Instructions:

- This is a two-hour closed book final exam.
- You are allowed one 8.5x11 page of notes.
- Answer all of the questions below.

Name: ______________________

Question #1 (20 points)

- Describe how histogram equalization operates. Be specific about the algorithm and use equations and diagrams as necessary to explain how this technique works.
- Apply histogram equalization to the 5x5 image below to perform image enhancement. You are welcome to use any output pixel range you wish.

```
2  2  3  9  9
2  2  3  9  9
3  3  3  9  9
9  9  9  9  9
9  9  9  9  8
```

Question #2 (20 points)

The traditional method to implement neighborhood averaging is to perform a convolution with an MxM mask of 1’s. One way to improve the computation time is to take advantage of the fact that the neighborhood for [x+1,y] will overlap most of the neighborhood for [x,y]. Hence, we can calculate the average for pixel [x+1,y] in terms of the average for pixel [x,y] plus M pixels that are added to the right side of the [x+1,y] region, minus M pixels that are removed from the left side of the [x,y] region.

- How many operations are needed to calculate each output value using the traditional neighborhood averaging for MxM masks as M varies from 1 to 10?
- How many operations are needed to implement the fancy neighborhood averaging algorithm described above as M varies from 1 to 10?
- Do you think this approach can be used for any convolution operation? Explain why or why not.
- Do you think this approach can be extended to work for median filtering? Explain why or why not.
Question #3 (20 points)
Assume that you are given an NxN image and you want to calculate the 2D Fourier Transform.

• Describe how this 2D FT can be computed using 1D Fourier Transforms.
• How many operations will this take if we use the slow 1D Fourier Transform?
• How many operations are needed if the 1D Fast Fourier Transform is used?

Assume we are given an NxN image f(x,y) and we calculate the 2D Fourier Transform F(u,v). Then we create another NxN image G(u,v) as follows:

    for (int u=0; u<N; u++)
    for (int v=0; v<N; v++)
        G(u,v) = F(u/2,v/2);

• What will G(u,v) look like compared to F(u,v)?
• What will g(x,y) = IFT{ G(u,v) } look like compared to f(x,y)?

Question #4 (20 points)
Suppose that someone has given you a side-view photograph of a car speeding past the finish line of a race. Unfortunately, the camera shutter speed was too slow to capture the image clearly. Describe how you could recover a clear picture of the car in the image using image restoration.

• Describe the degradation model you think is appropriate for this case. Try to be specific about the blurring process for the car.
• Describe how your favorite image restoration method could be used to restore this image. Give the equations explaining your method, and the specific sequence of steps necessary to perform the restoration.
• Once your restoration has been applied to the input image, what do you expect the car to look like? What do you expect the background to look like?
Question #5 (20 points)

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>4</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>6</td>
<td></td>
</tr>
</tbody>
</table>

- Calculate the intensity histogram for the image above. Using this, calculate the Huffman code tree. What are the resulting Huffman codes for each pixel value?
- What is the average number of bits per pixel needed to compress this image using your code? Do you think this is optimal? Explain.