ASSESSMENT AND INTERNAL VERIFICATION FRONT SHEET (Individual Criteria)

Course Title	**Advanced Diploma**			**Lecturer Name & Surname**	**NEIL AQUILINA**	
Unit Number & Title	**Programming for Computer Games**					
Assignment Number, Title / Type	**Research and Design – Home (24 Hours)**					
Date Set	18/12/2020		**Deadline Date**	**19/12/2020**		
Student Name	Ishmael Galea		**ID Number**	**276501L**	**Class / Group**	**4.2B**

X	*Student's declaration prior to handing-in of assignment:* ✝ *I certify that the work submitted for this assignment is my own and that I have read and understood the respective Plagiarism Policy*	
☐ ☐	***Student's declaration on assessment special arrangements (Tick only if applicable)*** ✝ *I certify that adequate support was given to me during the assignment through the Institute and/or the Inclusive Education Unit.* ✝ *I declare that I refused the special support offered by the Institute.*	
Student Signature:	Ishmael Galea	**Date :** **18/12/2020**

Assessment Criteria	Maximum Mark	Mark Achieved
KU1: Identify and describe different game engines for different tasks	5	
KU3: Describe file types for media assets	5	
KU4: State the relevance of compression settings in media assets	5	
SE1: Design and specify the details of the game to be developed, including a state machine	10	
Total Mark	25	

Assessor's feedback to student

(If necessary, use reverse side of page for IV feedback on assignment brief / sample of assessment decisions)

	Name & Surname	Signature	Date
Internal Verifier : Approval of a_ssignment brief_		For approval signature, please refer to electronic audit trail	
Lecturer / Assessor : Issue of results and feedback to student		For approval signature, please refer to electronic audit trail	
Internal Verifier : Approval of _assessment decisions (Sample)_		For approval signature, please refer to electronic audit trail	

Learner's signature upon collection of corrected assignment.		

Assessment Criteria
KU1: Identify and describe different game engines for different tasks
KU3: Describe file types for media assets
KU4: State the relevance of compression settings in media assets
SE1: Design and specify the details of the game to be developed, including a state machine

PCG Home Assignment 1

Ishmael Galea- SWD4.2B – 276501L

Task 1:

- Unity

- o Language: C-sharp

    - o Game: Cuphead

    - o Dimension: Both

- Unreal Engine

    - o Language: C++

    - o Game: Tom Clancy Splinter Cell

    - o Dimension: Both

- Frostbite

    - o Language: C++

    - o Game: Battlefield 1

    - o Dimension: Both

- Pygame

    - o Language: Python

    - o Game: Frets on Fire

    - o Dimension: 2D

- Source 2

    - o Language: C++

    - o Game: Dota

    - o Dimension: Both

Task 2a:

- JPG – A JPG image file is the most common used for any photos taken by a digital camera since this format allows the image to be compressed without losing much detail.

- GIF – A GIF image is very compressed but looses no quality. In a single file it holds and display a sequence of images that generate and animation.

- BMP – A BMP is used to store a bitmap. This is a 2d image that could be monochrome or coloured.
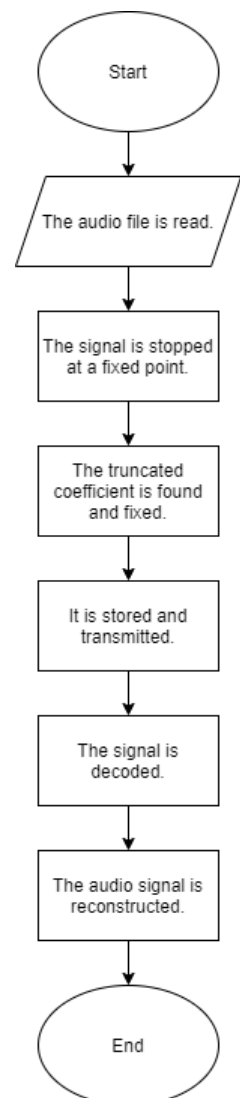
Task2b:

- MP3 – This is one of the most common audio file format since it keeps the audio quality similar to the original but take less storage space.

- WAV – This file format is more used to save raw and uncompressed audio.

Task 3a:

Image compression is when you decrease the size of the file without loosing the quality of the image to an unacceptable level. This will allow for more images to be stored in a small amount of disk space. The most common file type for internet use is the JPG and GIF formats. The JPG is used for photographs or images, while GIF is used for line art or animations. When compressing an image there are two types which are lossy or lossless. Lossy compression is used when it is not that important if the image losses minor details. While lossless is when no detail can be lost and the entire images has to be kept like the original.

Task3b:

The audio player starts by reading the audio file, then the player will stop the signal at a fixed point. After the player will find the truncated coefficient below the threshold, this will get fixed by encoding it and storing the new signal to be able to transmit. Once it is transmitted, it starts to decode the signal which is then reconstructed and would be able to be heard.

Start

The audio file is read.

The signal is stopped at a fixed point.

The truncated coefficient is found and fixed.

It is stored and transmitted.

The signal is decoded.

The audio signal is reconstructed.

End

Task 4:

Room

Walk to table → Table

Walk to chair → Chair

Walk to door → Locked Door

Walk to PC → PC

Find Nothing

Has Screwdriver → Free

No screwdriver

Find Screwdriver