



COMP 8051 – Advanced Games Architecture Assignment 3

Work in groups of two

Total marks: 100

1. Modify the maze program from assignment 2 to load and manipulate a model as follows:
 - a. [40 marks] Have the game load a model from a file at startup. It can be a standard file format like FBX or OBJ, or a custom file format. You must use VBOs and it should load the model file as an asset included in your XCode project. You may need to scale the model to have it appear in a single cell of your maze.
 - b. [20 marks] The user should also be able to double-tap the model and have it toggle between moving around and being stationary. If you are already using double-tap for some other function in the maze program, you can use double-tap with 3 fingers. When stationary and in the same cell in the maze as the user, the user should be allowed to rotate, zoom and pan, start, stop and restart the animation (if one exists), and move the model along any one of the 3 axes. Include in the README a description of all the user controls and any other information needed to play the game or understand your code.
 - c. [40 marks] When moving, the model should be displayed as an enemy object, which moves around randomly (no need for AI) in the maze. It should not leave the maze or be allowed to walk through walls. The model should not know where the walls are. Instead, the code should do collision detection to ensure it does not walk through walls. The model should also not be allowed to exit the maze (the code controlling the model's movement can know where the exit walls are to keep things simple).

The following are optional exercises if you want to practice incorporating text:

2. Using the sample code provided in class, write an iOS app that uses OpenGL to display 3D text.
 - a. Allow the user to zoom in/out.
 - b. Allow the user to pause and restart the auto-rotation.
 - c. Allow the user to change the speed of the auto-rotation interactively.
 - d. When auto-rotation is paused, allow the user to manually rotate about the y-axis.
 - e. Allow the user to manually rotate about the x-axis.
 - f. Allow the user to interactively move the light source using the mouse.
 - g. Allow the user to change the colour of the text.
3. Modify your maze program from the previous assignment to place a rotating text in the centre of the maze, above the user's view. That is, by looking up (using an appropriate touch gesture), the user should see some text rotating about the y-axis. The text should say "BCIT Games Development Option". You can set a constant, reasonable rotation speed.
4. Add to the maze program a HUD that uses the 2D text code and the freetype library as provided in the sample code. The HUD should show the current frame rate.

Submit your entire project, including documentation (at least a README file with any notes and a description of the user controls for each part) to the D2L dropbox. You are to work in groups of two and all will receive the same mark. Your submission should be in a single ZIP file using the naming convention A00ABC_A00DEF_Asst3.zip, where ABC and DEF are your student numbers.