CSCI128 Lab 7

Chris Greencorn

Lab 7.1

1. Write a function picCopy which takes a source picture, target canvas, sBeginX, sBeginY, sEndX, sEndY as parameters, where sBeginX, sBeginY, sEndX, sEndY indicate the source's beginX, beginY, endX and EndY to be copied.

```
setMediaPath()

def picCopy(source,target,sBeginX,sBeginY,sEndX,sEndY):
  src = makePicture(source)
  trg = makePicture(target)
  targetX = 0
  for x in range(sBeginX,sEndX):
    targetY = 0
    for y in range(sBeginY,sEndY):
      color = getColor(getPixel(src,x,y))
      setColor(getPixel(trg,targetX,targetY),color)
      targetY = targetY + 1
    targetX = targetX + 1
  show(trg)
  return(trg)
```

2. Rewrite copyHorseFaceSmall to call the function picCopy to accomplish the copy of the horse face.

```
canvas = makeEmptyPicture(163,308)

def copyHorseFaceSmall():
  picCopy("horse.jpg",canvas,104,114,267,422)
```

3. Write code to output the total pixels copied and total columns copied.

```
def picCopy(source,target,sBeginX,sBeginY,sEndX,sEndY):
  src = makePicture(source)
  trg = target
  targetX = 0
  pxlcount = 0
  clmcount = 0
  for x in range(sBeginX,sEndX):
    targetY = 0
    for y in range(sBeginY,sEndY):
      color = getColor(getPixel(src,x,y))
      setColor(getPixel(trg,targetX,targetY),color)
      targetY = targetY + 1
      pxlcount = pxlcount + 1
    targetX = targetX + 1
    clmcount = clmcount + 1
  show(trg)
  return(trg)
  writePictureTo(trg,"HorseFaceSmall.jpg")
  print pxlcount," pixels and ",clmcount," columns copied."
```

4. Write code to output the total time used for the function copyHorseFaceSmall.

```
from time import clock

def picCopy(source,target,sBeginX,sBeginY,sEndX,sEndY):
 src = makePicture(source)
 trg = target
 start = clock()
 targetX = 0
 pxlcount = 0
 clmcount = 0
 for x in range(sBeginX,sEndX):
   targetY = 0
   for y in range(sBeginY,sEndY):
     color = getColor(getPixel(src,x,y))
     setColor(getPixel(trg,targetX,targetY),color)
     targetY = targetY + 1
     pxlcount = pxlcount + 1
   targetX = targetX + 1
   clmcount = clmcount + 1
 show(trg)
 return(trg)
 writePictureTo(trg,"HorseFaceSmall.jpg")
 print pxlcount,"pixels and",clmcount,"columns copied."
 print 'Time:',clock()-start,'s'
```

Lab 7.2

Write a function to copy a portion of a source picture to canvas while changing the colors of some pixels according to a criterion like the copyHorseFaceSmallBlack() function.

```
def copyPortionChangeSomeColours(source,target,sBeginX,sBeginY,sEndX,sEndY):
  hcol = makeColor(216,169,143)
  src = makePicture(source)
  trg = makePicture(target)
  targetX = 0
  for x in range(sBeginX,sEndX):
    targetY = 0
    for y in range(sBeginY,sEndY):
      if distance(color,hcol) < 40:
        setColor(getPixel(canvas,targetX,targetY), black)
      else:
        setColor(getPixel(canvas,targetX,targetY), color)
      targetY = targetY + 1
    targetX = targetX + 1
  show(trg)
  return(trg)
  writePictureTo(trg,"HorseFaceSmallBlack.jpg")
```

Lab 7.3

Write code to make a collage to put four pictures in the four corners and the center of a canvas. The top left picture is the original source picture. The top right is the mirror vertical picture along the middle vertical line. The center is the grey scale version of the original picture. The bottom left is a scale down picture and the bottom right is a rotation of the original picture. Carefully design the size of the canvas such that it can host all the pictures. Write code to output the total number of pixels written to the canvas.

```
def makeCollageWithFourPictures():
  setMediaPath()
  src = makePicture(pickAFile())
  canvas = makeEmptyPicture((getWidth(src)*3),(getHeight(src)*3))
  writePictureTo(canvas,'/tmp/canvas.jpg')
  trg = makePicture('/tmp/canvas.jpg')
  pxlcount = 0
# First Copy (Source, top left)
  targetX = 0
  for x in range(0,(getWidth(src)-1)):
    targetY = 0
    for y in range(0,(getHeight(src)-1)):
      color = getColor(getPixel(src,x,y))
      setColor(getPixel(trg,targetX,targetY),color)
      targetY = targetY + 1
      pxlcount = pxlcount + 1
    targetX = targetX + 1
# Second Copy (Vertical Mirror, top right)
  mirrorpoint = (getWidth(src)/2)
  targetX = ((2/3)*getWidth(trg))
  for x in range(targetX,(getWidth(trg)-1)):
    targetY = 0
    for y in range(0,(getHeight(trg)-1)):
      color = getColor(getPixel(src,(getWidth(src)-1),getHeight(src)-1))
      setColor(getPixel(trg,targetX,targetY),color)
      targetY = targetY + 1
      pxlcount = pxlcount + 1
    targetX = targetX + 1
# Third Copy (Greyscale, center)
  targetX = (getWidth(src))
  for x in range(targetX,(targetX+getWidth(src)-1)):
    targetY = (getHeight(src))
    for y in range(targetY,(targetY+getHeight(src)-1)):
      for p in getPixels(src):
        gs = ((getRed(p)*0.299)+(getBlue(p)*0.587)+(getGreen(p)*0.114))
        setColor(p,makeColor(gs,gs,gs))
      targetY = targetY + 1
      pxlcount = pxlcount + 1
    targetX = targetX + 1
# Fourth Copy (bottom left, scaled down)
  targetX = 0
  for x in range(0,((getWidth(src)/2)-1)):
    targetY = (2.5*(getHeight(src)))
    for y in range(0,((getHeight(src)/2)-1)):
      color = getColor(getPixel(src,x,y))
      setColor(getPixel(trg,targetX,targetY),color)
      targetY = targetY + 2
      pxlcount = pxlcount + 1
    targetX = targetX + 2
# Fifth Copy (bottom right, rotated)
  targetX = (getWidth(trg)-getHeight(src))
  for x in range(0,(getWidth(src)-1)):
   targetY = (getHeight(trg)-getwidth(src))
   for y in range(0,(getHeight(src)-1)):
     color = getColor(getPixel(src,x,y))
     setColor(getPixel(trg,targetY,targetX),color)
     targetY = targetY + 1
     pxlcount = pxlcount + 1
   targetX = targetX + 1

  show(trg)
  print pxlcount,'pixels copied to canvas.'
```