Programming for computer games

Question 1

A:

There are multiple game engines being used to produce games in the world. Two of which are **Unreal Engine** and **Unity**. These have various advantages when it comes to both of them. Although both are excellent at creating games, **Unity** was chosen.

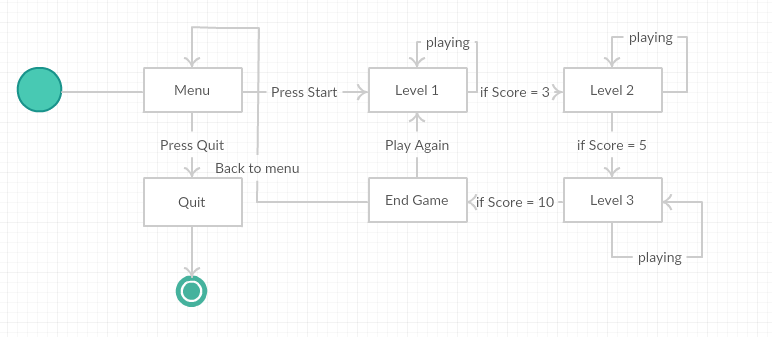
* This is due to the fact that Unity has great 2D features (pluralsight, n.d.) Hundreds of thousands of creators rely on Unity for regular updates with the latest and best tools for developing, launching and earning revenue with 2D games. (tech, 2017)
* The fact that Unreal Engine uses C++ whilst Unity uses C#. In this case Unity was better as C# was found more useful as it has a bigger standard library and along with the fact that you can set classes, methods and fields to be assembly-internal (Marcovic, 2013)
* Both Unity and Unreal Engine have the ability to allow the developer the use of an asset store. However, Unity has a much larger asset store in size than Unreal Engine, making it the better choice when it comes to the benefits that a game engine gives regarding the amount and types of assets. (pluralsight, n.d.)
* Unity as a game engine is more popular to regular people creating simple 2D games. With this comes the benefit that there are much more tutorials and community input thus making the creation of a 2D game on Unity easier for the developer.

B:

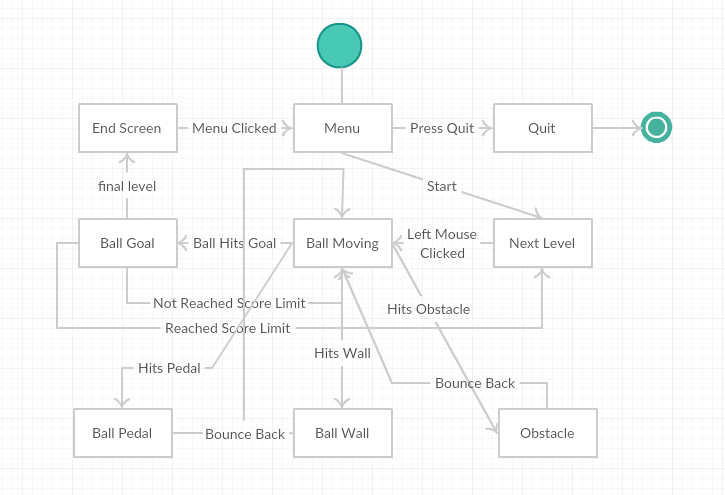
There are multiple programming languages used in game development. Two of which are C# and java. C# was chosen to support the game engine. Four of its features are, it’s deeply integrated with Windows (as the game made will be featured on Windows). It has dynamic variables along with the fact that it allows you to define new value (or non-reference) types and finally that it doesn’t have checked exceptions, in this case it is good as the code is not very complex and will not be important. (Marcovic, 2013)

Question 2

A:



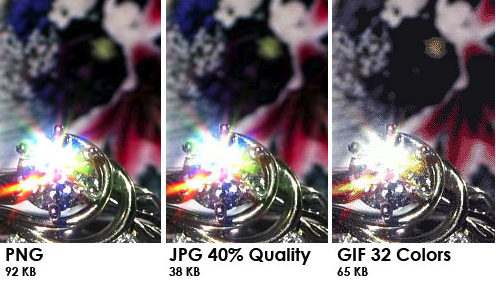
B:



Question 3:

When it comes to Compression when using media assets such as images, videos and audio, it is most present in mobile devices or for app stores that impose a size limit. The best way to reduce the size of a file or app is to determine which assets contribute most to storage build up, as these assets are most likely candidates for optimization. It is suggested when compressing, that the main assets to be compressed are textures, meshes and animations and DLL’s. Textures usually take up most of the space, the best way to compress them is to use compressed texture formats. Mesh compression only produces smaller data files, and does not use less memory at run time. On the other hand animation keyframe reduction both produces smaller data files and uses less memory at run time. All this will improve the performance in the end as well as having a smaller file in size. (tech, 2017)

The Image below shows the difference in quality when it comes to compression.



# Bibliography

Marcovic, Y., 2013. *StackExchange.* [Online]   
Available at: https://softwareengineering.stackexchange.com/questions/125712/for-what-reasons-should-i-choose-c-over-java-and-c  
[Accessed 29 november 2017].

pluralsight, n.d. *pluralsight.* [Online]   
Available at: https://www.pluralsight.com/blog/film-games/unreal-engine-4-vs-unity-game-engine-best  
[Accessed 29 november 2017].

tech, u., 2017. *unity documentation.* [Online]   
Available at: https://docs.unity3d.com/Manual/ReducingFilesize.html  
[Accessed 29 november 2017].

tech, u., 2017. *unity3d.* [Online]   
Available at: https://unity3d.com/solutions/2d  
[Accessed 29 november 2017].