# COMP 8051 – Advanced Games Architecture
# Assignment 2

## ***Work in groups of two or three***

Total marks: 120

In this assignment, you will demonstrate your ability to use lighting, OpenGL ES shaders and textures, and to render 3D objects with a moving camera.

1. [30 marks] Create an iOS app that draws a 3D maze using GLSL. The maze can be generated randomly or programmatically, or be loaded in as a resource. The walls and floor of the maze should have a texture. If you like, the code in COMP8051Asst2Start.zip contains C++ classes that generate a random maze. The default view should be at the entrance of the maze facing inside the maze as in an RPG game.

2. [30 marks] Apply textures to each wall and the ground. The same texture should be used for each floor tile. For each wall, one of four different textures should be applied, depending on whether there are no walls to either side, a wall to the left only, a wall to the right only, or walls on both sides.

3. [10 marks] Add an option to switch between "daytime" and "nighttime" modes, which changes the ambient and/or diffuse lighting.

4. [10 marks] Add the ability to move through the maze as in an RPG game using touch and drag. You do not need to worry about collision detection (i.e., the user can walk through walls). A double tap should reset to the initial default view at the start of the maze entrance.

5. [10 marks] Add the option to use a flashlight. You can either try using an OpenGL spotlight, or find another way to simulate a flashlight.

6. [10 marks] Place a rotating cube at the entrance of the maze, with the crate.jpg file as a texture to each side, making the cube look like a crate.

7. [5 marks] Add the option to turn fog on or off. The user should be able to select the fog function and the relevant parameters, either by specifying them as command line arguments to your program, or some other user interface.

8. [15 marks] If the user performs a double-tap with two fingers, a translucent console should appear, showing a 2-D map of the maze, and the user's current location and orientation. Another two-finger double-tap should hide the console.

The code must be written in Objective-C, Objective-C++ and C++, and all files required to build and deploy the app must be provided. 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 or three and all will receive the same mark. Your submission should be in a single ZIP file using the naming convention A00ABC_A00DEF_Asst2.zip, where ABC and DEF are your student numbers.