Final Project Reflection

For this final project, I will adapt some features from the previous projects to make a fantasy combat game. I will create an Adventurer from the Character class and then the creatures that the adventurer must face to be derived from there. The space class will have several derived classes that will be the spaces that hold the creatures. For the time limit it will be the player's health and they will have access to a certain number of health potions. There will be a backpack to hold the items. The spaces will be set up in a linear fashion so the player will move forward from one space to another. Since its linear the up and down pointers will just be set to null the whole time. I will keep a current Space Space pointer to track where the player is.

Menu at the beginning of each round

Input	Expected output	Actual output
A non-integer is entered	Loops back for user input and saves it	Displays an error message and asks for input again
Integer != 1 or 2	Loops back for user input and saves it	Displays an error message and asks for input again
Input is 1	The player attacks the monster and the monster attacks the player before looping again	Outputs the current health of the player and monster after both attacks
Input 2	The Player takes a health potion reviving their health back to 100	Displays that a potion has been used and the current player's health

Changes/Issues: The first issue I came up with was with the backpack. I initially created the backpack as an stl queue container to then contain the potions and the keys each time a creature was defeated. However there needed to be a container limit on the number of items to be added. I then decided to change it to an stl deque container so that I could use an algorithm to check the number of potions in the container and to restrict the number past a certain point. The other tweaking came from adjusting the Player and creature's health and attack capabilities to prevent the game from taking too long. Initially I had thought to run the game function within the derived classes of Space but then realized it would be more logical to run the game logic within the Space parent class and have it then inherit to the child classes. I had a few vtable errors due to the virtual functions that needed to be changed. Along with changing how the health potion worked so that it would not have the player's health exceed 100 at the same time.