Working with Game Engines

Coursework Guide

Marking Scheme

- **Scene 1 (36 marks)**
  - General scene management
- **Scene 2 (40 marks)**
  - Cameras and tools
- **Report (24 marks)**

Scene 1

- Loading
- Events
- Dynamic Mesh
- Inventory
- Sorting

Loading XML

6 Marks

- Load a mesh from XML
- Use XML structure supplied in labs
  - Copy structure from AssessmentChunk1.xml and AssessmentChunk2.xml
  - VoxelChunk.xml from lab exercises has the same structure
  - Full marks if UI interface allows file name entry

Events

8 Marks

- Possible events
- Block destroyed/Block placed
  - Attach audio cues
  - Audio dependent on block type
- Block Collected
  - Update Inventory
- UI Menu enabled/disabled
  - Disable first person controller

Dynamic Mesh

6 marks

- Use the texture supplied from the lab
- Allow block destruction from ray casting
- Allow block placement from ray casting
  - 4 block types to be placed
  - Need mechanism to switch between blocks

Inventory

8 marks

- Must include mechanism for creating dropped blocks of the 4 types
  - Blocks need to be collectable
  - Voxel code might be useful for this
  - Must store amount of each block type
  - Provide basic UI to view the inventory
    - Can be menu
    - Can be HUD

Sorting

8 marks

- Must implement merge sort
- 4 sorting criteria
  - By inventory item name (A to Z)
  - By inventory item name (Z to A)
  - By inventory item amount (ascending)
  - By inventory item amount (descending)
- Should use delegates methods for ordering

Scene 2

- 2D follow camera.
- A Dialogue Editor.
- Save Dialogue to XML.
- Load Dialogue from XML.
- Dialog system that triggers when the player gets to the non-player character.

2D Follow Camera

10 Marks

- Camera in the scene must follow the 2d player character in the premade scene.
  - Pick a method suitable considering the player characters movements and the level they exist in.
  - It should be pleasing to the player, worth researching this issue.
  - Camera should shake momentary when they land on the ground.
  - Camera should change it's focus (which character is speaking.) see tutorial 3.

2D Camera

- For full marks, suitable use of events and non-linear interpolation are expected.
- The game is a 2d platformer so research into 2D platformer cameras should be done.
  - http://www.gamasutra.com/blogs/ItayKeren/20150511/243083/Scroll_Back_The_Theory_and_Practice_of_Cameras_in_SideScrollers.php

Saving/Loading

14 Marks(8 + 6)

- Dialogue needs to be stored using XML.
- A suitable XML structure should be used.
- Students should think carefully on their chosen XML structure as it affects how they will read and write to it in C#.
- Note: don’t over use attributes, objects can be stored within objects so you might want to nest objects together if it’s suitable.
- Can reuse the same loading/saving code for the dialogue editor and dialogue system.

Dialogue System

6 Marks

- The dialogue that is loaded into the scene needs to played out when the player enters a trigger box near the NPC.
- It should take control away from the player character.
- Appropriate GUI should be created to allow players to see character’s dialogue responses and the player options.

Dialogue Editor

10 Marks

- Create an editor window(See lecture 6) that allows a developer to create and edit dialogue sequences.
- Dialogue should be saved in XML.
- Developer should be able specify the name of the file before saving and opening it.
- Test Data specified in the assignment sheet will be used as a test case.

Editor Utility

- A utility class that provides access to some useful static methods.
- `string EditorUtility.OpenFilePanel("Open Dialogue file (.xml)", "", "xml");`
  - Create a windows open file window and returns the path the user selected.
- `String EditorUtility.SaveFilePanel("Open Dialogue file (.xml)", "", "",  "xml").`
  - Create a windows save file window and returns the path the user selected.
- Parameters: title, directory, default name, extension.

Report

Scene 1 (8 marks)

- Overview of each script used to implement the scene and the purpose of each script. Highlight and explain important methods. Discuss purpose of member variables where appropriate.
- Explain communication between scripts, highlighting where events are used.
- Describe where software design patterns have been used
  - What pattern?
  - How has this been implemented?
  - What scripts are involved and what is their function?

Report

Scene 2 Part 1 (6 marks)

- Aim to describe what techniques and software design patterns you used to control the camera and make it follow the player.
- Review the PlayerController2D.cs and PlayerMovement2D.cs and evaluate the techniques used and if it was suitable or not.

Report

Scene 2 Part 2 (6 marks)

- Describe the structure of the XML file loaded
  - Show the XML structure with tags and use a DOM style diagram.
- Describe the dialogue editor, write it for a developer wanting to know how the editor works and how they should use it.

Report

- For full marks, The document should be correctly formatted with:
- Table of contents.
- Suitable choice of fonts.
- Suitable use of style.
- Suitable use of images.

Final Hand in

- Your zip file size should not exceed 400MB and MUST include the following:
  - The build data folder with the executable.
  - The project folder
  - A .docx or .pdf file report, name this:
    - <Last Name>_ <First Name> _WGE Report2019
- Marks will be deducted if any of these items are missing or incomplete.