|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Name | | Test Intention | Input Files | Output Files |
| LOGIN | test\_single\_login | No subsequent login is accepted after login | 1.input\_test\_single\_login.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_single\_login.txt  2. transaction\_file\_empty.txt |
| test\_txn\_login | Only accept machine transactions and logout once logged in | 1. input\_test\_txn\_login.txt  2. valid\_accounts\_test\_txn\_login.txt | 1. output\_test\_txn\_login.txt  2. transaction\_file\_empty.txt |
| test\_teller\_all\_txn\_login\_agent | If in agent mode all transactions are accepted | 1. input\_test\_teller\_all\_txn\_login\_agent.txt  2. valid\_accounts\_file\_test\_teller\_all\_txn\_login\_agent.txt | 1. output\_test\_teller\_all\_txn\_login\_agent.txt  2. transaction\_file\_empty.txt |
| test\_login\_read\_acc\_file | Test that after the login the system reads & stores the account list file | 1. input\_test\_login\_read\_acc\_file.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_login\_read\_acc\_file.txt  2. transaction\_file\_empty.txt |
| test\_mode \_agent | Test that the mode is successfully changed to agent | 1. input\_test\_mode\_agent.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_mode\_agent.txt  2. transaction\_file\_empty.txt |
| test\_mode\_ machine | Test that the mode is successfully changed to machine | 1. input\_test\_mode\_machine.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_mode\_machine.txt  2. transaction\_file\_empty.txt |
| test\_invalid\_user\_input | Test that user input is invalid and return true | 1. input\_test\_invalid\_user\_input.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_invalid\_user\_input.txt |
|  |  |  |  |  |
| LOGOUT | test\_logout\_when\_login | Test successful logout if logged in | 1. input\_logout\_when\_login.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_logout\_when\_login.txt  2. transaction\_file\_empty.txt |
| test\_logout\_when\_logout | Logout should only be accepted when logged in | 1. input\_test\_logout\_when\_logout.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_logout\_when\_logout.txt |
| test\_txn\_logout | Don’t accept machine transactions if logged out | 1. input\_test\_txn\_logout.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_txn\_logout.txt  2. transaction\_file\_empty.txt |
| test\_teller\_all\_txn\_logout\_agent | No machine transactions accepted if agent logged out | 1. input\_test\_teller\_all\_txn\_logout\_agent.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_teller\_all\_txn\_logout\_agent.txt  2. transaction\_file\_empty.txt |
| test\_after\_logout | Accept the login if you are logged out | 1. input\_test\_after\_logout.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_after\_logout.txt  2. transaction\_file\_empty.txt |
| test\_create\_txnfile\_logout | Test that after logout transaction file is successfully created | 1. input\_create\_txnfile\_logout.txt  2. valid\_accounts\_file\_test\_create\_txnfile\_logout.txt | 1. output\_create\_txnfile\_logout.txt  2. transaction\_file\_empty.txt |
|  |  |  |  |  |
| CREATEACCT | test\_before\_login | No CREATEACCT transaction accepted before the login | 1.input\_test\_before\_login.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_before\_login.txt |
| test\_accept\_agent | Test “privileged” transaction accepted if in agent mode | 1. input\_test\_accept\_agent.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_accept\_agent.txt  2. transaction\_file\_test\_accept\_agent.txt |
| test\_deny\_machine | Test “privileged” transaction denied if in machine mode | 1. input\_test\_deny\_machine.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_deny\_machine.txt  2. transaction\_file\_empty.txt |
| test\_acc\_seven\_digits | Test if the account is exactly 7 decimal digits | 1. input\_test\_acc\_seven\_digits.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_acc\_seven\_digits.txt  2. transaction\_file\_empty.txt |
| test\_acc\_first\_digit\_non\_zero | Test if the account number starts with non-zero decimal digit | 1. input\_test\_acc\_first\_digit\_non\_zero.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_acc\_first\_digit\_non\_zero.txt  2. transaction\_file\_empty.txt |
| test\_acc\_unique | Test that the new account number is unique from all others | 1. input\_test\_acc\_unique.txt  2. valid\_accounts\_file\_test\_acc\_unique.txt | 1. output\_test\_acc\_unique.txt  2. transaction\_file\_empty.txt |
| test\_acc\_name\_num\_char | Test that the new account name is between 3-30 characters | 1. input\_test\_acc\_name\_num\_char.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_acc\_name\_num\_char.txt  2. transaction\_file\_test\_acc\_name\_num\_char.txt |
| test\_acc\_number\_alpha\_numeric | Test non- alpha numeric characters are rejected in account number. | 1. input\_test\_acc\_number\_alpha\_numeric.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_acc\_number\_alpha\_numeric.txt  2. transaction\_file\_test\_acc\_number\_alpha\_numeric.txt |
| test\_acc\_name\_alpha\_numeric | Test non- alpha numeric characters are rejected, excluding space | 1. input\_test\_acc\_name\_alpha\_numeric.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_acc\_name\_alpha\_numeric.txt  2. transaction\_file\_test\_acc\_name\_alpha\_numeric.txt |
| test\_acc\_name\_trailing\_space | Test that there is no beginning or ending space in name | 1. input\_test\_acc\_name\_trailing\_space.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_acc\_name\_trailing\_space.txt  2. transaction\_file\_test\_acc\_name\_trailing\_space.txt |
| test\_notxn\_on\_new\_acc | Test that no transactions are accepted on a new account created | input\_test\_notxn\_on\_new\_acc.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_notxn\_on\_new\_acc.txt  2. transaction\_file\_test\_notxn\_on\_new\_acc.txt |
| test\_txn\_logout | Don’t accept CREATEACCT transaction if logged out | 1. input\_test\_txn\_logout.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_txn\_logout.txt  2. transaction\_file\_empty.txt |
|  |  |  |  |  |
| DELETEACCT | test\_before\_login | No DELETEACCT transaction accepted before the login | 1.input\_test\_before\_login.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_before\_login.txt |
| test\_accept\_agent | Test “privileged” transaction accepted if in agent mode | 1. input\_test\_accept\_agent.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_accept\_agent.txt  2. transaction\_file\_test\_accept\_agent.txt |
| test\_deny\_machine | Test “privileged” transaction denied if in machine mode | 1. input\_test\_deny\_machine.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_deny\_machine.txt  2. transaction\_file\_empty.txt |
| test\_txn\_on\_delacc | Test that no transactions are accepted on a deleted account | 1. input\_test\_txn\_on\_delacc.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_txn\_on\_delacc.txt  2. transaction\_file\_test\_txn\_on\_delacc.txt |
| test\_acc\_number\_alpha\_numeric | Test non- alpha numeric characters are rejected in account number | 1. input\_test\_acc\_number\_alpha\_numeric.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_acc\_number\_alpha\_numeric.txt  2. transaction\_file\_test\_acc\_number\_alpha\_numeric.txt |
| test\_valid\_acc\_num | Test account number that exists in the accounts list can be deleted | 1. input\_test\_valid\_acc\_num.txt  2. valid\_accounts\_file\_test\_valid\_acc\_num.txt | 1. output\_test\_valid\_acc\_num.txt  2. transaction\_file\_test\_valid\_acc\_num.txt |
| test\_txn\_logout | Don’t accept DELETEACCT transaction if logged out | 1. input\_test\_txn\_logout.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_txn\_logout.txt  2. transaction\_file\_empty.txt |
|  |  |  |  |  |
| DEPOSIT | test\_before\_login | No DEPOSIT transaction accepted before the login | 1.input\_test\_before\_login.txt  2. valid\_accounts\_test\_before\_login.txt | 1. output\_test\_before\_login.txt |
| test\_valid\_acc\_num | Test account number exists in the list | 1. input\_test\_valid\_acc\_num.txt  2. valid\_accounts\_file\_test\_valid\_acc\_num.txt | 1. output\_test\_valid\_acc\_num.txt  2. transaction\_file\_test\_valid\_acc\_num.txt |
| test\_valid\_amount | Test amount doesn’t exceed “1,000.00” and is non-negative | 1. input\_test\_valid\_amount.txt  2. valid\_accounts\_file\_test\_valid\_amount.txt | 1. output\_test\_valid\_amount.txt  2. transaction\_file\_test\_valid\_amount.txt |
| test\_valid\_amount\_leading\_zero | Test amount doesn’t have leading zeros. | 1. input\_test\_valid\_amount\_ leading\_zero.txt  2. valid\_accounts\_file\_test\_valid\_amount\_ leading\_zero.txt | 1. output\_test\_valid\_amount\_ leading\_zero.txt  2. transaction\_file\_test\_valid\_amount\_ leading\_zero.txt |
| test\_valid\_amount\_alpha\_numeric | Test amount doesn’t contain non-numeric characters. | 1. input\_test\_valid\_amount\_alpha\_numeric.txt  2. valid\_accounts\_file\_test\_valid\_amount\_ alpha\_numeric.txt | 1. output\_test\_valid\_amount\_ alpha\_numeric.txt  2. transaction\_file\_test\_valid\_amount\_ alpha\_numeric.txt |
| test\_valid\_amount\_agent | Test amount doesn’t exceed “999,999.99” and is non-negative | 1. input\_test\_valid\_amount\_agent.txt  2. valid\_accounts\_file\_test\_valid\_amount\_agent.txt | 1. output\_test\_valid\_amount\_agent.txt  2. transaction\_file\_test\_valid\_amount\_agent.txt |
| test\_txn\_invalid\_acc\_num | Test that the account number is invalid | 1. input\_test\_txn\_invalid\_acc\_num.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_txn\_invalid\_acc\_num.txt  2. transaction\_file\_empty.txt |
| test\_txn\_logout | Don’t accept DEPOSIT transaction if logged out | 1. input\_test\_txn\_logout.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_txn\_logout.txt  2. transaction\_file\_empty.txt |
|  |  |  |  |  |
| WITHDRAW | test\_before\_login | No WITHDRAW transaction accepted before the login | 1.input\_test\_before\_login.txt  2. valid\_accounts\_test\_before\_login.txt | 1. output\_test\_before\_login.txt |
| test\_valid\_acc\_num | Test account number exists in the list | 1. input\_test\_valid\_acc\_num.txt  2. valid\_accounts\_file\_test\_valid\_acc\_num.txt | 1. output\_test\_valid\_acc\_num.txt  2. transaction\_file\_test\_valid\_acc\_num.txt |
| test\_valid\_amount | Test amount doesn’t exceed “1,000.00” and is non-negative | 1. input\_test\_valid\_amount.txt  2. valid\_accounts\_file\_test\_valid\_amount.txt | 1. output\_test\_valid\_amount.txt  2. transaction\_file\_test\_valid\_amount.txt |
| test\_valid\_amount\_leading\_zero | Test amount doesn’t have leading zeros. | 1. input\_test\_valid\_amount\_ leading\_zero.txt  2. valid\_accounts\_file\_test\_valid\_amount\_ leading\_zero.txt | 1. output\_test\_valid\_amount\_ leading\_zero.txt  2. transaction\_file\_test\_valid\_amount\_ leading\_zero.txt |
| test\_valid\_amount\_alpha\_numeric | Test amount doesn’t contain non-numeric characters. | 1. input\_test\_valid\_amount\_alpha\_numeric.txt  2. valid\_accounts\_file\_test\_valid\_amount\_ alpha\_numeric.txt | 1. output\_test\_valid\_amount\_ alpha\_numeric.txt  2. transaction\_file\_test\_valid\_amount\_ alpha\_numeric.txt |
| test\_valid\_amount\_agent | Test amount doesn’t exceed “999,999.99” and is non-negative | 1. input\_test\_valid\_amount\_agent.txt  2. valid\_accounts\_file\_test\_valid\_amount\_agent.txt | 1. output\_test\_valid\_amount\_agent.txt  2. transaction\_file\_test\_valid\_amount\_agent.txt |
| test\_total\_withdrawal\_cap | Test that total withdrawals in a session doesn’t exceed “1,000” | 1. input\_test\_total\_withdrawal\_cap.txt  2. valid\_accounts\_file\_test\_total\_withdrawal\_cap.txt | 1. output\_test\_total\_withdrawal\_cap.txt  2. transaction\_file\_test\_total\_withdrawal\_cap.txt |
| test\_txn\_logout | Don’t accept WITHDRAW transaction if logged out | 1. input\_test\_txn\_logout.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_txn\_logout.txt  2. transaction\_file\_empty.txt |
|  |  |  |  |  |
| TRANSFER | test\_before\_login | No TRANSFER transaction accepted before the login | 1.input\_test\_before\_login.txt  2. valid\_accounts\_test\_before\_login.txt | 1. output\_test\_before\_login.txt |
| test\_both\_valid\_acc\_num | Test both provided account numbers exist in the list | 1. input\_test\_both\_valid\_acc\_num.txt  2. valid\_accounts\_file\_test\_both\_valid\_acc\_num.txt | 1. output\_test\_both\_valid\_acc\_num.txt  2. transaction\_file\_test\_both\_valid\_acc\_num.txt |
| test\_valid\_amount | Test amount doesn’t exceed “1,000.00” and is non-negative | 1. input\_test\_valid\_amount.txt  2. valid\_accounts\_file\_test\_valid\_amount.txt | 1. output\_test\_valid\_amount.txt  2. transaction\_file\_test\_valid\_amount.txt |
| test\_valid\_amount\_leading\_zero | Test amount doesn’t have leading zeros. | 1. input\_test\_valid\_amount\_ leading\_zero.txt  2. valid\_accounts\_file\_test\_valid\_amount\_ leading\_zero.txt | 1. output\_test\_valid\_amount\_ leading\_zero.txt  2. transaction\_file\_test\_valid\_amount\_ leading\_zero.txt |
| test\_valid\_amount\_alpha\_numeric | Test amount doesn’t contain non-numeric characters. | 1. input\_test\_valid\_amount\_alpha\_numeric.txt  2. valid\_accounts\_file\_test\_valid\_amount\_alpha\_numeric.txt | 1. output\_test\_valid\_amount\_alpha\_numeric.txt  2. transaction\_file\_test\_valid\_amount\_alpha\_numeric.txt |
| test\_valid\_amount\_agent | Test amount doesn’t exceed “999,999.99” and is non-negative | 1. input\_test\_valid\_amount\_agent.txt  2. valid\_accounts\_file\_test\_valid\_amount\_agent.txt | 1. output\_test\_valid\_amount\_agent.txt  2. transaction\_file\_test\_valid\_amount\_agent.txt |
| test\_both\_invalid\_acc\_num | Test that the “to-from” account numbers are invalid | 1. input\_test\_both\_invalid\_acc\_num.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_both\_invalid\_acc\_num.txt  2. transaction\_file\_empty.txt |
| test\_invalid\_to\_acc\_num | Test that the “to” account number is invalid and “from” is valid | 1. input\_test\_invalid\_to\_acc\_num.txt  2. valid\_accounts\_file\_test\_invalid\_to\_acc\_num.txt | 1. output\_test\_invalid\_to\_acc\_num.txt  2. transaction\_file\_empty.txt |
| test\_invalid\_from\_acc\_num | Test that the “to” account number is valid and “from” is invalid | 1. input\_test\_invalid\_from\_acc\_num.txt  2. valid\_accounts\_file\_test\_invalid\_from\_acc\_num.txt | 1. output\_test\_invalid\_from\_acc\_num.txt  2. transaction\_file\_empty.txt |
| test\_txn\_logout | Don’t accept TRANSFER transaction if logged out | 1. input\_test\_txn\_logout.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_txn\_logout.txt  2. transaction\_file\_empty.txt |
|  |  |  |  |  |
| TRANSACTION SUMMARY FILE | test\_line\_char\_count **\***  (Output in form of test results per line of input file) | Test max 61 characters plus newline | 1. input\_test\_line\_char\_count.txt | 1. output\_test\_line\_char\_count.txt |
| test\_txn\_code **\***  (Output in form of test results per line of input file) | Test first 3 characters are one of: DEP,WDR,XPR,NEW,DEL,EOS | 1. input\_test\_txn\_code.txt | 1. output\_test\_txn\_code.txt |
| test\_space\_between\_items **\***  (Output in form of test results per line of input file) | Test that all the items are separated by exactly one space | 1. input\_test\_space\_between\_items.txt | 1. output\_test\_space\_between\_items.txt |
| test\_acc\_seven\_digits **\***  (Output in form of test results per line of input file) | Test if the account is exactly 7 decimal digits | 1. input\_test\_acc\_seven\_digits.txt | 1. output\_test\_acc\_seven\_digits.txt |
| test\_acc\_first\_digit\_non\_zero **\***  (Output in form of test results per line of input file) | Test if the account number starts with non-zero decimal digit | 1. input\_test\_acc\_first\_digit\_non\_zero.txt | 1. output\_test\_acc\_first\_digit\_non\_zero.txt |
| test\_monetary\_amount **\***  (Output in form of test results per line of input file) | Test amount is between 3 and 8 digits | 1. input\_test\_monetary\_amount.txt | 1. output\_test\_monetary\_amount.txt |
| test\_acc\_name\_num\_char **\***  (Output in form of test results per line of input file) | Test that the new account name is between 3-30 characters | 1. input\_test\_acc\_name\_num\_char.txt | 1. output\_test\_acc\_name\_num\_char.txt |
| test\_acc\_name\_alpha\_numeric **\***  (Output in form of test results per line of input file) | Test non- alpha numeric characters are rejected, excluding space | 1. input\_test\_acc\_name\_alpha\_numeric.txt | 1. output\_test\_acc\_name\_alpha\_numeric.txt |
| test\_acc\_name\_trailing\_space **\***  (Output in form of test results per line of input file) | Test that there is no beginning or ending space in name | 1. input\_test\_acc\_name\_trailing\_space.txt | 1. output\_test\_acc\_name\_trailing\_space.txt |
| test\_unused\_num\_fields **\***  (Output in form of test results per line of input file) | Test unused numeric fields are filled with zero | 1. input\_test\_unused\_num\_fields.txt | 1. output\_test\_unused\_num\_fields.txt |
| test\_unused\_acc\_name\_field **\***  (Output in form of test results per line of input file) | Test unused account name fields are filled with 3(\*) | 1. input\_test\_unused\_acc\_name\_field.txt | 1. output\_test\_unused\_acc\_name\_field.txt |
| test\_eos **\***  (Output in form of test results per line of input file) | Test last line is EOS transaction code | 1. input\_test\_eos.txt | 1. output\_test\_eos.txt |
| test\_new\_acc\_created **\***  (Output in terms of transaction summary) | Test that line starting with “NEW” is for account number that was used in create account transaction | 1. input\_test\_new\_acc\_created.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_new\_acc\_created.txt |
| test\_delete\_acc **\***  (Output in terms of transaction summary) | Test that the account deleted starts with “DEL” in transaction message | 1. input\_test\_delete\_acc.txt  2. valid\_accounts\_file\_test\_delete\_acc.txt | 1. output\_test\_delete\_acc.txt |
| test\_deposit **\***  (Output in terms of transaction summary) | Test that the deposit transaction starts with “DEP” in transaction message | 1. input\_test\_deposit.txt  2. valid\_accounts\_file\_test\_deposit.txt | 1. output\_test\_deposit.txt |
| test\_deposit\_amount **\***  (Output in terms of transaction summary) | Test that deposited amount is correct in the transaction message | 1. input\_test\_deposit\_amount.txt  2. valid\_accounts\_file\_test\_deposit\_amount.txt | 1. output\_test\_deposit\_amount.txt |
| test\_withdraw **\***  (Output in terms of transaction summary) | Test that the withdraw transaction starts with “WDR” in transaction message | 1. input\_test\_withdraw.txt  2. valid\_accounts\_file\_test\_withdraw.txt | 1. output\_test\_withdraw.txt |
| test\_withdraw\_from **\***  (Output in terms of transaction summary) | Test that the transaction message has correct “from” account number | 1. input\_test\_withdraw\_from.txt  2. valid\_accounts\_file\_test\_withdraw\_from.txt | 1. output\_test\_withdraw\_from.txt |
| test\_withdraw\_amount **\***  (Output in terms of transaction summary) | Test that withdrawal amount is correct in the transaction message | 1. input\_test\_withdraw\_amount.txt  2. valid\_accounts\_file\_test\_withdraw\_amount.txt | 1. output\_test\_withdraw\_amount.txt |
| test\_transfer **\***  (Output in terms of transaction summary) | Test that the transfer transaction starts with “XFR” in transaction message | 1. input\_test\_transfer.txt  2. valid\_accounts\_file\_test\_transfer.txt | 1. output\_test\_transfer.txt |
| test\_transfer\_to\_from **\***  (Output in terms of transaction summary) | Test that the transaction message has correct to and from account numbers | 1. input\_test\_transfer\_to\_from.txt  2. valid\_accounts\_file\_test\_transfer\_to\_from.txt | 1. output\_test\_transfer\_to\_from.txt |
| test\_transfer\_amount **\***  (Output in terms of transaction summary) | Test that transfer amount is correct in the transaction message | 1. input\_test\_transfer\_amount.txt  2. valid\_accounts\_file\_test\_transfer\_amount.txt | 1. output\_test\_transfer\_amount.txt |
|  |  |  |  |  |
| VALID ACCOUNTS LIST FILE | test\_acc\_line\_char\_count **\***  (Output in form of test results per line of input file) | Test max 7 characters plus newline | 1. input\_test\_acc\_line\_char\_count.txt | 1. output\_test\_acc\_line\_char\_count.txt |
| test\_acc\_seven\_digits **\***  (Output in form of test results per line of input file) | Test if the account is exactly 7 decimal digits | 1. input\_test\_acc\_seven\_digits.txt | 1. output\_test\_acc\_seven\_digits.txt |
| test\_acc\_first\_digit\_non\_zero **\***  (Output in form of test results per line of input file) | Test if the account number starts with non-zero decimal digit | 1. input\_test\_acc\_first\_digit\_non\_zero.txt | 1. output\_test\_acc\_first\_digit\_non\_zero.txt |
| test\_acc\_list\_ends\_invalid\_acc **\***  (Output in form of test results per line of input file) | Test that the list ends with the invalid account number (0000000) | 1. input\_test\_acc\_list\_ends\_invalid\_acc.txt | 1. output\_test\_acc\_list\_ends\_invalid\_acc.txt |
|  |  |  |  |  |
| INPUT TEST | test\_invalid\_inputs | Test that the user input doesn’t match the required inputs | 1. input\_test\_invalid\_inputs.txt  2. valid\_accounts\_file\_empty.txt | 1. output\_test\_invalid\_inputs.txt  2. transaction\_file\_empty.txt |

Tests **with** “**\***” beside their respective test name are white-box tests. These tests assume that particular input parsing methods can be tested individually. For example, the valid account’s file parsing can be tested using white-box testing by verifying that the functions used to extract the valid account numbers from the file are appropriately following the system constraints.

All tests **without** “**\***” are black-box testing using purely system input and output for validation.

**Requirements Problems (missing information and assumptions made):**

* Assuming that the requirement for creating the account and not allowing any transactions on the new account is only until the end of processing day as this is not explicitly stated.
* Assuming that newly created accounts must be unique to not only the current account numbers in the valid accounts file but also from any newly created accounts created within this same transaction day which will be kept track of in a separate list.
* Assuming that if a successful logout does not occur, no transaction summary file will be created.
* Assumes that max withdraw and deposit limits will remain equal (to each other) and never change from their initial value.
* Assumption that max line character counts for transaction summary file and validate account list file will stay static (never change).
* Assumption that valid accounts file and transaction summary file can be tested using white-box testing by the front-end test suite. It was no clear in instructions if these components are part of the front end of the system.