CISC327 Course Project Assignment #3

1. **Modification from A1 and A2**
2. **Failure report in table form**

Company: Canada Trust

Yudong Zhou 20083467
Yitong Liu 20028039
Runze Yi 20073329
Tong Bu 20079649

1. Modifications form A1 and A2

There were 2 major mistakes pointed out by the TA in our feedback from Assignment 1. The first one being the summary file for deleting and creating accounts. Those files should have 0000000 instead of 000 as money. The second one being transfer should not and could not check for whether the amount to transfer exceeds the account balance. According to these, output summary files for deleteacct and createacct are modified to have seven zeros instead of three, and test cases R5T5 and R5T6 for transfer are removed.

We also discovered that we are missing the “eos” in our summary file as a end of session. We then modified our summary output to add that.

Another mistake that had little to do with the code is that we did not allow spaces in names for creating accounts at all, where as spaces that are not in front and end are permitted. The test cases R6T1 and R6T2 are modified as well as the code.

R3T11 for deposit has a mistake of getting the mode selection as ATM instead of machine. The unified choice was supposed to be a machine, and is fixed now.

R4T1 for deleteacct is designed for testing the feature of not being able to perform transactions on the deleted account. It had an error of not deleting the account before depositing from it, and is fixed now.

R5T1 to R5T5 for createacct had no logout at the end of their input, making it impossible to compare the output files. Since they are functionality tests instead of expected errors, that matters. Logouts are added to the inputs now.

Our code for A2 was initially 2 classes and do not require the input of “login”. One class called Login and one called frontend. The first class asks for the mode of login and the second class takes care of how transactions are performed. These two classes are merged into one class here and do need the input of “login”.

Our original A2 code also had no “console output” to demonstrate that an expected error is happening. For example, it should print out “The account name is invalid” when a wrong name is entered when creating an account. Print statements are added so that console outputs now behave as intended.

For the convention of testing, we designed the test in the fashion of being able to login after a logout, so tests could be performed much easier. Console outputs for similar cases are unified.

We also added a start feature which will reset the Valid account file to default (1234567 and 7654321) before every new case is tested. This prevents any error caused by changed parameters from previous tests.

2. Failure Table

Test Number	Description
createacct R7T1,R7T2,R7T3 No transactions on newly created accounts(Cannot accept the transaction for the new account in this session)	In our code form A2, a bug of transactions on newly created accounts were allowed. A verification of whether it is a newly created account is added. We have failed that the creating new account that can do transaction such as deposit, so right now we have added a couple lines of code for a “newAccountList”, that the new account cannot do transaction before a new transaction day
deleteacct R2T2 Check account name for deleting account	In our code from A2, a bug of being able to delete an account without checking the correct account name existed. A verification of the account name was added.
login R3 No subsequent logins before logging out	In our code from A2, “login login” was treated as the wrong mode, and is fixed to be treated as subsequent login, and is not allowed. A few exception throws were added to those tests.
createacct R5T2-R5T5 Check name for creating account, only alphanumeric characters are accepted	In our test cases from A1, there are no logouts at the end. They are added so we could compare the desired output files. In the code, several lines of assert are added to help to build the summary file. This is considered as both a failure case and a modification to previous work since it is impossible to compare the output files without those lines of assert.

createacct R6T1 R6T2 No spaces in name at beginning and end when creating account	In our code from A2, these tests failed to detect spaces in account names. Turns out that code which is responsible for that is missing, and is added while we tested it.
deposit R4T2-R4T5 Account numbers that are too long, too short, contains letters, start with 0, not exist account number are not allowed.	A unified mode of agent and machine were used in A1, but inconsistency of agent and ATM appeared in our code from A2, causing failure reports for mode selection, and is fixed now.(We have check if the mode is ATM, but actually the mode is machine or agent, not ATM, which will cause bug when asking the deposit amount)
withdraw R4T7-R4T9 Account numbers that are too long, too short, contains letters, start with 0, not exist account number are not allowed.	A unified mode of agent and machine were used in A1, but inconsistency of agent and ATM appeared in our code from A2, causing failure reports for mode selection, and is fixed now.(We have check if the mode is ATM, but actually the mode is machine or agent, not ATM, which will cause bug when asking the withdraw amount)
Transfer R5T7-R6T1 Account numbers that are too long, too short, contains letters, start with 0, not exist account number are not allowed.	A unified mode of agent and machine were used in A1, but inconsistency of agent and ATM appeared in our code from A2, causing failure reports for mode selection, and is fixed now.(We have check if the mode is ATM, but actually the mode is machine or agent, not ATM, which will cause bug when asking the transfer amount)