Surprisingly, I did not encounter any issues with pointers. I believe that is because I had had such a hard time with the operator overloading assignment, I was able to get through this assignment with ease. It was because of the headaches I had gone through in previous assignments that I took the time to learn pointers properly.

One way I would avoid pointer errors is to only use them when necessary. I am not saying never use them, but if a programmer can use a reference or a copy to accomplish the task with a minimal tradeoff, use it to avoid any pointer problems. The other is to never perform pointer arithmetic. Simply treat the pointer as something that cannot be changed and a programmer will avoid sever headaches leading to segmentation faults, memory corruption, access violations, etc.

The next assignment will require more planning than usual. The best idea is to have each item a part of its own class, with some basic functionality for accessing item in the class. A hash set maybe needed to store the unique items on the list, and a few counters for the receives/requests of different items. The design should not take more than 3-5 hours. The implementation may take 15-20 hours, mostly dependent on simple compile and syntax errors. The major task will be the reading of the input file, and ensuring the lines are properly processed. The classes representing each line will not take too much time at all. With a partner we may do the reading of the input file together, but the classes on our own. By working on separate items of the project, and having defined the classes well, my partner can work on one side of the project without worry about how I am implementing the classes because I will follow the contract. Merging the files will simply be a matter of combining each class.