Project 4

CS 320

75 points, due April 19 at noon

Design a small maze, based on several rooms that fit into a 100×100 grid. (Rooms will be simpler to deal with than hallways.) (Optionally, you can think about generalizing the 2-D maze.) Your display should provide a small map view (with some indication of what direction the viewer is looking toward) and a first-person view. It would be "really cool" to be able to drag the overhead view around. An alternative would be to create a second window for this view. Put a few obstacles in each room as well as using two balls (spheres) which roll around the maze. Here are the project's requirements:

1. Your project's documentation should include a short user guide, explaining how to play the game.
2. Your collision detection should be "smart," as we'll discuss.
3. Minimize the number of "building" blocks used.
4. Movement:
   (a) Up arrow: forward one unit.
   (b) Down arrow: backward one unit.
   (c) Left arrow: rotate left 1/16th of a turn.
   (d) Right arrow: rotate right 1/16th of a turn.

   (To eliminate floating point round-off errors, store the rotation position as a small integer and convert that to radians as necessary.)

   You may modify the keys assigned to movement and add additional keys to follow gaming conventions. Make sure to document this in your user guide!
5. Include a "zombie" which slowly and weakly follows the viewer. If the zombie ever collides with the viewer, the "game" is over. The zombie should have the general shape of a human and has the power to pass through walls.

Submitting Your Project

We will follow this procedure for all projects this semester. Your project files are to be emailed to me at kelliher[at]goucher.edu. All source files necessary for building and running your program (C++ source(s), .h files, vertex shader, fragment shader, and pool data file) and any documentation files you've created should be sent as a single zip archive attachment in a single email. One documentation file is required — a README.txt file that explains how to play your game and describes what additional feature, or features, you added to your game. I will build your program from the source files provided and test the resulting program.