

Project 4

CS 320

75 points, due April 22, 2011

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 overhead view (with some indication of what direction the viewer is looking toward) and an immersed view. It would be “really cool” to be able to drag the overhead view around. An alternative (probably better) 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 some design parameters:

1. Modular design and literate documentation. Documentation should include a 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. Use display lists as much as possible.
4. Movement:
 - (a) Up arrow: forward one unit.
 - (b) Down arrow: backward one unit.
 - (c) Left arrow: rotate left $1/16$ th of a turn.
 - (d) Right arrow: rotate right $1/16$ th 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

Your solution is to be e-mailed to me at `kelliher[at]goucher.edu`. All project files should be sent as attachments in a single e-mail. You may collect all the files into a single ZIP archive, changing the file extension if necessary. You should send all files necessary for me to build your program from source (generally, this is all `.h` and `.c` files), as well as any data, documentation or test files. You should send an ASCII file, named `README.txt`, describing the rest of the attached files. I will build your program from source and run it for myself. Your project is due at the beginning of class on the due date.