

Homework #6 - Robot Running with Lighting

README.txt: Be sure to include each piece of the README as required from project #2. If you need, refer to that pdf for a more complete list of required components. There are other questions

Be prepared to demo your robot on the due date. Your robot should have all of the following items.

1. Allow the user to start and stop the robot running motion through the use of a keyboard key.
2. Use the Idle function to implement a natural looking running motion for the robot. You should have 8 joints (two elbows, two shoulders, two knees and two hips). Examine running motion around you. You should see that while there are some symmetries, the motion is not perfectly symmetric, so you will want to handle all 8 rotations separately inside the idle function.
3. Move all objects to be Solid instead of wire. It will likely be neat to see both solid and wire frames, but only the solid mode is required for this assignment (glutSolidSphere or glutSolidCube).
4. Set material properties for your surfaces. In specific, make sure to make 4 function calls to glMaterialfv. The first argument should typically be GL_FRONT while the second argument is either GL_SPECULAR, GL_SHININESS, GL_DIFFUSE, and GL_AMBIENT. Write up which you choose for these 4 in your README and explain why.
5. You'll likely want to place these calls in the init function to help with the lighting. The variable model_amb is the global ambient color of the light. The local viewer forces OpenGL to calculate real vectors for some quantities for which it sometimes simply "hacks" a static direction.

```
glShadeModel(GL_SMOOTH); // colors determined at vertices
glLightModelfv(GL_LIGHT_MODEL_AMBIENT, model_amb);
glLightModeli(GL_LIGHT_MODEL_LOCAL_VIEWER, GL_TRUE);
glEnable(GL_NORMALIZE); //normalize all the normals
glEnable(GL_DEPTH_TEST); // turn on depth test
glEnable(GL_LIGHTING); //turn lighting on
glEnable(GL_LIGHT0); //turn on the first light
```

6. Set lighting properties for your light. In particular make sure to set the values in glLightfv. The first argument is GL_LIGHT0 and the second arguments are GL_POSITION, GL_AMBIENT, GL_DIFFUSE and GL_SPECULAR.

Remember that the position is affected by the current modelview matrix so to put the light in the right position you'll want it to undergo any matrix manipulations that would affect the positioning of other objects in your scene. Be sure to mention why you've chosen the values that you have chosen for your light in the README.

7. **Extra Credit:** I encourage you to experiment with other aspects of the lighting and materials. For example, try the attenuation and spotlight effects. Try several different types of materials instead of 1. In my implementation I have a function called `set_material(int)` that simply runs the material calls depending on which material was selected by the passed integer.