**Non default functions listed. Function signatures included in .h file**

Class/Struct

* Bank – Reads transactions.txt. Perform transactions on accounts. Stores errors
  + ConvertTxtToQueue
  + ConvertStringToTransaction
  + ProcessQueue
  + ProcessTransactions
  + Errors – vector<string>, stores “Account does not exist” error
* Tree – Easy storage/retrieval of accounts
* Account – Stores account info like fund balances and transaction history
  + IsValidTransaction(int Fund, int Amount) – Returns whether withdrawal is valid
    - Checks if fund has linked fund that can supplement transaction
  + Balances – length 10 vector<int> that stores the balance of each fund
  + BalanceHistories – vector<vector<string>>. Stores transaction history for each fund within each account
    - Appends (Failed) to end if transaction is not valid
* Node – Enables traversal within tree. Stores Account object
* Transactions – Stores action and info needed to perform action (Account, amount, name)
  + Attributes - Account1, Account2, Action, Amount, Name
  + Irrelevant attributes defaulted to easily identifiable value

Flow

1. Bank takes in transaction list as text file
2. Bank converts each line in text file to Transaction object
3. Transaction object added to Queue
4. While Queue is not empty, process each Transaction, popping when finished
5. Process each transaction
   1. OpenAction – Creates new account and inserts in tree
   2. HistoryAction – Display either whole account history or specific fund history
   3. BalanceAction – Adjusts Balance of one or more accounts if valid
6. If transaction