def collage():
    getMediaPath()
    #Create the pictures to copy to the canvas
    pic1 = makePicture("turtle-dragon.jpg")
    pic2 = makePicture("turtle-dragon.jpg")
    pic3 = makePicture("turtle-dragon.jpg")
    pic4 = makePicture("turtle-dragon.jpg")
    pic5 = makePicture("turtle-dragon.jpg")
    sigPic = makePicture("signature.jpg")
    #Create background color for canvas
    cyan = makeColor(0,173,238)
    #Create canvas with background color
    canvas = makeEmptyPicture(1000, 736, cyan)

    #Task 1: Scale down original picture
    scaling(pic1, canvas,325,175,0.75)

    #Task 2: Scale down picture and invert the colors for two copies
    invertPic = inverted(pic2)
    scaling(invertPic, canvas,50,50,0.5)
    scaling(invertPic, canvas,715,446,0.5)

    #Task 3: Scale down picture and grayscale picture for two copies
    grayPic = grayed(pic3)
    scaling(grayPic, canvas,715,50,0.5)
    scaling(grayPic, canvas,50,446,0.5)

    #Task 4: Scale down picture and sepia-tone picture for single copy at top
    sepiaPic = sepiaTone(pic4)
    scaling(sepiaPic, canvas,440,50,0.25)

    #Task 5: Scale down picture and make picture lighter for single copy at bottom
    lighterPic = lighten(pic5)
    scaling(lighterPic, canvas,440,550,0.25)

    #Task 5: Copy signature file into canvas at bottom right
    copySig(sigPic,canvas, 520, 575)
show(canvas)

def inverted(pic):
    for pix in getPixels(pic):
        redValue = getRed(pix)
        greenValue = getGreen(pix)
        blueValue = getBlue(pix)
        invert = makeColor(255-redValue, 255-greenValue, 255-blueValue)
        setColor(pix, invert)
    return pic

def grayed(pic):
    for pix in getPixels(pic):
        redValue = getRed(pix)
        greenValue = getGreen(pix)
        blueValue = getBlue(pix)
        colorGray = (redValue + greenValue + blueValue)/3
        grayColor = makeColor(colorGray, colorGray, colorGray)
        setColor(pix, grayColor)
    return pic

def scaling(sourcePic, targetPic, txStart, tyStart, factor):
    sourceX = 0
    for targetX in range(txStart, txStart + getWidth(sourcePic)*factor):
        sourceY = 0
        for targetY in range(tyStart, tyStart + getHeight(sourcePic)*factor):
            sourcePix = getPixel(sourcePic, int(sourceX), int(sourceY))
            targetPix = getPixel(targetPic, targetX, targetY)
            sourceColor = getColor(sourcePix)
            setColor(targetPix, sourceColor)
            sourceY = sourceY + 1.0/factor
        sourceX = sourceX + 1.0/factor

def copySig(sourcePic, targetPic, targetX, targetY):
    sigColor = makeColor(0,0,0)
    for sourceX in range(0, getWidth(sourcePic)):
        for sourceY in range(0, getHeight(sourcePic)):
            sPix = getPixelAt(sourcePic, sourceX, sourceY)
            sColor = getColor(sPix)
            targetPix = getPixelAt(targetPic, sourceX + targetX, sourceY + targetY)
            if distance(sigColor, sColor)< 180:
                setColor(targetPix, red)

def sepiaTone(pic):
    grayed(pic)
    for p in getPixels(pic):
        red = getRed(p)
        blue = getBlue(p)
        if(red < 63):
            red = red*1.1
            blue = blue*0.9
        if(red > 62 and red < 192):
            red = red*1.15
            blue = blue*0.85
        if(red > 191):
            red = red*1.08
        if(red > 255):
red = 255
blue = blue*0.93
setRed(p, red)
setBlue(p, blue)
return pic

def lighten(pic):
    for pix in getPixels(pic):
        color = getColor(pix)
        lighterColor = makeLighter(color)
        setColor(pix, lighterColor)
    return pic://www.pxfuel.com/en/search?q=fir+Tree"