Jennifer Frase

Requirements:

- Make a text based game where you have a goal
    - Allow the user to carry items
    - 10 rooms, with at least 3 types
    - The user must be timed in some way

Design:

Room class

Needs to hold four pointers

The name of the room

The name of the four walls in the rooms

Bedroom

Only one way in and out- so all pointer are null except one

The other three walls are desk, bed, and closet

If the user chooses those wall, if there is something hidden there, they will find it

If they find and object, they can choose to carry it

Guarded room

Will hear voices if they approach

If they enter without a specific item, they will be caught

Hall Ways

Basic room but with all four sides being usable (not that they are, just can be)

Elevator only has three sides

Food room

Has table where items can be put and collected from, as well make sandwiches

Has relevant items

Kitchen has ingredients

Mess hall has dinnerware

Pack

What I want is to have an item, with a numerical value, and the items can be stored in pockets. This means that large items have to be carried.

Jennifer Frase

Main

create the linked structures, and the deconstructor.

Count each time that the user enters a new room and limit this as a form of time restraint

Run through the game, calling on the appropriate direction based on user input

Testing

See testing doc.

Reflections

My hardest part of this assignment was the designing. Not the program design, but the concept design. That has always been my weak point. And this week proved it, since I spent more than a week just coming up with the above that was literally as far as I could go only the design without sitting down and coding some. Then when I was coding it, I found where all my concepts were missing chunks. For example, I originally thought about using an array for the bag, but that would make it difficult/impossible to implement my weighting system. So then I thought about maps, since they have keys. However, this has the problem of being sorted, so I can't have them be randomized, like you would find in pockets. Your pockets are not alphabetized.

I thought maybe a struct, but where would I put it. All the classes and the main have to be able to see it, and creating a header just to store the pack seemed excessive. But I eventually gave up and went that route anyway. Then because of my weighting system and my pockets, I had to create special add functionality, since the tables are fairly straight forward. However, on that note, when I was debugging, I had the hardest time copying items in my pockets to the tables. It just wouldn't copy. I eventually figured out I had to overload the operator, but even then I kept getting errors. Which really isn't surprising since I hadn't done it before.

On top of the lvalue errors from the copying, I kept getting vtable errors. After about an hour I where I looked all over the place, I finally figured out the error wasn't in the header file, but the source file (and it wasn't even with the function it said it was). Once, I got that working, I could finally start working on all my grammatical mistakes, which sometime resulted in bugs. And then sometimes, I forgot to fix my debugging, so that caused errors. So all in all, I think I spent about a week of 4 hours a day working on the concept (yes that is how long it took me – partly because I wasn't sure what constituted as a different room) and then about 24 hours doing everything else.

I am pretty sure I will lose points based on my design, but that is literally what I could do without physically coding. Normally I can kind of see what I need to do, but this project was tough on me. I kept thinking of things and then going, no that won't work or how am I going to do that. The final product is actually very watered down compared to what I wanted to do. It's probably got about a quarter of the functionality I wanted it to have.