

# CMPE 327: Assignment 1

Steven Ding  
2019-10-03

Connor Crowe	- 20009994
Jordan Mack	- 20005220
Michael Briggs	- 20013906
Sasanka Wickramasinghe	- 10192504

## **Test Organization**

The written tests will be organized into two separate folders. These folders are named “In” and “Out” for the Input and Output respectively. The files located in the “In” folder will start with “in\_\_” to show that it belongs to the Input folder, and “out\_\_” for the output folder. Both folders will have the following folders: Create, Delete, Deposit, Login, Logout, Transfer, and Withdraw where the corresponding tests or outputs will be stored.

To see a list of all tests and the specific requirements they address, see TestList.xls.

## **Test Running**

Running the tests will be primarily executed through Powershell command line where the inputs will be run through the program. The tests will start with Create, Delete, and so on. In each test, any account that has been created will be deleted before exiting the test. This will allow for smooth running of the tests as there is no need to check for repeated account numbers or names. For these tests, the account numbers 1234567, and 7654321 must be reserved from regular use.

## **Output Validation**

A PowerShell script will run to compare the output of the program given the input file against the pre-recorded, expected output files with the proper results. If the test transaction summary does not match the one written ahead of time it reports that the test failed. This result is then saved as a timestamped document in the Results folder. The result will contain the actual output transaction summary from the program. This way, if needed each test output can be manually compared to the correct output.

## **Results Organization**

On GitHub there will be a Results folder. In the results folder will be the same folders listed in the “In” and “Out” folders; that is, one folder for each transaction type (“Deposit”, “Login”, etc.). Then in each of these folders the results will be stored. Each result file will be named so that the name contains the test number, a date that the test was run, and whether it passed or failed.

Example: Results/Deposit/

```
DE1_2019-10-13_16:27_FAIL.txt
DE1_2019-10-14_08:30_PASS.txt
DE2_2019-10-13_16:28_FAIL.txt
DE2_2019-10-13_16:50_PASS.txt
```

This way the results are sorted first by test type, then by date, and then each entry also at a glance indicates if the overall result was a pass or a fail. Inside each document will be the actual transaction summary generated by the program for the input data given, so that it is easy to manually compare it to the correct output by hand when it is time to find errors.