Luke Williams

Mr. Landry

Algorithm

15 November 2018

Project 3

1. There will need to be three classes (transaction, item, and summary).

Class transaction{

Public:

Transaction()

~transaction()

Transaction(int id, double sellPrice, double costPrice, string time)

Void setTransactionID()

Void setTransactionTime()

Int getTransactionID()

Char\* getTransactioinTime()

Private:

Int transactionID

String time

Item item

Class item

Public:

Item()

Item(double sellPrice, double costPrice, string name)

~item()

Void setItemName()

Void setItemCost()

Void setItemSell()

Char\* getItemName()

Double getItemCost()

Double getItemSell()

Private:

String name

Double sell

Double cost

Class summary

Public:

Summary()

~summary()

Void printSummary()

Void newTransaction()

Void removeTransaction()

Private:

Transaction transactions[10]

1. Steps for solving:
2. Create a item class
3. Create a transaction class
4. Create a summary class
5. Create a switch for user options in main function
6. Nest the switch into a loop for a functioning menu
7. Finalize program
8. As usual I ran into a lot of issues. My compiler won’t run my program and I can’t figure out why. My solution isn’t the greatest looking back on it. I didn’t need to implement a summary class. I should have put it in my main function. I wanted a dynamic array for my summary but couldn’t figure out how to get new and delete to work properly with my objects. I have functions that call functions that call functions which is probably bad coding. I always feel like I understand the concepts in class and do well when I test my knowledge, but I can’t ever figure out how to put things together and implement them in a program. Stage 1 was to construct transaction and item classes. My train of thought was off from the beginning. I had to change my member functions multiple times to figure out how I should be getting the necessary member variables. Stage 2 was building a summary and time class. Originally, I was going to put time as members of the transaction class, but decided to break things up a little more. It was difficult trying to figure out how to put transaction objects into a summary array. Once that was accomplished it was difficult to access the private members of time and item classes since they are members of the transaction objects (hence why I have functions that call functions). Stages 1 and 2 both compiled before I moved on to putting the main function together. I set up a menu and realized I needed to go back and put in more summary functions. This is where I ran into most of the issues with my program.