CS 2024: Assignment 6 Report

The problem we are solving in this assignment is to build an Account class to keep track of an account balance. The Account class has a double variable called balance to track the balance of the account and 3 three methods: debit, credit, and getBalance. The constructor of the Account class should prevent an initial balance of less than zero. An error should be printed out and the balance of the account should be set to 0. The debit function would deduct a certain amount from the Account’s balance which would be passed as an argument into the function. If the amount being deducted is more than the total available balance, an error message should be printed out and nothing should be deducted. The return value of the debit function would be a bool to check whether the function completed correctly. The credit function would add a certain amount to the Account’s balance which would be passed as an argument into the function. The getBalance function would return the Account’s current balance value. After creating the Account class, we were then asked to create the SavingsAccount and CheckingAccount class. CheckingAccount and SavingsAccount should have the exact same methods and variables as Account with some extra variables and methods added. CheckingAccount should have an additional variable called transaction fee. The transaction fee should be deducted from the CheckingAccount’s balance every time a debit or credit is completed successfully. The SavingsAccount would have an additional variable called interestRate. The interestRate would be used to calculate the SavingsAccount’s interest in the calculateInterest function.

Since SavingsAccount and CheckingAccount share all of Account’s variables and methods, this problem is best solved using inheritance. Account was set up bas the base class from which SavingsAccount and CheckingAccount would be derived from. To do this, I created the Account class normally as according to the specifications above. Then, when creating the SavingsAccount and CheckingAccount classes, I included the AccountH.h header file for the Account class and inherited all of the variables and methods from the Account class when creating the CheckingAccount and SavingsAccount classes. This was easy to do as C++ provides nice syntax to inherit classes. When creating the constructors for CheckingAccount and SavingsAccount, I used the constructor of Account and then initialized the remaining variables for the respective classes that were not covered by the constructor of Account. To rework CheckingAccount’s debit and credit methods to deduct the transaction fee every time the operations completed successfully, I overloaded the debit and credit methods of the Account function. However, when implementing the debit and credit methods of the CheckingAccount class, I used the Account’s class debit method to deduct the transaction fee. By overloading the functions, I was able to make the debit and credit methods perform new functions, yet still keep the functionality of the debit and credit methods of the Account class.

The main point of this project was to teach me inheritance and overloading in C++. The only difficulty I had with this project was that it was tricky dealing with the “includes” of the various files. I had to make sure that the “includes” weren’t being doubly included and that each class would only be defined once. I ensured that the Account class is only defined once by surrounding it by a #ifndef and #endif clause to prevent the class from being defined multiple times. This allowed the program to compile properly.