**Oscar So**

**CS 2024 – C++ Programming**

**Assignment #11: Input/Output**

**Nov 16, 2019**

This assignment was really short and simple given my background knowledge in Python and Java input output for files. I really like fstream in C++ because it allows the programmer to treat each stream just like a cin or cout, except we are using outputfile and infile instead. For my thought process, I knew that all that was necessary was to save the state at the end of the program termination, and reload them upon starting. This meant that there was only two new “functions” needed inside the program to be added. I started with the removal, which was really simple since I already had a freeAccounts() function inside the Bank file that deleted the dynamically allocated pointers from last assignment. Thus, when the user types in the last b, the lambda function returns true while doing the function that removes the pointers, which now also writes to the .txt file. The three things that are written in each line is: the number, the name, and the balance. This is the same as every other bank account. We don’t need to care about in program updates because the file won’t need to know that since it only stores final values and reuploads them after.

As for updating the program run into its old state, aka “reloading”, all we had to do was read the file before mainmenu.select() is called. Therefore, I created a function called addAccounts() inside Bank, that would read the .txt file line by line, get the number, name, balance, which are spaced out with a space, which is a whitespace, that input treats as a separate input. Thus, we are able to simply get all three variables of the bank account with one read, and you just run a while loop getting all information, while creating a new Bank Account and its pointers and adding them into the map. This is done using the addingAccount() function, which is just creating a new BankAccount pointer and setting variables. Since removing bankaccounts wasn’t an option, this was rather easy to code. Furthermore, I also took an easy route using whitespace as a delimiter instead of using commas, which would require me to use getline() or something alone that nature to get all the inputs instead.