

# Computer Graphics

## ECSE-4750 MidTerm Exam

October 20, 2015

Name \_\_\_\_\_

You have the whole class to finish the exam, take your time and remember to answer the questions you know first. Each question is worth 5 points total with the points divided between the various parts. Please feel free to use the back of the page.

### 1. OpenGL

a. (1 Pt) What is OpenGL?

**A platform independent, 2D and 3D graphics API. Graphics are rasterized and scan converted to the resultant image as the function call is made using the current state.**

b. (1 Pt) What is meant by immediate and retained mode graphics?

**As you issue OpenGL drawing commands, the specified object is drawn to a display.**

**Retained is when graphic primitives are held by the library and there is no need to respecify,**

c. (3 Pts) Name and describe each of the material color components.

**Emitted light is the simplest, it originates from an object and is unaffected by any light sources.**

**The ambient component is the light from that source that's been scattered so much by the environment that its direction is impossible to determine. When ambient light strikes a surface it is scattered in all directions.**

**Diffuse light comes from one direction, so it is brighter if it comes down squarely on the surface rather than at an angle. Any light coming from any direction has a diffuse component.**

**Specular light comes from a direction and bounces off a surface in a particular direction. Shiny metal and plastic have a high specular component and chalk and carpet have almost none. You can think of specularly as shininess.**

## 2. OpenGL

a. (2 Pts) Explain why OpenGL needs a utility library.

**OpenGL was designed as a platform independent API. Therefore, it needs an extra level of interface to user interfaces/window systems and display hardware. Since there is no retained mode we also need higher level data/geometric objects.**

b. (3 Pts) Explain what is happening with this OpenGL code snippet, and any possible problems.

```
// look up where the vertex data needs to go.
var positionLocation = gl.getAttribLocation(program, "a_position");

var buffer = gl.createBuffer();
gl.bindBuffer(gl.ARRAY_BUFFER, buffer);
gl.bufferData(
    gl.ARRAY_BUFFER,
    new Float32Array([
        -1.0, -1.0,
        1.0, -1.0,
        -1.0, 1.0,
        -1.0, 1.0,
        1.0, -1.0,
        1.0, 1.0]),
    gl.STATIC_DRAW);
gl.enableVertexAttribArray(positionLocation);
gl.vertexAttribPointer(positionLocation, 2, gl.FLOAT, false, 0, 0);

gl.drawArrays(gl.TRIANGLES, 0, 2);
```

**The snippet is supposed to draw 2 triangles. The problem is the gl.drawArrays is supposed to specify the number of coordinates in the array, it should say 6.**

### 3. Transforms

a) (2 Pts) What does this do to the input x and y.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} sx & 0 \\ 0 & sy \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$

**Scales each, the x by sx and y by sy.**

b) (3 Pts) Based on our RobotArm and knowing state is inherited from the parent, describe the position and color of the shoulder, elbow, and wrist.

```
shoulder = geoms[0];
shoulder.setColor([0,0,1]);

elbow = geoms[0].copy();
elbow.setTranslate([0,0,1]);
elbow.setColor([0,0,1]);

wrist = elbow.copy();
wrist.setTranslate([0,0,-1]);

shoulder.addGeom(elbow);
elbow.addGeom(wrist);
```

**Shoulder Pos: 0,0,0 Color: Blue**

**Elbow Pos: 0,0,1 Color: Blue**

**Wrist Pos: 0,0,0 Color: Blue**

## 4. Data Storage

a. (3Pts) The choice of data representation is important because it effects our ability to interface to external data and the performance of the overall visualization system. Name and describe 3 of the 5 design criterion for our visualization system described in class and in the text.

- . **Compact** Since we know graphics data tends to be very large, we need to design our structures to minimize computer memory requirements.
- . **Efficient** Scene data needs to be accessed many times and we would like to retrieve and store data in our structures in constant time.
- . **Mappable** Data representations need to efficiently map into graphics primitives. We must be able to easily convert external data into internal graphics data structures.
- . **Minimal Coverage** We must create a data representation that minimizes the number of data types while covering a large portion of external data types. We wish to balance efficiency with the number of data types.
- . **Simple** Simple algorithms and data sets are easier to implement and also easier to extend.

b. (2 Pts) If we group our Actor classes and contained Cells as data sets, describe what is meant by structure and attributes.

**Structure is made of two components:**

**Topology:** the information about the structure of the dataset (e.g. Triangle).

**Geometry:** the instantiation of the Topology (the x,y,z's).

**Attributes** are supplemental information associated with the geometry such as color, temperature, pressure, or density. They consist of scalars, vectors, normals, texture coordinates, tensors, and user defined data.

## 5. Graphics systems

- a. (3 Pts) Using the objects from robot example, discuss how to include animation object into the class hierarchy. Specifically, explain where the algorithm would exist (in its own object or attached to some other object) also would there be a need to change the class hierarchy.

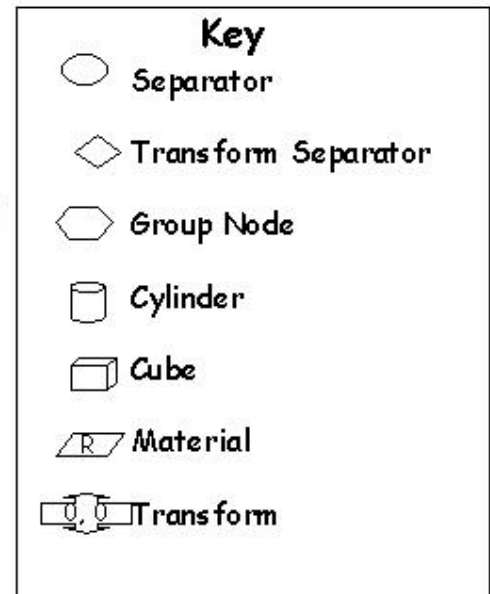
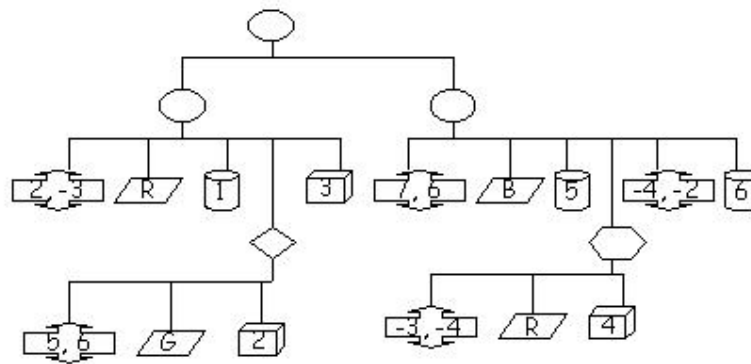
**I would create a secondary class to control the interactions with the graphical entities. There would be an animation loop, at each iteration the animation algorithm would need to set the movement (or transforms) of the graphical objects to set their current state. Then a render is called that traverses the scene and instructs all graphical entities to render themselves in order with containing objects forwarding the call to their children.**

- b. (2 Pts) Based on our discussion of scene graphs, explain how properties (materials and geometry) are shared.

**The graph is a depth first traversal of the branches and the nodes represent geometry, transforms, and materials. As the graph is traversed the current state of the system is accumulated, pushed, and popped according to the graph structure. Nodes inherit from parents and are effected by what comes prior in the graph.**

## 6. Data Algorithms

(5 Pts) Based on our discussions of the Scene Graph structure, fill in the blanks below with the position and color of each of the objects.



Object	Position	Color
1	2, -3	R
2	7, 3	G
3	2, -3	G
4	4, 2	R
5	7, 6	B
6	0, 0	R

### Extra credit:

(2 Pts) What is the acronym GLSL?

### OpenGL Shading Language.

(2 Pts) What are some weaknesses of the Object Rendering system we used for hw2?

**Lots of things: no material handling, no per vertex shading, no real collection of base primitives, no easy way to create multiple copies, limited animation functionality.**