**Assignment 1 (28 points)**

1. **Getting familiar with image manipulation in Matlab (or your favorite language) (*10*/*10*)**

Write a program/function that will:

1. Read and display an image
2. Calculate the size (total number of pixels) of the image
3. Calculate the maximum pixel value
4. Calculate the mean pixel value
5. Change the pixel values of the image in the following way: all pixels’ values less than the average calculated at (d) will be equal to 0 and all the others will be equal to 1. What type of image is the new generated image?
6. **Image Interpolation (*5/7*)**

Write a computer program that will, given an input image, reduce its spatial resolution, and then return it to its original resolution. Use all of nearest neighbor, bilinear and bicubic interpolation to do this. **481 Students (*2/0*):** Design your program such that the desired change in spatial resolution (e.g. 0.5, which will halve the image in each dimension, or 2.0, which will double the image in each dimension) is a variable input to your program. Show an example run of your code.

1. **Reducing the Number of Gray Levels in an Image (*8/11*)**

Write a computer program capable of reducing the number of gray levels in an image from 256 to 2, in integer powers of 2.  **481 Students (*3/0*):** Design your program such that the desired number of gray levels does not have to be a power of 2. Show an example run of your code.

**General submission instructions:**

1. Be kind to your aging, over-worked professor and submit only a single document. This can be pdf, MS Word, OpenOffice, etc. Do not submit a zip file.
2. Your single document should include the input image for your problem, if required, and answers to each of the sub-problems (text, image or both, as appropriate). For example, 1(e) will require you to show an output image and a text answer, 1(a) only an image, and 1(b) only text. Your document should also include code that you wrote to generate your answers.
3. You may use any images you like for the programming; I encourage you to use images that might be useful/interesting for your final project.
4. Feel free to use whatever functions MatLab supplies. Also feel free to write your own, if you are so inclined; it will take more time, but you will gain a deeper understanding of the material. It is one thing, for example, to implement bicubic interpolation using the matlab resize, quite another to write a bicubic interpolator yourself.
5. Feel free to use whatever programming language you like. Much of the class will use MatLab, but python, Java, C++ (and probably others) all have good support for image processing.
6. Point values for each question are indicated as ***x/y*** in which ***x*** is the point value for 481 students and ***y*** is the point value for 381 students.