Rony Vargas
155005725
CS214 - Systems Programming
Spring 2015

Assignment 4 – Readme

Behavior – The behavior of the server can be adjusted by changing the defined values, MAX_ATTEMPTS and REFRESH_RATE. MAX_ATTEMPTS (MA) will limit the amount of times to retry locking a mutex before it gives up. Obviously the more times you try the longer it will take for a response from the server. Its currently set to 1 which is min of this value. You can also change the RETRY_DELAY to change the wait time between attempts. This is currently set to 0. REFRESH_RATE will adjust the speed at which account status's are updated and printed onto the screen. This represents time in seconds.

Due to the string to float /vice-versa conversions, the displayed balance may be slightly inaccurate with really large/small numbers (decimals). However, this does not affect mathematical calculations when depositing and withdrawing from the accounts as long as the user keeps track of actual values.

Extras – I added the ability to transfer from one account to another. To use this ability you must be "served" into the account where the funds will be transferred from and the destination account must not be in a session.

transfer amount destination
transfer - command
amount – amount that you want to transfer
destination – account id of the account to transfer funds to.

Structures - The main structure is the bank structure. This contains an array of bank accounts (20), an int and a mutex. The mutex will be used to lock the whole bank structure.

The account structure has the attributes that belong to the bank. Account structure will represent an individual account. It contains its own mutex to protect the integrity of the structure. Two flag ints (exists & session_flag)as well as the accountName (char *) and balance (float).

Memory Integrity - The shared accounts are each protected by their own mutex. The account mutex is locked whenever a client has been served the account. It is then unlocked when the client logs off. The whole bank structure is protected by its own mutex as well. This mutex is locked whenever and account must be created as it must make sure the account ID does not already exists.

Rony Vargas
155005725
CS214 - Systems Programming
Spring 2015

Complexity – The accounts are on an array inside the bank structure. Therefore lookup time is at worst n, where n is the size of the array. It is the same complexity with inserting.

Memory usage – There is no dynamically allotted memory (for the bank struct) as there is only one bank structure that contains all the sub account structures. This could have been improved by dynamically allocating the memory for the accounts that aren't currently active. In this case I opted against dynamic memory as the there was no need for the struct to be able to dynamically change size.