Requirements

To create this game I must:

- Have a defined goal for winning the game
- The game must have at least 10 rooms with 4 pointers to other rooms or null
- There must be some sort of time limit. The user can't play forever
- There must be different types of rooms
- To win there the user must complete some sort of action. The player can't just walk through rooms and win.

Design

I designed the story after one of my favorite games that fit the requirements given; Dead Space. You will play as a character just awoken on a spaceship that has been boarded by alien; your crew has abandoned you and the core is melting down. To win you must find three keys to deactivate the core. You find the keys as you work your way through the ship fighting creatures along the way.

The implementation will be mostly be made up of parts of previous assignments. The rooms will be a string of node objects, all inheriting their characteristics and behaviors from a parent room. Different types of rooms will require different player activities to move on from the room. To understand what to do in the game the player will be prompt with a cout message describing what their available options are. They might have to fight a creature or have to search for a key, or simply move on. Players will navigate the rooms by choosing between one of the four cardinal directions n, e, s, w; or choose "i" to inspect.

To keep track of game details I will make a world class. This class will determine how many turns the player has left and whether the player has won or not.

Testing

To test the program I just ran through the game and fixed problems I came across. Things like making sure creatures die, and that collecting keys actually makes you win the game. I also made sure the game stopped when the number of turns ran out. I also made it so once you find a key you don't have to search for it again.

Reflection

This was a fun and motivating assignment because it showed me just how big the possibilities are. It also showed me the value of refactoring and creating reusable code. This assignment immediately reminded me of so many video games I've played before. The protagonist starts alone and must struggle through a never ending chain of hallways and rooms until the end. This made it easy to visualize how rooms could be nodes chained together.

Looking back it seems like I’ve made enormous progress since the beginning of the quarter, but now I feel I’ve just begun to scratch the surface of what’s possible. I’m excited to see what next quarter brings!