Use a convenient context such as either the [Algoritharium](http://people.uncw.edu/tompkinsj/algoritharium/index.html) ([updated jar file is here](http://people.uncw.edu/tagliarinig/courses/532/Lectures/algoritharium.jar)), the [Picture](http://people.uncw.edu/tagliarinig/courses/532/Lectures/Picture.java) class, or the necessary Python resources (DO NOT simply apply the library functions in PIL) to implement and demonstrate algorithms for:

1.  Color-to-gray-scale conversion. Is this an invertible operation?

2.  Convert an image to its negative. Is this an invertible operation?

3.  Find and “enhance” boundary contours.

4.  Rotating parts of an image.

5.  Warping parts of an image.(Nahar code)

6.  Implement image filters for smoothing and sharpening, as well as removing speckle

7.  Contrast enhancement (Is this an invertible operation?) and compare the results obtained via the histogram equalization method using:

        RGB space

Find histograms in the red, green, and blue color spaces and use the histograms to remap the RGB levels. Use the remapped values to assign pixel values in a contrast enhanced image. How would you describe the qualitative differences between the original image and the contrast enhanced image?

        HSB space

Transform the RGB values into HSB values (Hue, Saturation, and Brightness) to enhance an image using histogram equalizations over seven subsets HSB values (H, S, or B only, H and S, H and B, S and B, and HSB).

        What qualitative differences do you observe among the resultant images

        How do the images compare with the results obtained by RGB equalization?

8.  Compositing: Create and demonstrate the team’s implementation of algorithms to generate a composite of two or more images. For example, the team might replace the faces in a photograph of Mt. Rushmore with images of the team members’ faces.

9.  Create and demonstrate the team’s implementation of algorithms to identify a region in one image and search for a similar region in other images. For example, the team might take portrait images of its members, locate an “eye” in one of the images, and then search for “eyes” elsewhere in that image or the other team images. Can your team generalize the process to search for other features, such as key points on the lips, location of the nostrils or ears, or identification of the hair line (if any), in face images? Another application might be locating and labeling “heads” in a family portrait so that the individuals can be identified for an archive.

10.             Fading:  Create and demonstrate the team’s implementation of algorithms to fade from one image into another. For example, team member portraits could be used to form a fade sequence from one to another.

11.             Some test images:

a.    [Who](http://people.uncw.edu/tagliarinig/courses/532/Excercise%20data/testImages/who.jpg)

b.   [Boom](http://people.uncw.edu/tagliarinig/courses/532/Excercise%20data/testImages/boom.jpg)

c.    [Mars face](http://people.uncw.edu/tagliarinig/courses/532/Excercise%20data/testImages/marsface.jpg)

d.   [Shoemaker-Levy impact](http://people.uncw.edu/tagliarinig/courses/532/Excercise%20data/testImages/ShoemakerLevyImpactA.bmp)

e.    [Towers](http://people.uncw.edu/tagliarinig/courses/532/Excercise%20data/testImages/twin%20towers.bmp)

f.    [Tanks](http://people.uncw.edu/tagliarinig/courses/532/Excercise%20data/testImages/tanks.bmp)