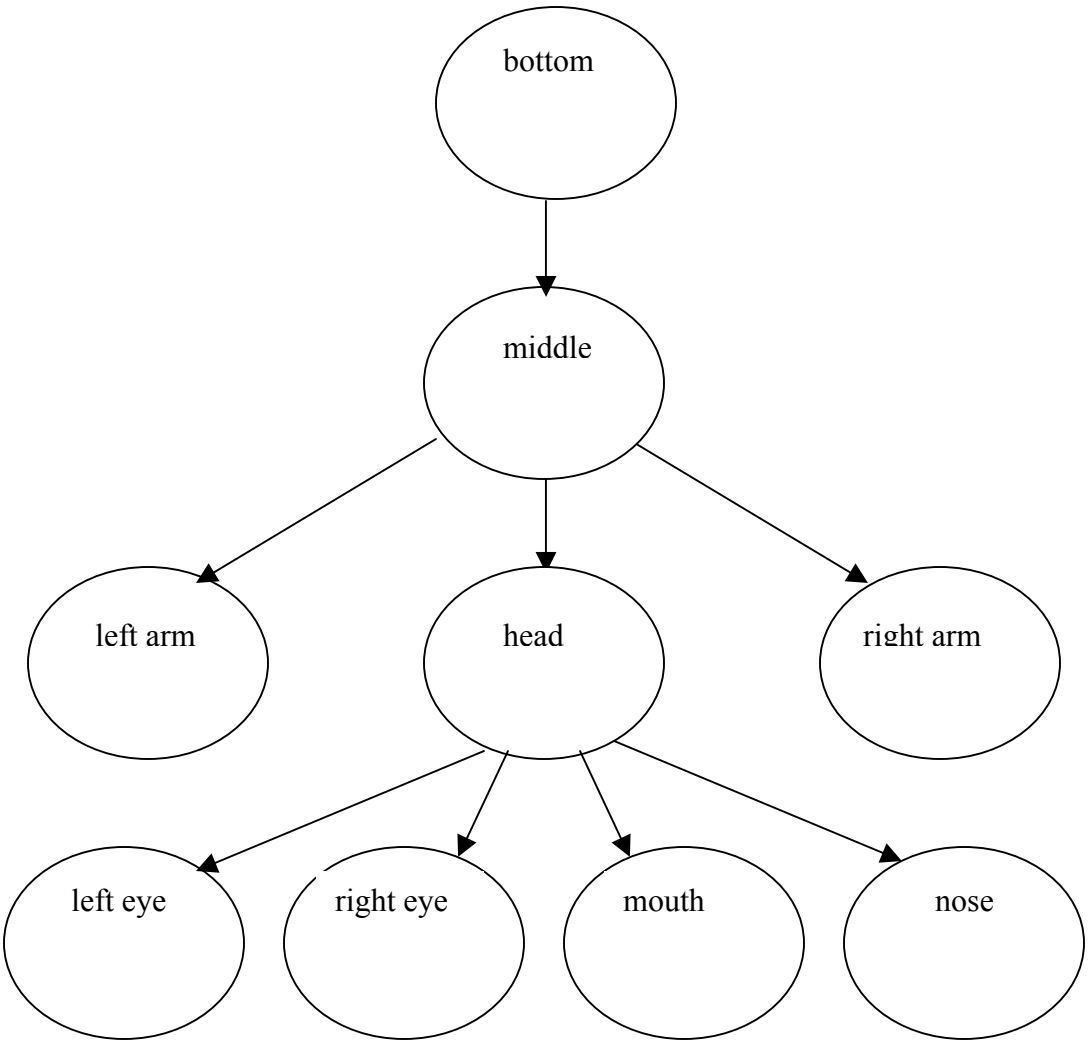
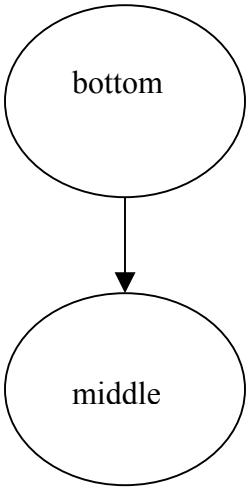
CS155 Computer Graphics
Lab 3

This lab is an exercise in modeling transforms and hierarchical coordinate systems. It also serves as an introduction to the ray-tracing project. Your objective is to build a model of a snowman and then render it using our sample ray-tracing program.

Your snowman should have a bottom, middle, and head. He should have two eyes, a nose, and a mouth. He should have arms. Beyond that you can improvise. Here is a scene graph representation of a basic snowman.



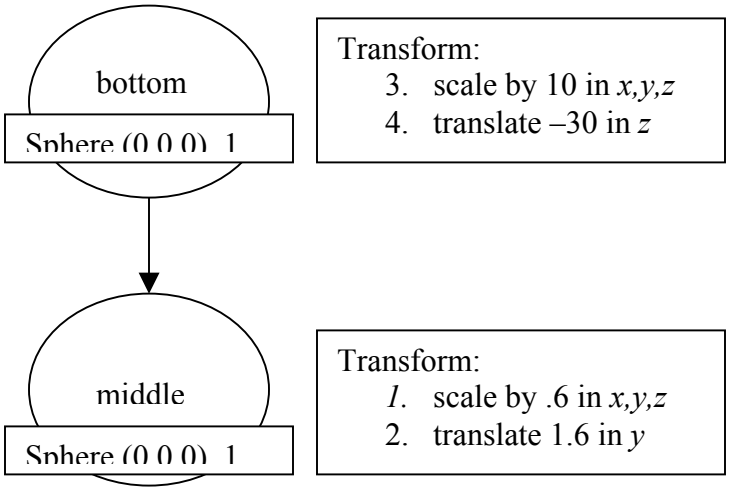
1. Download the sample ray-tracing program *rt*. Download the starter ray-file, snowMan1.ray. Run *rt* and right click to open the menu. Open the sample ray-file. The program will produce an image corresponding to the following scene graph.



The scene is rendered using OpenGL. To see the ray-traced scene, choose the render→raytrace option from the *rt* menu. The program will tell you when it is done but you shouldn't see any difference in the image.

2. Download the second starter file, snowMan2.ray, and open it in *rt*. You should see the same image produced in step 1. Open both ray files in a text editor to compare them. The common features are
   a. Background directive
   b. Ambient light directive
   c. Camera directive
   d. Material properties directive for snow

   The files differ in how they describe the snowman. Both use two spheres, but snowMan1.ray specifies the centers and radii of the spheres in world coordinates while snowMan2.ray uses modeling transforms and hierarchical coordinate systems. This description is depicted below.



The snowman's bottom is a unit sphere centered at the origin that is then scaled by 10 and translated –30 in the *z*-direction. The transform matrix shown for Group 1 in snowMan2.ray represents the product $M_T M_S$, where M_T is the translate matrix and M_S is the scale matrix. (Recall from lecture that transform-matrices are specified in reverse order.)

The transforms corresponding to a node are applied to the primitives of that node and all of its descendents. So the snowman's middle is defined relative to the un-transformed bottom. It is a unit sphere scaled by .6 and translated 1.6 (the sum of the two radii) in the *y*-direction.

3. To construct your snowman you should use spheres, cones, and cylinders as primitives. Decide which primitive you'll use for each body part and annotate the corresponding node in the basic snowman scene graph above.

4. Each primitive has size/position/orientation parameters or defaults. For details you should refer to the ray-file specification. You will use 1 for all size parameters and the origin for all position parameters. You'll then specify a composite matrix transform that scales, rotates, and/or translates the primitive relative to its ancestor in the scene graph. Annotate the scene graph nodes with the node transforms.

5. Extend snowMan2.ray to build your snowman and test it with *rt*. Each sub-tree of the scene graph should be encapsulated in a group. You will probably want to define additional material properties for the eyes, nose, arms, etc.

6. Email your ray file to me. Give me a paper copy of your annotated scene graph.