COS 225 Final Project
60 Points Total

For this final project, due at the time of the final, you'll be creating a large program that uses the listed aspects of class. You're allowed up to 3 members per group but what you're expected to turn in will increase.

Your project needs to have;

- File Input and output
- Using some form of data structure that involves objects using references to the same class of object (linked lists, BST, etc).
  OR
  An advanced sorting method (merge or quick) used on an arraylist/array of objects.
- Overloading Functions
- Command-line arguments and/or user input through the console.
- Be separated into multiple files, classes, and at least one namespace of your own.
- Error catching, both through try-catch blocks and through if statements. (basically QA test your project).

Each aspect needs to be included in a meaningful way and contributes to the overall goal of the program.

When commenting, in addition to comments at the top of each file you're required to comment on each class and each function. A brief description of the purpose of each and what type of data sent/returned (in the case of functions).

Before the due date of the final project you need to turn in a proposal for your project briefly explaining what your project is going to be and how it touches on each aspect, what additional things you're going to need to learn to accomplish your goal, your group members, and if you're delegating the work, who is responsible for what.

Safe project ideas to begin from (and a quick description on how to approach some aspects to fulfill requirements):

Dungeon Crawler:
A text-based game where the user navigates a “dungeon”, interacting with objects to accomplish a goal such as reaching a treasure or escaping.

Classes include Rooms that are linked to each other via references. The layout of the rooms, their contents, and the monsters can be constructed from a text file, the name of which can be provided via command line. For the output, a score based on how well the player performed before dying or succeeding.

Virtual Bot:
A program that takes in input via file and attempts to perform a series of actions in a virtual world based on those actions.

The world is just a simple 2d array, the robot (an object that contains several properties such as location (x and y) and orientation), and a box to grab. In the text file, a series of commands will be issued to the robot in the order provided out of a list of commands (move forward, move backward, turn left, turn right, grab). When the program starts, read in the file creating an action object for every command and adding it to a linked list. The world layout itself will also be provided via file. Once all the commands have finished running, output the new layout of the world.