Course: Introduction to Image Processing  
Homework: #1  
Due date: 3/9, before class

This is individual work. However, peer discussion is encouraged. For every programming assignment, clearly document your code so that the TA can follow your algorithms easily. Include the output images of your functions in the report.

Question 1. [20] Implement your versions of ind2rgb, ind2grey and ind2bw in Matlab. You should test your code with tree.gif and compare the result with the images in slide15 of lecture2.ppt. Include the output images with your work. Use of while or for loops is not allowed. Hint: the function reshape might be useful.

Question 2. [10] High-definition television (HDTV) generates images with a resolution of 1125 horizontal TV lines interlaced (where every other line is painted on the tube face in each of two fields, each field being 1/60th of a second in duration). The width-height aspect ratio of the images is 16:9. A company has designed an image capture system that generates digital images from HDTV images. The digital images have the same resolution as the TV. The color image consists of a red, a green, and a blue image, each having 8 bits for a pixel. How many bits would it take to store a 2-hour HDTV program? Show your analysis in detail.

Question 3. [25] Using lenna.png (a famous picture in DIP), perform the following tasks:
   a. Write a Matlab function that takes the number quantization levels as the input parameter and perform quantization using dithering. Test the function with quantization of 3 bits, 2 bits and 1 bit per pixel.
   b. Write a function to perform quantization using Jarvis-Judice-Ninke error diffusion. Test the function with quantization of 3 bits, 2 bits and 1 bit per pixel.
   c. Report your observation as the image is quantized using 8 down to 2 gray levels using the same method, and compare the performance of the 2 methods.

   For each function, enhance the image contrast for display purpose.

Question 4. [10] The following table gives the number of pixels at each of the gray levels 0-15 in an image with those gray values only. Draw the histogram corresponding to these gray levels and then perform histogram equalization and draw the resulting histogram.

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>20</td>
<td>40</td>
<td>60</td>
<td>75</td>
<td>80</td>
<td>75</td>
<td>65</td>
<td>55</td>
<td>50</td>
<td>45</td>
<td>40</td>
<td>35</td>
<td>30</td>
<td>25</td>
<td>20</td>
<td>30</td>
</tr>
</tbody>
</table>
Question 5. [25] Write your own Matlab functions to perform histogram equalization using the distributions below:
   a. uniform distribution  
   b. exponential distribution

Test your function with the lenna_low_contrast.png and compare the difference visually.

Question 6. [10] About bit-planes:
   a. What effect would setting to zero the lowest-order bit plane have on the histogram?
   b. What would be the effect on the histogram if we set to zero the highest-order bit planes instead?