Department of Computer Science
The University of Hong Kong
COMP 7502
Assignment 1

The purpose of this assignment is to get you familiar with the *Fast Fourier Transform* (FFT); *Inverse Fast Fourier Transform* (IFFT); and filtering in frequency domain. You are required to finish the missing code in the template provided to you.
Please be reminded of the policy regarding plagiarism:
https://intranet.cs.hku.hk/csintranet/contents/general/shared/plagiar.jsp

Please note the following:

1. ONLY hand-in your modified version of Assignment1.java. Do not modify any other files.
2. You should regard an image as an array of complex numbers. This complex number representation of the image is designed to ease your implementation. A complex number class is also provided so that you can perform complex number arithmetic in an easy manner.
3. The template will pad the image with zeros so that each input image is of size $2^m \times 2^n$. As a result, you do not need to check and pad an image in your implementation.
4. You should implement additional helper methods whenever it is necessary, e.g. to implement the recursive 1D fast Fourier transformation.

You are required to complete the following tasks in Assignment1.java:

public void fourierTransform(byte[] img, int width, int height)
Implement the FFT and display the Fourier spectrum. Results should be equivalent to those obtained in the Workshop 3. Implementation details can be obtained from section 4.11 of the textbook, which is available on moodle. (6 Marks)

changeImage(byte[] img, int width, int height)
Describe the appearance of the Fourier spectrum that is obtain from the image created in this function. Why does the Fourier spectrum appear as it does? Provide a detailed explanation as a comment to this function. No need to implement any code here. Hint: Read chapter 4 of the textbook. (2 Marks)

public void filtering(byte[] img, int width, int height, double d0)
Perform second order (n=2) ButterWorth low pass filtering in the frequency domain. (5 Marks)

public void filtering2(byte[] img, int width, int height)
Apply a suitable filter to the car image to attenuate the *impulse-like* bursts in the image. (2 Marks)