

PROBLEM STATEMENT:

You are tasked with writing a program which will write a monthly account summary of customers' transactions at the bank at which you are employed. The monthly transactions are stored in a file called transactions.txt in the follow form:

```
123    34.67 d45.10      d50.45      d198.56      w45.67
```

The first entry is the account number, followed by the current account balance at the beginning of the month. The remainder of the line, variable in length, contains all the transactions of the particular customer for the month: if the entry begins with a d, the number that follows is the amount of deposit; if it begins with a w, the number that follows is the amount of withdraw.

The customer "database" is stored in another file, accounts.txt which has entries of the form

```
123    daffy
```

where each line contains the customer's account number followed by their name. The first entry of this file indicates the number of entries in the so-called "database"

Your program will write report in the following form ( given the above data )

```
Name : daffy
Account Number : 123
Beginning Balance : $34.67
Ending Balance : $283.11
Amount Deposited : $294.11
Number of Deposits : 3
Amount Withdrawn : $45.67
Number of Withdraws : 1
```

The report is to be written to a file called monthly\_report.txt

#### CODE:

Your program will read a file called monthly\_transactions.txt, which contains the listing of all transactions for the month; a file called accounts.txt will contain a list of all customers, along with their account ids' ( see posted files for sample files ). Your program is to work with any files with the above names and correct format  
You must use the following structure to hold the data of each customer:

```
struct Account    // struct account type to hold customer info
{
    int acct_num;           // customer account number
    string name;           // customers name
    int acct_bal;          // customers account balance
};
```

Feel free to construct your own files to test/debug your program

#### DELIVERABLES:

hard :

- In a bound folder
1. documented source code
  2. user manual.
  3. programmer manual.

Due Date : 1:00pm 30 November 2023

soft:

- in a zipped file, called CS121\_P2\_yourLastName
1. all source code
  2. release version executable. The .exe must be compiled and run on windows ( no exe, grade = 0]; would you ever ship a product without the application exe for the user ?
  3. user manual.
  4. user manual.
- Of course, documentation standards need to be followed
5. **Do NOT include any other files. Other files included will receive an automatic 15 point deduction**

**submitted to BrightSpace to CS121\_P2\_ANS**

Due Date : 1:00pm 30 November 2023