COURSEWORK ASSIGNMENT, SEM2, 2017 **UNIVERSITY OF EAST ANGLIA**
School of Computing Sciences

MODULE: CMP-5010B: Graphics I
ASSIGNMENT TITLE: DESIGN AND IMPLEMENT A TWO DIMENSIONAL INTERACTIVE PLATFORM GAME USING C/C++ AND OPENGL

DATE SET	**:**	**Week 2 (Monday 11:00)**
DATE & TIME OF SUBMISSION	**:**	**Week 13 (report, demo)**
RETURN DATE	**:**	**Assessment Period Week 3**
ASSIGNMENT VALUE	**:**	**60%**
SET BY	**:**	**Dr. Rudy Lapeer**
CHECKED BY	**:**	**Dr. Stephen Laycock**

Aim:
To introduce the concepts of Computer Graphics.
To develop a program using OpenGL.
To gain experience in writing computer graphics applications, with the assignment focusing on a two dimensional platform game.
Learning outcomes:
To learn to write programs using OpenGL with C++ in Visual Studio.
To learn the concepts of 2D transformations, viewing and interactive control using OpenGL.
Assessment criteria:
Marks will be awarded for the following generic criteria:
1) Game Design (Objective, User Interface, Start screen, End screen).
2) Motion and control of the user controlled character(s) and/or objects.
3) Collision detection & Response between the objects/characters in the environment.
4) Graphical Quality of the environment (textures, sprites, background, …).
5) Novel Concepts.

Description of the assignment:

The **platform game** should enable the user to control (a) character(s) that move(s) on multiple platforms.

The following features are essential:

- The game starts at ground level;
- Has at least two levels above ground level;
- Each level has to be made up of at least two distinct (separated) platforms unless interlaced levels (zig-zag from left to right) are used;
- The target position of the platform has to be at the highest (or furthest) level;
- The character(s) can jump from one level to the next both horizontally and vertically;
- The character can collide with the platforms from all possible directions, i.e. it stays on the platform (obviously) but can also hit its head when jumping upwards;
- A clear objective and scoring system needs to be provided;
- The world should be larger than the screen size which means the background (world) should scroll both horizontally and vertically if the character(s) threaten to go off the screen out of the user's view.

The following features are optional (but desirable):

- The character is animated when walking or jumping (that is, use of different sprites/textures for different motions);
- NPC's (non-player characters) try to stop the player's character(s) from reaching its/their target;
- The character (s) can eliminate NPC's;
- Have character offspring which follows a main character (e.g. SPROGS example);
- Have moving (transition) platforms to travel from one fixed platform to another at the same (or even adjacent) level(s).
- Collision Response and physics
- Special effects, dynamic textures, particle systems
- More complex collision detection, e.g. GJK, BB hierarchies
- More advanced AI for NPC's
- Gravity
- More than one level
- A level editor
- Additional game information
- Any other clever ideas ☺

You may create windows using Win32, freeglut or GLFW or other suitable window managers (subject to approval).

Required:

A full submission should consist of:

1) Report detailing the design and implementation of the platform game and in particular outlining game-specific algorithms, methods, strategies, etc.	(max. 1,000 words)	(15%)
2) User Manual	(max. 1 side)	(5%)
3) Program + Demonstration, Week 13		(80%)

Handing in procedure:

Please hand in the report by 10/05/2017 before 15:00

A demo will be organised in Week 13 to present your game.

Please remember to print two copies of the mark sheet and give them to the markers during your demonstration. You can demonstrate on a lab computer or your own computer/laptop (PC not Apple!)

Please submit electronic copies of the executable, source code, textures and other supporting files (such as Visual Studio project files) by transferring it to my laptop during the demonstration **or** by submitting it on CD with the report.