CS150 Assignment 5 Checking Account

Date assigned: Wednesday, November 4, 2009
Design Documents Due: Friday, November 6, 2009, 8 pm (5 points)
Date due: Friday, November 13, 2009, 9:15 am (45 points)
Total points: 50

Problem

You are to write a program that will process a collection of checking account transactions for multiple customers.

The input will consist of a series of lines of data, the first line containing the account number and the previous balance. Each subsequent line will contain: (a) the account number, (b) a date in the form 991013, (c) one of two characters (D - deposit, or W - withdrawal) and (d) an amount of money to be deposited or withdrawn. A deposit transaction is identified by the character D followed by the amount of the deposit. A withdrawal transaction is identified by the character W followed by the amount of the withdrawal. Processing of a specific account continues until another account begins or the marker value of 99999 is encountered.

The summary statistics for each customer will consist of:

- number of withdrawals
- total sum of all the withdrawals
- number of deposits
- total sum of all the deposits
- lowest balance during month
- highest balance during month

Use the following format for each account output:

```
         Account: ####

Previous Balance: $####.##

Date    Withdrawals($)      Deposits($)   Balance($)
######       ######.##                     ######.##
######                        ######.##    ######.##
######                        ######.##    ######.##
######       ######.##                     ######.##

(##) Withdrawals Totaled $######.##
(##) Deposits    Totaled $######.##

Lowest  Balance during month was $######.##
Highest Balance during month was $######.##
```

Start the output for each account on a **new screen**. Print all information for an account, including the summary statistics, on a separate screen and pause the output for each account until the user hits a key. Remember, the commands you will need are:

```
system("cls");    // clears the screen.
system("pause"); // pauses the program until
                  // the user hits a key AND
                  // Prints "Press any key to continue . . ."
                  // on the screen
                  // No extra include libraries are needed!
```

Here is a **sample datafile** you can use to run your program.

```
12345 500.00
12345 971001  D  100.00
12345 971005  W   50.00
76543 400.00
76543 971001  W   39.95
99999
```

Notes

1. Minimally, a data file must consist of at least the first line (an account # and beginning balance) and the last line (a marker value of 99999).
2. I will not test your program with data that makes the balance go negative so do not worry about this case. Make sure your program can handle all other cases.
3. Assume that the output for each account will not run over a screen's worth of space.
4. Read your account data from a file called **bankdata.txt**.

To complete this assignment you must

1. Create a new C++ project in Visual Studio. Name your project **05Checkingxxxxxxxx**, where xxxxxxxx must be replaced by your PUNetID. As an example, my project would be called "05Checkingryandj". It is vital that you name your project correctly!
2. Type the solution (**fully documented/commented**) to the problem into your project.
3. Remember to enter in your name as the author of the program.
4. Make sure that your program compiles and runs correctly. If you get any errors, double check that you typed everything correctly. Be aware that C++ is case-sensitive. Also, there must not be any warnings when compiling your program.
5. Once you are sure that the program works correctly, it is time to submit your program. You do this by logging on to Turing and placing your complete project folder in the **CS150-02 Drop** folder. Make sure that you copy your program folder and don't move the folder. If you move the folder, then you will not have your own copy!

Submit an electronic copy of your design document

Before you start you need to think about the data in your program and the calculations and loops you will need to perform. Answer the following questions in a **new** GoogleDoc

(**CS150_05ProgramDesignPUNetID**) and share the document with the instructor (profchadd@gmail.com). Be sure to answer the questions in complete sentences where appropriate. This design document is due on **Friday at 8 pm.**

Design Questions:

1. List each variable declaration necessary to store the data and information in your program. Be sure to name your variables clearly so readers of your code will have no problem understanding their purpose. Pay very close attention to the data types for each variable.

2. Briefly describe the calculations you will need to perform in your program. Be sure to explain which variables from 1. will be used in each calculation.

3. For each loop used in your program, discuss what will happen in the loop and what data and conditions will be used by the program to stop the loop.

4. Write a small test case (data file) that will test some portion of your program that is not tested by the sample data file given above. Describe, in English sentences, which portion of your code will be tested by this new test case.

Notes

1. You must follow the coding standards.

2. You must use constants when possible.

3. Your program will be graded on **efficiency**. In other words, you will be marked down for repeating code statements unnecessarily.

4. You may only use the C++ programming concepts covered thus far in class. Do not use any more advanced concepts that we have not covered or any other programming concepts that you have had experience with.

5. Your output must look **exactly** like the sample given.

6. If this program sounds difficult, it's not that bad if you get an EARLY start. Make sure you understand all of the pieces before beginning to code your solution. Code your solution a piece at a time not all at once. It makes for much smoother debugging.

Remember, this is an individual assignment. Refer to the syllabus for assignment policies