Student: Eduardo de Oliveira Castro
Blazer ID: edc
Date: 04/08/2015

Report for lab 13

Problem)

Fix the problem on the provided classes that implements a Bank Account using threads. The deposit/withdraw methods are not working correctly.

Solution)

The fix for the problem was pretty easy, it was only necessary to add lockers at the code. Every time that a method is called the first thing that is made is activating a lock for that method in order to prevent other to go over it. If the algorithm for that method is done as expected(the try condition passes) so this lock is released for another method waiting for it. This is done for all the methods that returns something on the BankAccount class.

But one extra condition happens when we want to withdraw money an the amount requested is bigger than the balance, for this case I used the Condition class that will just wait for a signal until more money is deposited on this account. Overtime that a deposit is made the deposit method sends a signal for all the conditions, so they check if the balance is still lower, if it is so the condition keeps waiting, if now it is bigger so the code goes ahead and the withdraw is made before anything is made. After this that lock is released.

Conclusion)

Threads are very important for programming languages since it is the feature that possibilities us to do a lot of different procedures at the same time instead of doing it in a “procedure” way. We can receive orders from a lot of different places and the code can manage all the requests in order to provide results for all of them. This is important for distributed systems, per example.