Team Members:
Mark Kouris
Will McDonald

Research Topic and Goal:
We will be researching Dungeon Generation with variable parameters. With this, we hope to achieve the possibility of an endlessly replayable game for our player without the need for design capabilities. With our varying parameters, we also want to have the experience stay fresh even with numerous repetitions.

What our Demo will consist of:
We will develop a prototype using C# in the Unity engine. The Demo will provide tools for a user to produce a “dungeon”. This dungeon will be represented by a simple grid view of “rooms”, which the end user would use to define the terrain of their world. The user will be able to customize the general attributes of the dungeons created with sliders. We expect that by the end of our project, we will have a functional prototype with a player character simulating a more complete gameplay scenario, traversing the generated dungeons

Potential configuration sliders:
- Dimensions
- “Sparseness”
- Clustering
- Number of misleading paths

Why it’s worthwhile:
Many games use Procedural Content Generation to increase replayability, we value this and would like to simplify its implementation. Building a standalone library to handle the configuration of the shape and feel of dungeons generated will reduce the barrier to entry for other developers. This would have uses inside of tools such as RPG Maker, giving hobbyist game developers access to a new suite of tools. With our tools hobbyist game developers will be able to expand the scope of projects they can produce.

Procedural Content is used extensively in the roguelike and ARPG genre.
- Binding of isaac
- Deeprock Galactic
- Enter the gungeon
- Diablo
- Path of Exile
- Torchlight