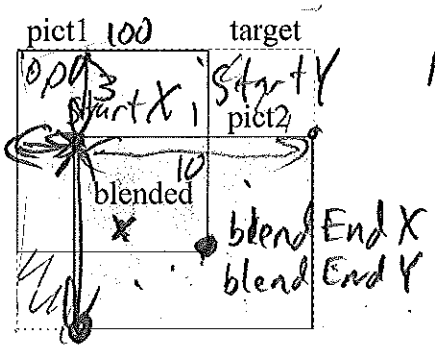
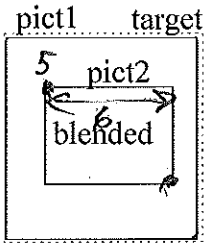
1. We can blend overlapping portions of pictures together by averaging the color components of each picture. This gives the appearance of transparency. For example, barbara.jpg and her daughter Katie-smaller.jpg with Katie's picture starting at x of 150.

The textbook's code (pp. 123-4) to blend these two images "hardcodes" the images and overlap region.

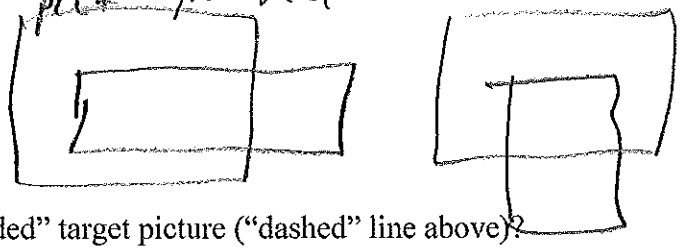
a) If we wanted to write a general function to blend two pictures what parameter should it be passed?




pict1, pict2, startX, startY, blendRatio




pictbackgroundColor = white



b) What would be the `width` and `height` of the "blended" target picture ("dashed" line above)?

c) With respect to the target picture's x and y coordinates what would be `blendStartX`, `blendEndX`, `blendStartY`, and `blendEndY`? (complete the code for both (b) and (c) below)

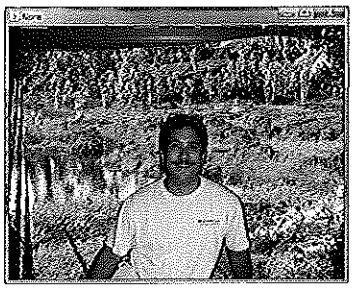
```
blendStartX = startX
if startX + getWidth(pict2) > getWidth(pict1):
  blendEndX = getWidth(pict1) - 1
  width = startX + getWidth(pict2)
else:
  blendEndX = startX + getWidth(pict2) - 1
  width = getWidth(pict1)
blendStartY = startY
if startY + getHeight(pict2) > getHeight(pict1):
  blendEndY = getHeight(pict1) - 1
  height = startY + getHeight(pict2)
else:
  blendEndY = startY + getHeight(pict2) - 1
  height = getHeight(pict1)
target = makeEmptyPicture(width, height)
```

d) Describe in English the loops and if-statements needed to build the blended target picture.

Nested loops to copy pict1 directly to target starting on 0,0.

Nested loops to copy pict2 to targetX and targetY starting at startX and startY, respectively. If a targetX, targetY is in the blended region, then set target pixel to blended color, otherwise set its color to that of pict2's pixel.

```
def blendPictures(pict1,pict2,startX,startY,fractionOfPict1=0.5,backgroundColor=white):
  """ Makes and returns a blending of two pictures with pict1 in the upper-left and
      pict2 starting at startX, startY coordinates. """

  fractionOfPict2 = 1 - fractionOfPict1
  blendStartX = startX
  if startX + getWidth(pict2) > getWidth(pict1):
    blendEndX = getWidth(pict1) - 1
    width = startX + getWidth(pict2)
  else:
    blendEndX = startX + getWidth(pict2) - 1
    width = getWidth(pict1)

  blendStartY = startY
  if startY + getHeight(pict2) > getHeight(pict1):
    blendEndY = getHeight(pict1) - 1
    height = startY + getHeight(pict2)
  else:
    blendEndY = startY + getHeight(pict2) - 1
    height = getHeight(pict1)

  target = makeEmptyPicture(width, height, backgroundColor)

  # copy pict1 to target
  for x in range(getWidth(pict1)):
    for y in range(getHeight(pict1)):
      pict1Pixel = getPixel(pict1, x, y)
      pict1PixelColor = getColor(pict1Pixel)
      targetPixel = getPixel(target, x, y)
      setColor(targetPixel, pict1PixelColor)

  targetX = startX
  pict1x = startX
  for x in range(getWidth(pict2)):
    targetY = startY
    pict1y = startY
    for y in range(getHeight(pict2)):
      pict2Pixel = getPixel(pict2, x, y)
      pict2Color = getColor(pict2Pixel)
      targetPixel = getPixel(target, targetX, targetY)
      if targetX >= blendStartX and targetX <= blendEndX and \
         targetY >= blendStartY and targetY <= blendEndY:
        pict1Pixel = getPixel(pict1, pict1x, pict1y)
        newRed = getRed(pict1Pixel)*fractionOfPict1 + getRed(pict2Pixel)*fractionOfPict2
        newGreen = getGreen(pict1Pixel)*fractionOfPict1 +   \
                   getGreen(pict2Pixel)*fractionOfPict2
        newBlue = getBlue(pict1Pixel)*fractionOfPict1 +    \
                  getBlue(pict2Pixel)*fractionOfPict2
        setColor(targetPixel, makeColor(newRed,newGreen, newBlue))
        pict1y = pict1y + 1
      else:
        setColor(targetPixel, pict2Color)
      targetY = targetY + 1

    targetX = targetX + 1
    pict1x = pict1x + 1
  return target   # returns the blended picture
```



2. *Background subtraction* and *Chromakey* both replace a known background with an image. For example, the above image of Mark Guzdial in front of a blue sheet is Chromakeyed into the jungle or Eiffel tower pictures.

```
def chromakeyPictures(foregroundPict, newBackgroundPict):
  width = min(getWidth(foregroundPict), getWidth(newBackgroundPict))
  height = min(getHeight(foregroundPict), getHeight(newBackgroundPict))
  target = makeEmptyPicture(width, height)

  for x in range(getWidth(target)):
    for y in range(getHeight(target)):
      targetPixel = getPixel(target, x, y)
      foregroundPixel = getPixel(foregroundPict, x, y)
      newBackgroundPixel = getPixel(newBackgroundPict, x, y)
      if getRed(foregroundPixel) + getGreen(foregroundPixel) < getBlue(foregroundPixel):
        setColor(targetPixel, getColor(newBackgroundPixel))
      else:
        setColor(targetPixel, getColor(foregroundPixel))

  return target   # returns the Chromakey picture
```

a) How are the width and height determined?

b) How is the blue detected?

c) How could this function be generalized?