CS331: Programming Language Lab

Assignment II: Advance Java Concurrent Programming

Soft deadline: 11.55 PM IST, 09th Feb 2021 (Hard deadline of MS team: submission allowed up to 11.55PM 10 Feb 2021)

Design and implement a Java multithreaded program to simulate a bank money transaction system for up to 10^6 users. Name of the bank is Guwahati National Bank (GNB) and the bank have 10 branches at different location of India and each branch have 10 updaters. Initially, 10^4 users will be there for each branch with a random amount of money in their accounts**. Every updater of the GNB can be modeled as a separate thread.**

- The updater gets a request to Cash Deposit, Cash withdrawal (withdrawal amount should less than amount money in the account).
- The updater gets a request to Transfer Money from one customer account to another customer account. The source account and destination account may be in different branches of GNB.
- The updater gets a request to add a customer with some initial money in that account, added customer will be in the **updater branch** of GNB.
- The updater gets a request to delete a customer from the system (or close the account of a user of the GNBs from **any branch**).
- The updater gets request to transfer customer account from **one branch to another branch** of GNB.

Assume the probability of getting cash deposit, cash withdrawal, money transfer, add a customer, delete a customer, transfer a customer to updater are 0.33, 0.33, 0.33, 0.003, 0.003, and 0.004 respectively.

Suppose information about all the customer accounts of the GNB is maintained by **an array of linked lists**. Each linked list represent (or hold data of a) branch of GNB, so there are 10 linked list and the array of linked list is maintained by a hash data structure. Every customer account number is represented by 10 digits and the first digit of the customer account number identify the branches of GNB.

You to be used linked list and hash should be thread-safe, and throughput should be high. You are allowed to use any inbuilt data structure, locking protocol, synchronized functions for the same.

Simulate up to 10^6 transactions per updater to test the correctness of your implementation. Report the execution time of your simulation program.

Submission Procedure:

- Upload your assignments code in the compressed folder (tgx/zip/gz) to MS team Grp_PLLab-CS331-2021 before the deadline.
- **Please embed comments, how to run and required inputs properly in the code, or a separate readme file**
- Source code will be checked for plagiarism, which can detect variable/function name change, minor structure change (while loop to for loop, vice versa), code displacement/repositioning**.**
- **Plagiarism case leads to F grades for both source candidate and destination candidate**. Make your code different from the internet code if available freely.