Final Project Report – Quinn Nagel

Data Structures – CIS152  
4/30/2024

Dr. Michelle E Ruse

# Proposal

We all struggle with making lists and having to edit them repeatedly. Sometimes we make them on notepads and find it hard to keep track of the items we have put on there. My project is a Building Assistance System, built mostly for a game called Minecraft. I often found myself making these long check lists of items I needed, so I decided why not make a program to keep track of it. This project allows you to add items to your list, append an item, and then see how much you have out of that item. You may also remove and deduct items from your total.

The most important part of the program is the class system. Using two unique data structures we have a custom data object called “Item” this item holds data information for every item in the list. Finally, we use a linked list in a program that is put through another class called BaseBuilderHelper. This class allows you to add items, sort the items, increment, and decrement the items as well. The Classes are housed in their own separate Python file from the Main to organize everything and provide some form of formatting to the project. The program also uses a single sorting function that uses bubble sort to sort the items when displayed from Greatest to Least, top to bottom. This function is also stored in a sperate file, as well as another file that is used to Unit Test the whole program using functions, but since they aren’t related to the program they are stored in a separate file. Finally, we have the main file which has the GUI that has some custom coloring to provide some visual flare.

# Time / Change Logs

During our first week of the final project, we of course proposed the idea of the final. This took some time for me to come up with. The simple idea would be a system to sort books or students. First that wasn’t allowed and second, I wanted to be unique and do something that I would use or have an interest in coding. So, I chose this Base Building Helper program that I could use for a game that I play. I play a lot of factory games and such so being able to have a way to keep track of and sort items is very important.

Second Week, I decided to make a flowchart to organize my ideas. This helps me be able to organize myself and keep myself on track not getting lost in about four hundred lines of code like my previous final project. Third Week, I began working on some of the functions in the final, I didn’t do much considering I have other finals at the time as well, but I did get many of the functions solved during this time. Fourth Week, I began working on the GUI of the Main code, this would allow me to interact with the functions and be able to test them and see if they even work as I wanted. Fifth Week, I made the UI nice and pretty but was mostly occupied with life, so I couldn’t get a majority of the project done unfortunately. Finally on the Sixth Week I finished the project and got the basic idea of the code done, now I would test it and fix any errors. Finals week, I made sure my project was following the requirements and would turn it in.

# Lessons Learned

The biggest lesson I learned was making sure I managed my time correctly. During my final I got sick many times and was also tasked with watching my dogs at home. This would of course take time away from me working on this final. I did get it done in the end, but it made the process more stressful than it had to be. Another thing I learned was probably making sure it complied with requirements before finishing the whole project, its hard to alter it after than during the process. Other than that, the project went smoothly.

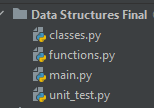
# GITHUB REPO LINK

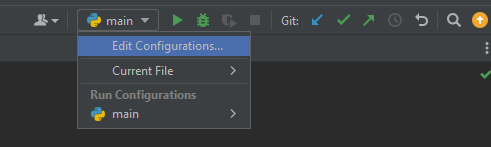
* <https://github.com/HSTVEnjoyer/Data_Structures_Final-CIS152>

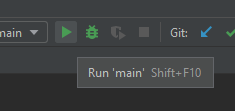
# User’s Manual

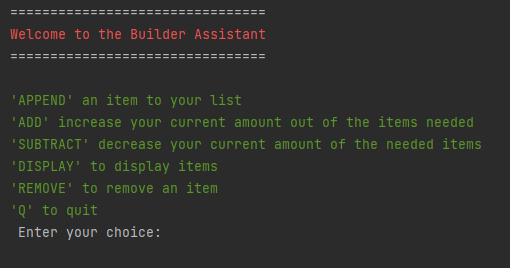
Using the Program is very simple, you must first import the program from GitHub into an IDE. I will use PyCharm as an example.

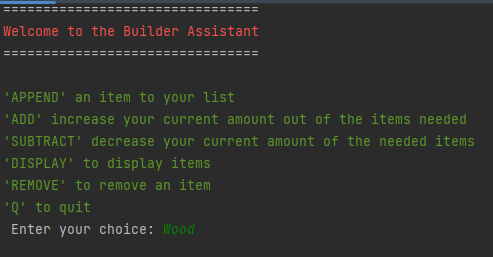
* First step is to make sure you have watched and followed the guide from this video by Simple Python - <https://www.youtube.com/watch?v=T9eySn3ORhs> This guide shows how to import the project into PyCharm.

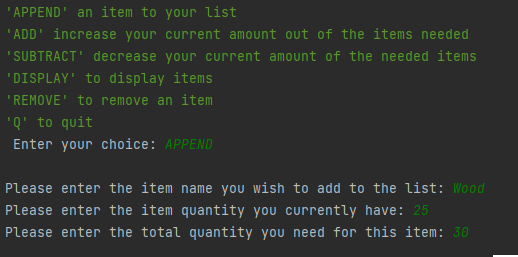
After importing into PyCharm you should see a folder called Data Structures Final. Inside this folder is four python files. One called main.py, another called functions.py, another called classes.py, and finally unit\_test.py. You need to run Main, so click on main and then open the file.

This should pull up the main file and you should be able to see the code. Now on the top right side of the PyCharm you should see the python symbol with a dropdown arrow make sure this is set to either Current Program or the Main file.

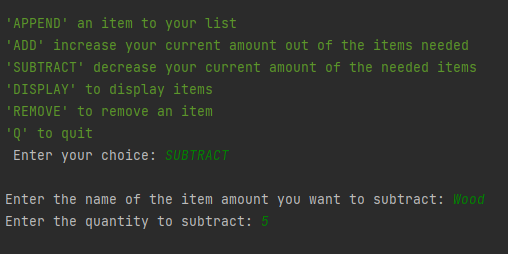
Then you can press the green arrow to run.

Now you should see something like the image below, if not read through the steps again and make sure you didn’t miss anything. Now seeing the GUI everything is pretty explanatory, I will give you an example using a basic item called Wood.

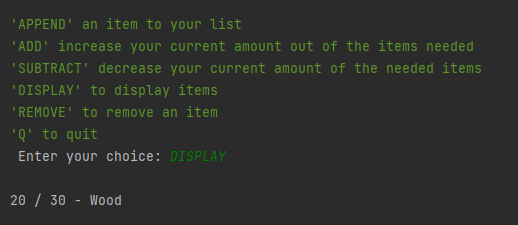
Oops I put in Wood and didn’t read the green text, luckily the program caught me and told me it was an invalid input, now let’s input ‘APPEND’ to add and item to the list.

Now I entered after typing just APPEND with no quotes around it, it takes me to the option to enter in the item name, I just put Wood. After that it asks me for the item quantity I currently have I said twenty-five, and the amount I need for the item which I said is thirty.

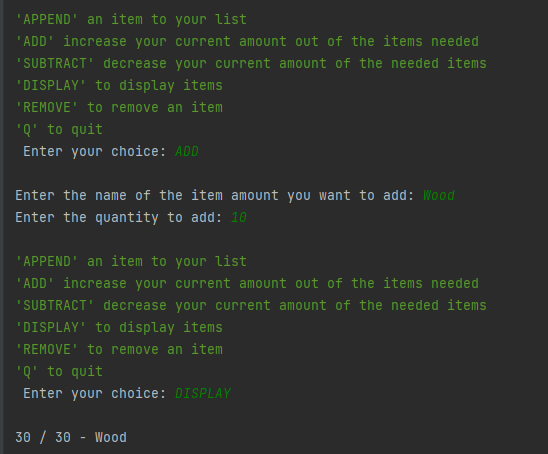
Let’s display the item by entering in ‘DISPLAY’ again with no quotes around it. You can see it says I have 25 / 30 Wood, which is exactly what I said.

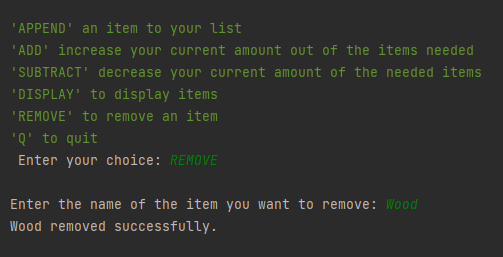
Now I want to subtract some of the wood, let’s say I lost five Wood. I can input ‘SUBTRACT and then subtract five wood from the item called Wood. So, I input the name of the item I want to subtract from which is Wood, and then input the quantity which is five. Now you can see that after using SUBTRACT I have lost five wood from the list.

You can see below it subtracted five wood from the previous amount.



Let’s say I want to add the rest of the wood, I would just type ADD, and then do the same as subtract. Type the name and followed by the amount of the item I want to add. For me it is 10, you can’t go above the base limit for the program and cannot go below zero.

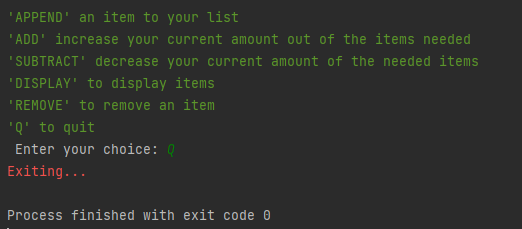


Finally let’s say you want to remove the item entirely from the list, you simply type REMOVE, and the name of the item. Then it’s gone from the list.

You can see below the item is gone.

A screenshot of a computer

Description automatically generated

Finally, if you wish to quit the program. Simply input Q and it will quit.

# Conclusion/Summary

MERUSE (Dr. Ruse’s principles of good programming) describe good programming as something that is modular, efficient, readable, usable, stable, and elegant. Currently in my program I have many of these. My program is modular, using multiple files to sort everything, and add modularity to my program allowing for further expansion. My program is efficient using the least number of processes needed to obtain the end goal. My program is readable, both inside the code and in the GUI. I have comments littered through the code allowing anyone to understand what it is doing. I even tested this with some friends of mine, and outside everything is properly described. My program is easily usable, needing only a surface level understanding of how to set up programs to use it. It’s stable and elegant, nothing is crashing and the program itself looks pleasant to look at.

In summary, my project is a builder assistance for any game or project in real life you might have. Allowing the user to keep track of items and remove items. Add and subtract items from the list and then simply just quit when needed. It’s a simple program for a simple purpose but at the same time its purpose is good for being so simple. The program is easy to use and read, even beginner programmers and people who have no experience have tested the program and found no issues.

In future versions I’d like to make a nice UI that is not just console code. Allowing the user to click buttons instead of entering commands. Possibly in the future as well I’d like to add more functions like adding pictures or icons to items. Perhaps have lists of items you can import or same them to an excel sheet. For now this version I am happy with and proud to call a small project of mine.