Name:

Instructions: Read questions carefully before answering. Do not hesitate to ask for clarifications. Show all work. Partial credits are given, so do not leave anything blank! Ask for extra paper if you need it! Good luck!

Bonus Question (1 point) What was your favorite topic in this class?

Question 1: (20 points) Basic Definitions

(5 points each) Define the following terms

a. Post-filtering antialiasing method

b. Distinguish between the umbra and penumbra of a shadow

c. Mach Band effect caused by flat shading

d. Which type of fog mimics the real world better? Linear fog or exponential fog? Explain
(16 points) Assuming that the vertices of a 2D polygon have the properties described in Table 1, what is the color at location (70, 60) when Gouraud shading is used?

<table>
<thead>
<tr>
<th>Vertex</th>
<th>X, Y</th>
<th>R, G, B (after Phong lighting model has been applied)</th>
</tr>
</thead>
<tbody>
<tr>
<td>V₁</td>
<td>10, 20</td>
<td>100, 200, 10</td>
</tr>
<tr>
<td>V₂</td>
<td>90, 20</td>
<td>200, 10, 100</td>
</tr>
<tr>
<td>V₃</td>
<td>90, 120</td>
<td>10, 100, 200</td>
</tr>
<tr>
<td>V₄</td>
<td>10, 120</td>
<td>10, 100, 200</td>
</tr>
</tbody>
</table>
**Question 3 Projection (6 points)**

Fill in the blank
(6 points) When applying a perspective transformation, lines that were parallel to the z axis before projection become projected to lines that are _______________ after projection.

**Question 4: (10 points) Lighting and Shading**

a. (6 points) Distinguish between the **lighting** and **shading** operations.

b. (4 points) Which term of the Phong shading model do we increase or reduce in order to make a dull plastic surface have more shiny highlights? Explain.
Question 5: (22 points) Raster Graphics

(22 points) Using the following flood-fill algorithm code, for each white square in figure 1 below, write the number corresponding to the order in which it will be filled. The "seed" pixel is denoted by a '1'.

```c
void floodFill( int x, int y, int initColor, int newColor ) {
    if( getPixel( x, y ) == initColor ) {
        setPixel( x, y, newColor );
        floodFill( x - 1, y + 1, initColor, newColor ); // upper-left
        floodFill( x    , y + 1, initColor, newColor ); // upper-middle
        floodFill( x + 1, y + 1, initColor, newColor ); // upper-right
        floodFill( x - 1, y    , initColor, newColor ); // left
        floodFill( x + 1, y    , initColor, newColor ); // right
        floodFill( x - 1, y - 1, initColor, newColor ); // lower-left
        floodFill( x    , y - 1, initColor, newColor ); // lower-middle
        floodFill( x + 1, y - 1, initColor, newColor ); // lower-right
    }
}
```

[Figure 1: Flood-fill region]
Question 6 (14 points): Texture Mapping

(14 points) During cube mapping, the reflected vector $\mathbf{R}$ is calculated as $(x,y,z) = (8,6,3)$. Using $\mathbf{R}$ to look up the cube map, what face of the cube is looked up? What are the values of $s$ and $t$ used to perform this look up?
**Question 7: (12 points) Clipping**

The Liang-Barsky clipping algorithm continuously calculates and updates the values of \( t_{in} \) and \( t_{out} \) and exits if ever \( t_{out} < t_{in} \). Using a 2D square in place of the actual 3D Canonical View Volume (CVV), show a **straight line** that goes from Point A to Point C, which will trigger this condition. Make sure you label clearly points A, C and also the hit points corresponding to \( t_{in} \) and \( t_{out} \) in your diagram.