### Home - UHI

**UI 110012 Dissertation (Computing)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Module title:** | **Dissertation (Computing)** | | |
| **Module code**: | **UI 110012** | **Word-count:** |  |
| **Deadline:** | 25th September 2024, 11:59 pm | **Date submitted:** |  |
| **Student number:** | 21010093 | | |

**This assessment is worth 10% of the total marks for this module**.

**Completing your assessment:**

This assessment should be word processed using a plain 12-point font, with one-and-a-half spacing and 1 inch margins. Submit your completed report, including this front cover sheet, via the VLE.

**Declaration of originality and authorisation to hold this assessment electronically and verify that it is original:**

UHI recognises that plagiarism, where deliberately engaged in, is unacceptable and is considered serious academic malpractice. **Students are responsible for ensuring the work they submit is their own**. If you have any queries you should contact your PAT or the Module Leader before submitting your assessment.

**By submitting this assessment I declare that the attached piece of work is my own**. I have acknowledged all the sources I have consulted and where I have used words which are not my own, I have clearly indicated this in the references.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Student number:** |  | **Date:** |  | |
| **Checklist** [**Note:** a checklist must be provided. The following questions are examples, you may use your own questions.] | | | | | **✓**[[1]](#footnote-1) |
| Answer explicitly addresses the question? | | | | | **✓** |
| Structure: introduction, middle section, conclusion? | | | | | **✓** |
| Citations in the text using the Harvard referencing system? | | | | | **✓** |
| All cited sources are listed alphabetically in full in the bibliography? | | | | | **✓** |
| All images are appropriately captioned and referenced? | | | | | **✓** |
| Images have been formatted to minimise file size? | | | | | **✓** |
| Word count is within 10% of the target length? | | | | | **✓** |
| Document has been spell checked and proof read? | | | | | **✓** |
| File saved as a Word (.docx) or rich text file (.rtf) with the filename format ‘Student number\_module initials\_assignment number’? | | | | | **✓** |

|  |  |
| --- | --- |
| **Declaration** | **✓**[[2]](#footnote-2) |
| In submitting this work, I confirm that I have read and understood UHI regulations and am aware of the possible penalties. | **✓** |
| I have completed this assigned work by myself, in my own words and using my own notes, figures or rough workings (except where group work specifically forms part of the assignment) | **✓** |
| I have acknowledged fully any sources used by means of in-text citations, and the creation of a List of References in the UHI approved system of Harvard referencing system | **✓** |
| I have endeavoured to ensure that my work has not been made available for copying by other students (with or without my permission) | **✓** |
| In submitting this work for assessment, I agree to be bound by the conditions laid out above, and by the latest version of the UHI academic regulations <https://www.uhi.ac.uk/en/about-uhi/governance/policies-and-regulations/regulations> | **✓** |

|  |  |
| --- | --- |
| **Originality checker (to be completed if Turnitin is used)** | **🗸** [[3]](#footnote-3) |
| I confirm that I received information about the use of Turnitin and was directed to Turnitin training | **✓** |
| I understand that this assignment will be submitted to Turnitin for originality checking | **✓** |

**For staff use only**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Lecturer contact number: | | | | | |
| Lecturer email: Graham.Wilson.Moray@uhi.ac.uk | | | | | |
| Feedback/feedforward return date | | | |  | |
| Feedback |  | | | | |
| Feedforward |  | | | | |
| Agreed mark - to be confirmed by Exam Board | |  | Final total / 100% | |  |

Assessment 1: Dissertation Proposal

|  |  |
| --- | --- |
| **Student name:** | Calum Lindsay |
| **Student number:** | 21010093 |
| **Project title:** | Where Have L-Systems Been Under-Utilized in 3D Games? |
| **Supervisor:** | Magnus Tullock |
| **Partner college:** | Shetland |

**Project summary/overview**

The goal of the project is to explore existing usage of L-Systems to generate content for 3D video games, propose and create solutions using L-Systems in areas where other techniques are traditionally used, and pioneer solutions using L-Systems in areas where little to no research has taken place.

Functional requirements:

A comprehensive review of the current literature must be performed to identify areas of greater and lesser exploration. Solutions to relatively unexplored problems must be proposed and developed. Solutions to well-known problems using other techniques must be developed using L-Systems. Experimentation must be performed to produce performance metrics of both the new solutions and existing counterparts. Created solutions must be critically analysed against existing solutions where appropriate. A 3D application must be created to demonstrate the created solutions.

Non-functional requirements:

Existing solutions using other techniques must be acquired or developed to perform performance measurements upon. An interface must be created to allow the tweaking of solution inputs for demonstration purposes. The performance of the created solutions must be reasonable for their intended use-case which may be real-time or offline depending on the specific solution.

Justification:

Lindenmayer Systems are self-rewriting string generation algorithms at heart and were first proposed as a way of modelling the development of plant like structures by Aristid Lindenmayer (https://www.sciencedirect.com/science/article/pii/0022519368900799) with later work expanding on this and importantly introducing the concept of a turtle renderer to produce rasterized images, culminating in a fascinating book: The Algorithmic Beauty of Plants (http://algorithmicbotany.org/papers/abop/abop.pdf).

Procedural generation in video games allows the fast creation of large amounts of content with smaller development teams which is important as over time “The most popular commercial games get larger, prettier, more atmospheric, and more detailed.” (https://dl.acm.org/doi/pdf/10.1145/2422956.2422957?casa\_token=\_uJTWI93H4QAAAAA:EV86R8uCLHtdWp8Te0I1We9TYMyGBx4jWuD3f-2Zteqz6DTaaaYhRFQWUY0x4NP2-5O5d0fJR1am3A : pg2) thus requiring more and more high quality content which is a challenge when “manual content production is already expensive and unscalable” (“).

Lindenmayer systems have seen limited use in games, mostly being used “for botanic modeling and generation” (<https://www.diva-portal.org/smash/get/diva2:832913/FULLTEXT01.pdf> : pg8) with famous examples that use SpeedTree being Battlefield 3, The Witcher 2, Gears of War 3(“), Elden Ring and Ghostwire: Tokyo (https://en.wikipedia.org/wiki/SpeedTree), however, their usage has been fairly limited in other areas such as terrain and scenario generation, and finding solutions in these unexplored areas could prove valuable if they improve current methods in an established area or if they open up an area that hasn’t been extensively investigated.

**Previous work**

Extensive literature exists for solutions that use L-Systems to produce plants and trees:

* The Algorithmic Beauty of Plants (“) outlines the
* combined genetic algorithms to evolve plants (<https://link.springer.com/chapter/10.1007/978-3-319-90418-4_2>)
* implemented dynamic animation of the growth of plants using L-Systems to produce key frames and interpolating between these (<https://nccastaff.bournemouth.ac.uk/jmacey/OldWeb/MastersProjects/MSc09/Hampshire/thesis.pdf>)
* produced L-Systems that simulate the death of a plant (<https://www.niclab.mx/portal/sites/default/files/SemanticDeathInPlantSimulationUsingLindenmayerSystems_0.pdf>)
* and generation of 3D plants using L-Systems (<https://www.scitepress.org/PublishedPapers/2009/17853/pdf/index.html>).

Additionally there has been some preliminary work in other areas:

* level generation (https://www.jait.us/uploadfile/2024/JAIT-V15N2-276.pdf)
* generation of cities (https://web.archive.org/web/20060114082225/http://www.vision.ee.ethz.ch/~pmueller/documents/procedural\_modeling\_of\_cities\_\_siggraph2001.pdf) (there’s another..)
* volumetric spaceship generation (<https://dl.acm.org/doi/pdf/10.1145/3520304.3528775?casa_token=7MOs98nNai0AAAAA:uSQgCc8x3ttin0TiPVI1fvd0pSMtNW2dfx58NTSt3GDJ9bWYb59Fse3slkaIeJgihCmeyD9tb-hsBQ>)
* and generation of river deltas (<https://core.ac.uk/download/pdf/322445609.pdf#page=132>).

**Technologies**

Using a game engine will significantly reduce the workload compared to writing all the rendering code from scratch which will allow more focus to be given to the main task of researching and developing L-System based algorithms.

Unreal engine 5 has been chosen due to its extensive use in the industry with Epic Games’ own titles such as Fortnite, Rocket League and Fall Guys as well as big hitters like Kingdom Hearts III (https://www.unrealengine.com/en-US/spotlights/unreal-engine-spotlight-kingdom-hearts-iii) and Hogwarts Legacy (https://www.unrealengine.com/en-US/developer-interviews/why-avalanche-worked-to-deliver-a-hogwarts-game-with-soul).

There are other options such as Unity and Godot. In this case it would not make much difference which of these options was used due to the student having no experience with any of them, however, they do have extensive experience with C++ which is what Unreal Engine is written in and a pre-existing desire to learn Unreal Engine.

C++ will be used to create the procedural generation code as it easily integrates with Unreal Engine and it is a high-performance low-level language well-suited to producing code that may run on a critical path such as during game environment initialization.

**Development methodology**

As this project is quite large, does not have a clearly defined destination, and the timescales of individual tasks are not reasonably predictable the simple waterfall method where timelines and tasks are planned then plotted out ahead of time doesn’t work particularly well. Instead an iterative and incremental development methodology will be used as these provide more flexibility and are “associated with many successful large projects, and recommended by standards boards” ( <https://www.craiglarman.com/wiki/downloads/misc/history-of-iterative-larman-and-basili-ieee-computer.pdf> pg 10).

A specific named methodology will not be used as these are typically designed for teams creating commercial software instead of lone developers without involved clients. The chosen methodology will use a Kanban board to track what tasks are pending, in progress and complete, and Scrum inspired sprints will be used where tasks are chosen each week based on a ranking of their estimated value. The pending column of the Kanban board will serve as this ranking and will be updated each week reconsidering the value of all the tasks in the list as well as uncompleted tasks from the previous sprint which may be moved to pending if their estimated value drops below other tasks. New tasks will be added to the pending column as they are formulated.

**Project plan/timeline**

A screenshot of a computer screen

Description automatically generated

**Supervisor meeting arrangements**

I will meet online with my supervisor every Wednesday afternoon at 1pm for 30 mins using Microsoft Teams.

**Ethics**

I have submitted a student application for ethical approval through the online portal. There are no ethical risks associated with my project.

**Reference list**

1. Copy and paste **✓**into each box to confirm that you have read and agree with the statements [↑](#footnote-ref-1)
2. Copy and paste **✓**into each box to confirm that you have read and agree with the statements [↑](#footnote-ref-2)
3. Copy and paste **✓**into each box to confirm that you have read and agree with the statements [↑](#footnote-ref-3)