

Project in C++ OOP

Hergeir Winther Lognberg
Hewi1600

1 Preamble

Assignment was to create a Bank which operated the way the lab described.

2 The Code

2.1 placement

I've decided to keep all files associated with the lab in the root of the project folder. There are in all 12 files 2 for each class and 2 for often used functions.

2.2 namespace

I chose to remove

```
1 using namespace std;
```

from all files it was previously used in. I find that this improves readability and clearly separates std functions from self-made ones.

2.3 code

I chose to implement the class as it was put up to in the Lab. But not exactly. To ease the readability (and manageability) of the read/load from file stuff. I overloaded the *std::ostream* and *std::istream* with operators << and >> so that i could use following syntax:

```
1 Account account;  
2 while (is >> account)  
3 {  
4     //push account to accounts vector  
5 }
```

By the way I just accidentally discovered the ternary operator and found out that I love it! :D For some reason I had never seen it used anywhere at all and didn't know what it was. I might have over used it (really don't know) but I like the conciseness of code it gives. f.x.

```
1 const bool Account::withdraw(const double amount)
2 {
3     return amount>getUsableBalance() ? false : (balance-=amount);
4     /*
5     if (amount > getUsableBalance())
6     {
7         return false;
8     }
9     else
10    {
11        balance-=amount;
12        return true;
13    }
14    */
15 }
```

In this case ternary reduces 9 lines of code to 2 just as readable lines. Still I have not had the need for a shared pointer.

2.4 AccountInfo

To easily manage and return account info i created the following struct:

```
1 struct AccountInfo
2 {
3     const unsigned int accountNo;
4     const double balance;
5     const double credit;
6     const double available;
7     //default constructor
8     AccountInfo()
9     :accountNo(0),balance(0),credit(0),available(0){}
10    //constructor
11    AccountInfo(const unsigned int pAccountNo,const double ↵
12                pBalance, const double pCredit, const double pAvailable)
13    :accountNo(pAccountNo),balance(pBalance),credit(pCredit),↵
14      available(pAvailable){}
15 };
```

I return this struct up through the textclasses:

$$Account \rightarrow Cutsomer \rightarrow Bank$$

All of them contain a function called:

```
1 ClassName::getAccountInfo(const unsigned int)
```

for whenever printing account info on specific account is needed.

3 Building/Compiling

Just run *make* in the Lab directory. To run the program run *make run* in same directory.

4 Enviroment

I'm programming on an Arch linux 64-bit system. I've got the gcc compiler installed and compile using it's g++ alias which links necessary libraries automatically. To compile I use the recommended flags: "-std=c++11 -Wall -pedantic". The flags let me choose to use c++11 standard and give me useful compiling warnings and errors. For editing of code i use VS code.

5 Backup

And if anything's missing you can find it on:

github: <https://github.com/Hergeirs/Cpp-Obj/tree/master>
Cpp-obj/Lab1

September 17, 2017