COMP 3501 Final Exam

December 2009

180 Minutes. 200 marks. 2 pages. Read each question carefully. Answer succinctly: a portion of marks awarded are for style and clarity. Please use ink or a soft lead pencil.

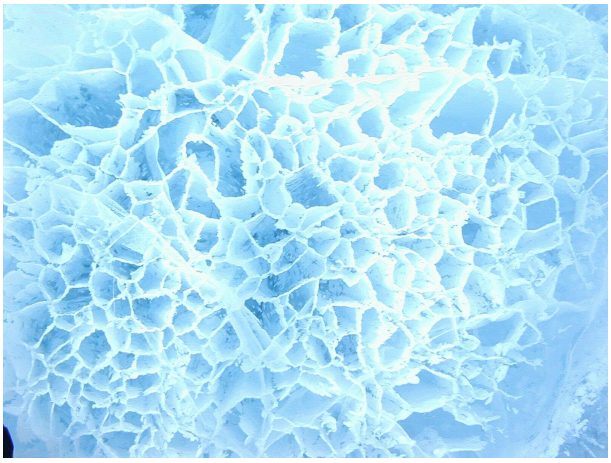
Write your answers in the answer booklet.
You may refer to a sheet of notes you brought with you.

1. [10] Describe the flow of control through an XNA game, from initialization to closing, possibly assisted by a diagram.

2. [20] Two crucial vector operations are the *dot product* and the *cross product*.
(a) Give three examples of significant applications of the dot product, with details about how the dot product is used.
(b) As (a), but for the cross product.

3. [15] Writing shader code makes use of so-called "semantics".
(a) Explain what is meant by "semantics" in this context.
(b) Your friend has written a pixel shader and tries to access the pixel's position, which has been defined as part of the pixel shader input structure. The shader won't compile. Explain what the source of the problem is, in as much detail as possible.

4. [15] To use texture, you need to supply texture coordinates for each point on a model's surface. The model you want to texture is a sphere. Describe a method for coming up with texture coordinates for the sphere, and mention any disadvantages of your approach.



5. [20] You have functions PNoise (Perlin noise) and WNoise (Worley noise) at your disposal, plus various common mathematial functions and operators. Write pseudocode for a procedure that generates a texture resembling the above image. (It doesn't have to be an exact match – just do your best.) Explain the reasoning behind your pseudocode.

6. [20] The Z-buffer is a key algorithm for modern real-time rendering.
(a) Explain what problem the Z-buffer solves and how it solves it.
(b) The z-buffer algorithm can sometimes display a defect called "z-fighting", where two nearby objects flicker: sometimes one seems to be in front, sometimes the other. Why does z-fighting occur and how can it be addressed?

7. [20] You have a character (in 2D) standing exactly at (-2,2) pointing his gun parallel to the X-axis. Write a matrix transformation that takes him to (1,1) with his gun pointing parallel to the Y-axis. If you like, you may leave the final outcome as a series of matrices (that is, you do not actually need to multiply the matrices together).

8. [15] Write a unit quaternion that rotates the camera (in 3D) from looking in the positive X direction to looking in the positive Y direction.

9. [15] You are planning on using a particle system to model atmospheric effects: more will be added later, but for now *falling snow* and *windblown dust* are your targets.
(a) Identify the key design decisions you will encounter in implementing this idea.
(b) For both the snow and the dust effect, make preliminary suggestions about how to resolve each of the decisions you listed in part (a).

10. [20] "Bump mapping" is a technique that alters an object's surface normal vector based on a texture lookup. In the following, assume that the object's color will be calculated using the 3-term lighting model.
(a) Which portions of the lighting calculation are most strongly affected by bump mapping? Say a few words about the overall lighting changes when bump mapping is applied.
(b) You have just heard of a new technique called "gaze mapping", that uses a texture to adjust the view direction. Predict how gaze mapping will affect lighting calculations.
(c) Suggest a possible application (e.g., a specific special effect) for gaze mapping.

11. [15] You are writing a game of microscopic exploration in which the player, piloting a airplane the size of a wasp, must dodge hazards including hailstones, falling leaves, and blades of grass. One of the challenges in developing this game is the collision detection. For each of the mentioned objects – the airplane, hailstones, leaves, and grass – suggest what bounding geometry could be used and say why you think it is a good choice.

12. [15] The *convolution* is a versatile image processing operator.
(a) Describe the convolution, and write pseudocode for calculating C*I for a convolution kernel C and an image I. State clearly any additional assumptions you need.
(b) Give two applications of the convolution and say how to adjust the pseudocode you gave for part (a) to produce the given effects.

13. [bonus, 1] In mipmapping, what does MIP stand for?

14. [bonus, 1] Give the name of a commercial game that used one of the techniques learned in class, and say what technique you are referring to.