Quiz 5, Comp 123

You may use the computer, and any resources on Moodle. You may also use any paper resources you like.
Place all answers in a single file and mark each question with a clearly visible comment.
You *may and should* ask me for clarifications, or hints.
You have 30 minutes to complete this quiz.

1. (10 points) You will write a function to convert a photo into a black and white (not gray-scale, only black and white) photo. For any pixel, it should be black if the luminance of that pixel (the average of the red, green, and blue values) is equal to or lower than a given cutoff. Otherwise the pixel should be white.

   Write a function `blackWhite(picture, cutoff)` which takes two parameters. The first parameter is a picture object representing the image to manipulate. The second parameter is a number representing the cutoff between white and black. Your function **should not** modify it's input image, it should instead duplicate that image, modify the duplicate and then return the duplicate. (Your function should be a pure function) The image your function returns should be the input image after being made black and white by the above definition. For examples of this see below.

(a) Original Image (b) blackWhite(image, 100) (c) blackWhite(image, 205)

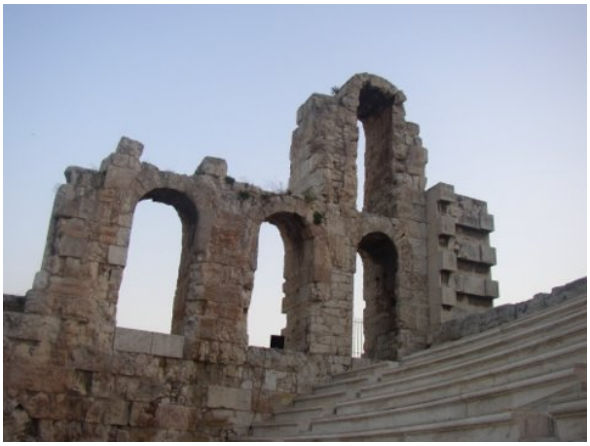
Figure 1: Examples of blackWhite function.

2. (20 points) Write a function called `gradualBlend`. It should take in two `Picture` objects. You can assume that the two images are the same size It should make a new picture of the same size as the images, and then it should blend the two pictures together. However, unlike the blend we performed in class, it should blend the pictures in a gradual way. At the left edge of the picture it should be mostly the first input picture, and at the right edge of the picture it should be mostly the second input picture.

   This is going to be similar to the weighted blending from the blending activity (remember that you can pull up that activity and your code from that activity). Unlike the weighted blending in the activity, however, we will need to use a different weight for every column of the image. There are several ways to do this. You can either calculate the weight as the current x position divided by the largest x position (so that the left most x position has weight 0 and the rightmost position has weight 1) or you can start with a weight of 0 and add $\frac{1}{getWidth(image)}$ each time the x value gets bigger. **The next page has sample calls to this function.**

```
pic1 = makePicture('greekRuins.jpg')
pic2 = makePicture('passionFlower.jpg')
pic3 = gradualBlend(pic1, pic2)
```

pic1



pic2

pic3

A note on submission You should submit only one file `quiz5.py`. You do not need to (and should not) compress (create a zip file) the file.