THE UNIVERSITY OF BRITISH COLUMBIA
Midterm Examination – 9 Feb 2001

Computer Science 414 – Section 201
Introduction to Computer Graphics

Time: 45 minutes

Student’s Name: ____________________________

(Please print in BLOCK letters, SURNAME first.)

Student Number: __________________________

Signature: _________________________________

Instructor’s Name: Rob Scharein

This examination consists of 10 pages, including this cover page.
Check to ensure that this exam is complete.

This is a closed book examination.
The weight of each question is given in parentheses. The total number of marks is 55. Start with
the questions you think are easiest, and then go back and do the harder ones. Show all your work.
Good Luck!

1. Each candidate should be prepared to produce, upon request, his/her library card.

2. READ AND OBSERVE THE FOLLOWING RULES:

- No candidate shall be permitted to ask questions of the invigilators, except in the cases of supposed errors or ambiguities in examination questions.

- **CAUTION** – Candidates suspected of any of the following, or similar, dishonest practices shall be immediately dismissed from the examination and shall be liable to disciplinary action.

  (a) Having at the place of writing communication devices, any books, papers or memoranda, audio or visual cassette players, mobile phones, or other memory aid devices other than that specifically approved by the instructor.

  (b) Speaking or communicating with other candidates.

  (c) Purposely exposing written papers to the view of other candidates. The plea of accident or forgetfulness shall not be received.

- **Smoking is not permitted during examinations.**

<table>
<thead>
<tr>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
<th>Q4</th>
<th>Q5</th>
<th>Q6</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>5</td>
<td>6</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>55</td>
</tr>
</tbody>
</table>
Use the pictures of “Lupin” above to answer the following question. In the following, assume that \texttt{Lupin()} is a function that draws a picture of Lupin filling a unit square in the $xy$-plane with vertices $(0.5, 0.5)$, $(-0.5, 0.5)$, $(-0.5, -0.5)$, and $(0.5, -0.5)$ and that \texttt{draw_rectangular_grid()} is a function that draws grid lines on the $xy$-plane with a thick line spacing of 1.0 and a thin line spacing of 0.5.
Suppose we have the following display callback function in our drawing program.

```c
void paint (void) {
    glClear (GL_COLOR_BUFFER_BIT);
    glMatrixMode (GL_MODELVIEW);
    glLoadIdentity ();
    gluLookAt (0.0, 0.0, 3.0, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0);

    glPushMatrix (); draw_rectangular_grid (); glPopMatrix ();
    draw_stuff ();
    glutSwapBuffers ();
}
```

For each of the following definitions for the body of the function `draw_stuff()`, match the output of our program with a picture on the previous page. If no picture matches, indicate this with the word `none`. Each box below is worth 2 marks (except the first one).

Matching picture: 1
Lupin ();

Matching picture: 
```
Lupin ();
```

Matching picture: 
```
Lupin ();
```
Question 2  (5 marks)

Explain what the GLUT function `glutPostRedisplay()` is used for and how you would use it in an application. What are the restrictions, if any, on where the function should be called?

Question 3  (6 marks)

Consider that points and vectors in three-dimensional space can be represented in either homogeneous or non-homogeneous coordinates. Explain how to transform points and vectors between these two representations. Why do we use homogeneous coordinates in computer graphics anyway?
Question 4  (10 marks)

Explain what Euler angles are and how they are used in computer graphics (give an example). What is one potential problem with using Euler angles?
Question 5  (10 marks)

Given the 3D vectors \( \mathbf{a} = a_x \mathbf{i} + a_y \mathbf{j} + a_z \mathbf{k} \) and \( \mathbf{b} = b_x \mathbf{i} + b_y \mathbf{j} + b_z \mathbf{k} \), their cross product is denoted by \( \mathbf{a} \times \mathbf{b} \). It is defined in terms of the standard unit vectors \( \mathbf{i}, \mathbf{j}, \mathbf{k} \) by

\[
\mathbf{a} \times \mathbf{b} = (a_y b_z - a_z b_y) \mathbf{i} + (a_z b_x - a_x b_z) \mathbf{j} + (a_x b_y - a_y b_x) \mathbf{k}.
\]  

(1)

Instead of presenting it as a definition, this form of the cross product can be derived from more fundamental ideas. We need only assume that

(a) The cross product operation is linear.

(b) The cross product of a vector with itself is zero.

(c) \( \mathbf{i} \times \mathbf{j} = \mathbf{k}, \mathbf{j} \times \mathbf{k} = \mathbf{i}, \text{ and } \mathbf{k} \times \mathbf{i} = \mathbf{j} \).

NOTE: Do either Part A or Part B, but not both.

Part A

Show that the above assumptions imply that the cross product is anti-symmetric (recall this means that \( \mathbf{e} \times \mathbf{f} = -\mathbf{f} \times \mathbf{e} \) for any two vectors \( \mathbf{e} \) and \( \mathbf{f} \)).
Part B

By writing $a = a_x \mathbf{i} + a_y \mathbf{j} + a_z \mathbf{k}$ and $b = b_x \mathbf{i} + b_y \mathbf{j} + b_z \mathbf{k}$, and using the three assumptions together with the property of anti-symmetry, derive the proper form for $\mathbf{a} \times \mathbf{b}$ given in equation 1 above.
Question 6  (10 marks)

Euler’s formula for a polyhedron is

\[ V + F - E = 2 \]

where \( V, \) \( F, \) and \( E \) are the number of vertices, faces, and edges in the polyhedron, respectively. Show that this relationship is preserved if we take an arbitrary polyhedron and subdivide its faces into triangles (you don’t have to show this for all possible triangulations, just pick one method for triangulating faces and prove the result for that case).
Extra space in which to work
More extra space in which to work