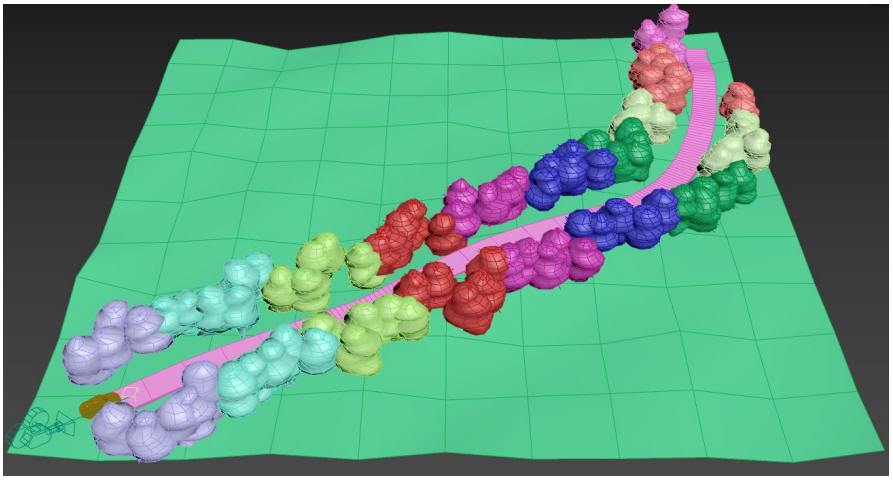
Tools Scripting 2019

Final Work
Content: Maxscript & C++ Tools

Exercises

Part1

1. **Create a maxscript code that: [5 Points]**



   a. **Generate a random terrain (plane) of size nxn given by the user. The plane generated must have irregularities that resemble to an organic shape.**

   b. **Generate a pseudo random road along the terrain previously generated that starts in one of the terrain corners and ends in the opposite one. Road shape must be irregular and the road must stick to the terrain at any given point along it (no floating parts).**

   c. **Generate random trees along the road. The trees MUST NOT touch each other and MUST NOT intersect the road.**

   d. **Create a car animation. The car must be animated travelling along the previous road (spline) created.**

   e. **Create a camera and attach it to the car, so that an animation clip can be recorded from the previous car animation created.**

Part2

1) **Design your own level (Whitebox) and import it into the engine: [2.5 Points]**

   a) **Design and create a single whitebox map by providing an sketch of the map and the .max scene with the content. Whitebox must be block based.**

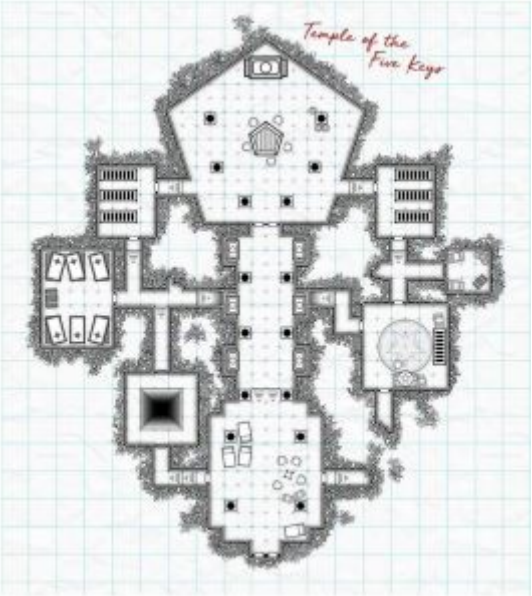


Fig 1.1: whitebox layout map from fps game

   b) **Export the scene from 3DSMax to the engine using the tools created at class.**

   Note 1: Student can use map designs from google or create its own. Max scene must be of his own.
   Note 2: Follow the principles from the given links

   https://www.gamasutra.com/view/feature/131736/beginning_level_design_part_1.php

2) **Create a new component (movable platform e.g). [2 Points]**

   a) **The student should be able to attach a component into an object from a 3dsmax scene, export this object with its component attached to it (must be linked in the json).**

   b) **The student should be able to read this component and load it into the engine.**

c) **The component must appear in the UI to be edited. Display in debugrender method.**

d) **Implement add new component button in the inspector window.**

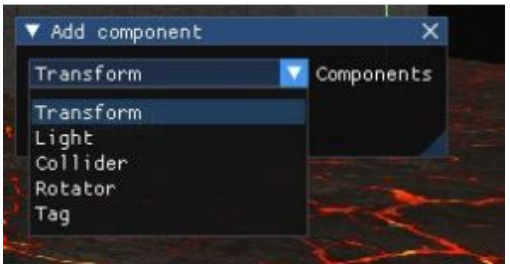


Fig 1.2: Add component menu as seen in the custom engine

3) **The user should be able to execute scripts from the console by introducing commands (move player to position, spawn item..)** **[0.5 Points]**

a) **Create atleast one new command and describe its functionality in the readme.**

Fig 1.3: moveplayer command executed on the console

Note 1: Implementation on BuildCommand method at ConsoleModule class.
Note 2: Following links might give an idea of the different commands that games use in their scripting system to allow the developer and the player to interact with the engine.
https://nwnlexicon.com/index.php?title=Category:Beginning_Scripting
https://scripts.zeroy.com/

4) **Extra functionality:** **[Bonus 1.5 Points]**

a) **Add a button to delete entity. The user should be able to delete the current selected object from the scene by pressing a button on the inspector or pressing the “del” button.**

b) **Select an object by ray intersecting on scene view click. The user should be able to select an object by simply clicking on it in the scene view.**

Submission

- **Limit date [14/01/2020 – 23:59:59]**
- **The student must provide a .zip file (or github repo) with the engine and tools files required by the app. Readme and map design image must be included.**
- **Content must be sent to alberto.s@salle.url.edu before limit date.**
- **Mail header must be [MVDTools-MyFullName]**
- **Work must be done individually by each student.**
- **Provide a readme (.txt file) with the description of the work done.**
- **The plugins must execute without errors.**
- **The engine must execute without errors.**