UNIVERSITY OF LINCOLN

School of Computer Science
Assessment Package Briefing Document

Title: CGP2012M Graphics Assessment 1	**Indicative Weighting: 50%**

Learning Outcomes:

On successful completion of this assessment package a student will have demonstrated competence in the following areas:

- [LO 1] Design and develop interactive 3D graphics software, applying appropriate mathematical/algorithmic techniques for efficient 2D and 3D graphics
- [LO 2] Demonstrate deep understanding of computer graphics programming techniques and approaches

For this assignment, you are required to:

1) Submit a PDF document outlining the features implemented as part of your application, as well as a reflection on the development process
2) Develop and produce the code for a C++/OpenGL application.

You should develop the graphics for a simple 2D game: **Breakout / Arkanoid**

The assignment aims to assess practical capability in writing programs that:

- generate 2D graphics
- update the 2D scene based on human input (i.e. interactive graphics)

The assignment will use the modern programmable graphics pipeline (i.e. OpenGL 3.1+) and will require a set of basic 2D graphics features, such as:

- 2D geometry
- moving/rotating objects
- coloured objects
- texturing

Specification

- You should implement a simple 2D version of Breakout
- Your implementation will need some very simple game-logic, but the collision detection doesn't need to be very good
    - but should let the game be generally playable
    - this could be using "bounding boxes" or some approach using the position of the centre of the bouncing ball and its radius
- The specific geometry (shapes) that you use are not defined - they are up to you

- Your scene should contain the following
    - 1 Player Character that moves under user control
    - Blocks which and don't overlap each other
    - Ball bounced upwards by the Player Character, that destroy blocks on contact
    - World boundaries
    - An indication of score

Summary

- The assignment will use the modern programmable graphics pipeline (i.e. OpenGL 3.1+)
- The assignment is a technical showcase formed as a (complete) game
- Your program should be useable with mouse and keyboard on a PC
- You can re-use code from your Games Programming Module, BUT your must do all the rendering with OpenGL
    - you CANNOT use SDL2's built in renderer.
    - using SDL2's built in renderer will **fail** this module

Submission

PDF document

- Submit to Blackboard
    - summary document of features implemented, with screenshots of each
    - reflection on the development process
    - at least four screenshots of your game running on PC

Code and application

- Submit to Blackboard, a single .zip file containing:
    - all your C++ source code
    - all your shader source code (if not in your C++ source)
    - files sufficient to create a working build environment (e.g. conan.io files, cmake files, or VS project)
    - all your assets (models, textures)
    - a compiled, running executable (for PC) (it doesn't need to be able to run outside the IDE)
    - a video of your program in action (for PC) (between 30 seconds and 1 minute long)
        - use OBS (https://obsproject.com), fraps (http://www.fraps.com), Screencast-O-Matic (https://screencast-o-matic.com), or other tool of your choice
        - the video should show all the features implemented

Languages / Toolkit (what you can use)

- You MUST use C/C++
- You MUST make the calls into OpenGL yourself
- You MUST use appropriate support libraries:
    - SDL2, SDL2_image, SDL2_ttf

    - GLEW
    - GLM
    - ASSIMP
- You CANNOT use ANY tutorial code from the web, even if you cite it - you must write the code
- You CANNOT use an existing object-oriented wrapper (e.g. oglplus)
    - but you can write and use your own if you like …
- You CANNOT use a graphics engine (e.g. Ogre3D, Irrlicht, Three.js)
- You CANNOT use a games engine (e.g. Unity)
- If in doubt - ASK

Submission Guidelines

Please zip up your project files (which should include an executable file, and your source files, along with any other
accompanying files) as a compressed ZIP file (**no RAR or any other file formats**) and should be submitted through
Blackboard in the '***CMP2090M Assessment Item 1 Supporting Documentation Upload***' section of the Assessments
folder.

The written report should be submitted separately on Blackboard to the '***CMP2090M Assessment Item 1 Upload'***
submission site

This module is graded using a criterion reference grid. You should be clear in your understanding of the grading principles; if you are not, please seek the advice of the module co-ordinator.

Hand In Instructions

See hand in schedule.
DO NOT include this briefing document with your submission.