pic

def mirrorRightToLeft(pic):
    mirrorPoint = getWidth(pic)/2
    width = getWidth(pic)
    for y in range(0,getHeight(pic)):
        for x in range(0,mirrorPoint):
            leftPixel = getPixel(pic,x,y)
            rightPixel = getPixel(pic,width -x -1,y)
            color = getColor(rightPixel)
            setColor(leftPixel, color)
    return pic

def mirrorHorizontal(pic):
    mirrorPoint = getHeight(pic) /2
    height = getHeight(pic)
    for x in range(0,getWidth(pic)):
        for y in range(0,mirrorPoint):
            topPixel = getPixel(pic,x,y)
            bottomPixel = getPixel(pic,x,height - y - 1)
            color = getColor(topPixel)
            setColor(bottomPixel,color)
    return pic

def mirrorBotTop(pic):
    mirrorPoint = getHeight(pic) /2
    height = getHeight(pic)
    for x in range(0,getWidth(pic)):
        for y in range(0,mirrorPoint):
            topPixel = getPixel(pic,x,y)
            bottomPixel = getPixel(pic,x,height - y - 1)
            color = getColor(bottomPixel)
            setColor(topPixel,color)
    return pic

def rotateRight(pic, canvas):
    targetX = 0
    width = getWidth(pic)
    height = getHeight(pic)
    for sourceX in range(0,getWidth(pic)):
        targetY = 0
        for sourceY in range(0,getHeight(pic)):
            color = getColor(getPixel(pic,sourceX,sourceY))
            setColor(getPixel(canvas,width - targetY -1, targetX), color)

```

```
    targetY = targetY + 1
    targetX = targetX + 1
return pic
```

```
def scale(src, canvas, factor):
    sourceX = 0
    for targetX in range(0, int(getWidth(src) * factor)):
        sourceY = 0
        for targetY in range(0, int(getHeight(src) * factor)):
            color = getColor(getPixel(src, int(sourceX), int(sourceY)))
            setColor(getPixel(canvas, targetX, targetY), color)
            sourceY = sourceY + 1.0 / factor
        sourceX = sourceX + 1.0 / factor
```

```
def scaleUp(pic, factor):
    canvas = makeEmptyPicture(int(getWidth(pic)*factor), int(getHeight(pic)*factor))
    scale(pic, canvas, factor)
    return canvas
```

```
def scaleDown(pic, factor):
    canvas = makeEmptyPicture(int(getWidth(pic) / factor), int(getHeight(pic)/factor))
    scale(pic, canvas, 1.0 / factor)
    return canvas
```

```
def posterize(pic):
    for i in getPixels(pic):
        r = getRed(i)
        b = getBlue(i)
        g = getGreen(i)

        if(r < 64):
            setRed(i,31)
        if(r > 63 and r < 128):
            setRed(i,95)
        if(r > 127 and r < 192):
            setRed(i,159)
        if(r > 191 and r < 256):
            setRed(i,223)

        if(g < 64):
            setGreen(i,31)
        if(g > 63 and g < 128):
            setGreen(i,95)
        if(g > 127 and g < 192):
            setGreen(i,159)
        if(g > 191 and g < 256):
            setGreen(i,223)

        if(b < 64):
            setBlue(i,31)
        if(b > 63 and b < 128):
            setBlue(i,95)
        if(b > 127 and b < 192):
```

```

        setBlue(i,159)
    if(b > 191 and b < 256):
        setBlue(i,223)
    return pic

def grayPosterize(pic):
    for i in getPixels(pic):
        r = getRed(i)
        b = getBlue(i)
        g = getGreen(i)
        luminance = (r+g+b)/3
        if luminance < 64:
            setColor(i,black)
        if luminance >= 64:
            setColor(i,white)
    return pic

def luminance(pixel):
    r = getRed(pixel)
    b = getBlue(pixel)
    g = getGreen(pixel)
    return (r+b+g)/3

def edgeDetect(pic):
    for i in getPixels(pic):
        x = getX(i)
        y = getY(i)
        if y < getHeight(pic)-1 and x < getWidth(pic)-1:
            botrt = getPixel(pic, x+1,y+1)
            thislum = luminance(i)
            brlum = luminance(botrt)
            if abs(brlum-thislum) > 10:
                setColor(i,black)
            if abs(brlum-thislum) <= 10:
                setColor(i,white)
    return pic

def copy(pic, target, targX, targY):
    targetX = targX
    for sourceX in range(0,getWidth(pic)):
        targetY = targY
        for sourceY in range(0,getHeight(pic)):
            px=getPixel(pic,sourceX,sourceY)
            tx=getPixel(target,targetX,targetY)
            setColor(tx,getColor(px))
            targetY = targetY + 1
        targetX = targetX + 1

def chromaSig(source, target, targetX, targetY):
    for x in range(0,getWidth(source)):
        for y in range(0,getHeight(source)):
            px = getPixel(source, x, y)
            color = getColor(px)
            targ = getPixel(target, x + targetX, y + targetY)
            if distance (black, color) < 200:
                setColor(targ, black)

```