**CSCI 310 Group Project**

Banker’s Algorithm

**Team Member:**

Lucas Reller

Jeremy Nelson-Stalmer

Charline Rouamba

**Account:**

cs310126

**Date:**

November, 27th 2017

**Overview**

The assignment is about implementing a banker algorithm using thread and synchronization. To solve this problem, we use three classes in our program:

* Main.java
* Bank.java
* Client.java

Main retrieves the user input and run the highest-level controls of the program. Bank controls the resources used by the threads. Client holds the actual threads and run() method. These classes will be explained in more detail in the section titled full documentation.

**How to Run**

To run the program, the user must enter on the terminal this following commands:

**javac \*.java**

**java Main**

Then, the user would be prompted to enter a number between 1 and 10 for the number of bank resources and the number of clients. Then, the user has to wait for the program to compile and prints the output.

**Full Documentation**

Main:

Main is the simplest of the three classes. It retrieves the users’ inputs— the number of bank resources and the number of clients— and verifies that it is valid. From there it creates the necessary Bank and Client objects and makes the function calls to begin the execution of the program.

Bank:

Bank contains the functions that request and release resources from the bank. The request function will make a request and check if the request is granted. If it is not granted, it will be put in the waiting list. Then, if the request is granted, the program will check whether it is on the safe state, which means there is no deadlock. If yes, it will grant the request, otherwise, it will go to the waiting list. The release function releases the resources when they are available. After a customer request is met, the function removes the resources from the allocation matrix, and pushes the recourses back in the matrix. Both of theses methods are synchronized and notifyAll() a the end of their execution.

Client:

Client extends thread and holds the run() method. Run sets a random amount of resources to request then attempts to run its thread. It will also cause a sleep for 1-5 seconds to simulate time being consumed. Finally, each thread will release its resources when it is complete.

**Testing and Outputs**

**Test 1:**

Please input the number of bank resources (between 1-10): 9

Please input the number of clients (between 1-10): 5

Printing Initial/Final Available:

[20 17 9 12 1 5 15 17 18]

Printing Allocation:

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

Printing Maximum Amounts:

[18 6 0 5 1 2 15 6 16]

[10 7 6 2 0 3 6 14 16]

[7 3 2 12 0 2 5 11 7]

[17 4 7 0 1 0 2 17 6]

[1 9 2 2 1 0 11 17 1]

Client 0 making request:

[15 0 0 5 1 1 7 1 2]

Safe Seq:

[0 0 0 0 0]

Printing Allocation:

[15 0 0 5 1 1 7 1 2]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

Client 0 request 0 granted

Client 4 making request:

[1 3 1 1 1 0 8 2 0]

Not enough resources

Client 3 making request:

[17 4 5 0 1 0 0 4 4]

Not enough resources

Client 2 making request:

[4 0 0 3 0 2 4 2 6]

Client 2 must wait

Client 1 making request:

[5 5 0 2 0 2 5 0 9]

Client 1 must wait

Client 0 making request:

[0 4 0 0 0 0 3 2 6]

Safe Seq:

[0 0 0 0 0]

Printing Allocation:

[15 4 0 5 1 1 10 3 8]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

Client 0 request 1 granted

Client 2 making request:

[7 3 0 2 0 2 0 10 0]

Not enough resources

Client 0 making request:

[3 2 0 0 0 0 0 3 0]

Safe Seq:

[0 0 0 0 0]

Printing Allocation:

[18 6 0 5 1 1 10 6 8]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

Client 0 request 2 granted

Client 3 making request:

[17 4 3 0 1 0 0 1 3]

Not enough resources

Client 1 making request:

[4 5 3 1 0 0 6 7 4]

Not enough resources

Client 4 making request:

[0 7 0 1 0 0 10 11 0]

Not enough resources

Client 0 releasing resources:

[18 6 0 5 1 1 10 6 8]

Client 1 making request:

[8 3 0 0 0 3 6 6 3]

Safe Seq:

[0 0 0 1 0]

Printing Allocation:

[0 0 0 0 0 0 0 0 0]

[8 3 0 0 0 3 6 6 3]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

Client 1 request 2 granted

Client 2 making request:

[4 3 1 5 0 1 0 2 7]

Safe Seq:

[0 0 0 1 2]

Printing Allocation:

[0 0 0 0 0 0 0 0 0]

[8 3 0 0 0 3 6 6 3]

[4 3 1 5 0 1 0 2 7]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

Client 2 request 2 granted

Client 2 releasing resources:

[4 3 1 5 0 1 0 2 7]

Client 4 making request:

[0 8 1 0 0 0 2 0 0]

Client 4 must wait

Client 3 making request:

[16 4 2 0 1 0 2 14 1]

Not enough resources

Client 1 releasing resources:

[8 3 0 0 0 3 6 6 3]

Client 4 releasing resources:

[0 0 0 0 0 0 0 0 0]

Client 3 releasing resources:

[0 0 0 0 0 0 0 0 0]

Final Available Vector:

Printing Initial/Final Available:

[20 17 9 12 1 5 15 17 18]

Final Allocation Matrix:

Printing Allocation:

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0]

**Test 2:**

Please input the number of bank resources (between 1-10): 10

Please input the number of clients (between 1-10): 10

Printing Initial/Final Available:

[16 20 5 1 16 12 7 3 15 10]

Printing Allocation:

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

Printing Maximum Amounts:

[4 6 3 1 16 5 1 0 14 7]

[6 15 4 0 2 7 0 2 10 10]

[6 6 2 0 11 12 3 2 8 5]

[12 6 5 1 11 1 7 0 8 9]

[4 11 1 0 5 11 0 0 11 5]

[2 11 0 1 1 2 0 1 7 4]

[16 14 2 0 12 6 2 0 3 9]

[16 12 3 1 14 3 7 3 2 8]

[5 9 3 1 10 2 1 2 9 9]

[9 6 1 1 10 2 6 2 0 5]

Client 0 making request:

[4 2 2 1 14 3 0 0 6 3]

Safe Seq:

[0 0 0 0 0 0 0 0 0 0]

Printing Allocation:

[4 2 2 1 14 3 0 0 6 3]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

Client 0 request 0 granted

Client 9 making request:

[3 2 0 1 7 0 3 1 0 2]

Not enough resources

Client 8 making request:

[4 3 1 0 8 1 1 0 9 4]

Not enough resources

Client 7 making request:

[5 8 2 1 3 0 4 3 0 4]

Not enough resources

Client 6 making request:

[0 13 0 0 11 4 2 0 3 0]

Not enough resources

Client 5 making request:

[1 0 0 1 1 1 0 0 0 3]

Not enough resources

Client 4 making request:

[3 2 0 0 1 8 0 0 6 2]

Client 4 must wait

Client 3 making request:

[11 3 2 0 0 0 4 0 1 0]

Client 3 must wait

Client 2 making request:

[1 3 2 0 9 6 1 0 2 3]

Not enough resources

Client 1 making request:

[6 14 4 0 0 7 0 1 4 5]

Not enough resources

Client 0 making request:

[0 3 1 0 2 0 1 0 2 0]

Safe Seq:

[0 0 0 0 0 0 0 0 0 0]

Printing Allocation:

[4 5 3 1 16 3 1 0 8 3]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

Client 0 request 1 granted

Client 0 making request:

[0 0 0 0 0 0 0 0 0 1]

Safe Seq:

[0 0 0 0 0 0 0 0 0 0]

Printing Allocation:

[4 5 3 1 16 3 1 0 8 4]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

Client 0 request 2 granted

Client 1 making request:

[3 6 0 0 0 4 0 2 3 8]

Not enough resources

Client 8 making request:

[5 8 2 0 8 0 1 1 4 8]

Not enough resources

Client 4 making request:

[0 11 1 0 1 5 0 0 10 5]

Not enough resources

Client 3 making request:

[10 5 1 1 8 0 6 0 4 0]

Not enough resources

Client 6 making request:

[5 4 0 0 1 6 1 0 0 3]

Not enough resources

Client 2 making request:

[0 1 0 0 5 10 1 2 1 0]

Not enough resources

Client 7 making request:

[0 0 2 0 11 1 2 3 1 6]

Not enough resources

Client 9 making request:

[3 2 0 1 2 0 2 1 0 1]

Not enough resources

Client 5 making request:

[0 11 0 1 0 0 0 0 7 3]

Not enough resources

Client 0 releasing resources:

[4 5 3 1 16 3 1 0 8 4]

Client 7 making request:

[15 6 0 0 8 0 2 0 2 0]

Safe Seq:

[0 0 0 7 0 0 0 0 0 0]

Printing Allocation:

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[15 6 0 0 8 0 2 0 2 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

Client 7 request 2 granted

Client 1 making request:

[3 14 0 0 2 3 0 2 3 3]

Not enough resources

Client 9 making request:

[4 5 1 1 2 0 4 1 0 2]

Not enough resources

Client 3 making request:

[9 4 4 1 2 0 1 0 7 5]

Not enough resources

Client 5 making request:

[0 1 0 0 0 1 0 0 2 4]

Client 5 must wait

Client 6 making request:

[0 6 2 0 3 0 0 0 2 6]

Client 6 must wait

Client 4 making request:

[3 5 0 0 4 5 0 0 7 5]

Not enough resources

Client 2 making request:

[4 1 1 0 2 6 3 2 5 4]

Not enough resources

Client 8 making request:

[1 9 0 0 0 2 0 0 8 4]

Client 8 must wait

Client 7 releasing resources:

[15 6 0 0 8 0 2 0 2 0]

Client 9 releasing resources:

[0 0 0 0 0 0 0 0 0 0]

Client 8 releasing resources:

[0 0 0 0 0 0 0 0 0 0]

Client 1 releasing resources:

[0 0 0 0 0 0 0 0 0 0]

Client 5 releasing resources:

[0 0 0 0 0 0 0 0 0 0]

Client 4 releasing resources:

[0 0 0 0 0 0 0 0 0 0]

Client 3 releasing resources:

[0 0 0 0 0 0 0 0 0 0]

Client 2 releasing resources:

[0 0 0 0 0 0 0 0 0 0]

Client 6 releasing resources:

[0 0 0 0 0 0 0 0 0 0]

Final Available Vector:

Printing Initial/Final Available:

[16 20 5 1 16 12 7 3 15 10]

Final Allocation Matrix:

Printing Allocation:

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

**Test 3:**

Please input the number of bank resources (between 1-10): 10

Please input the number of clients (between 1-10): 9

Printing Initial/Final Available:

[13 0 19 17 0 18 4 5 2 20]

Printing Allocation:

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

Printing Maximum Amounts:

[5 0 18 0 0 2 1 0 0 18]

[11 0 2 5 0 15 1 2 2 20]

[7 0 0 12 0 3 2 1 1 1]

[7 0 6 13 0 1 3 3 1 1]

[13 0 7 4 0 10 1 3 2 1]

[5 0 9 8 0 2 3 3 2 1]

[13 0 12 4 0 1 4 1 1 17]

[12 0 6 8 0 17 4 0 1 7]

[13 0 4 3 0 7 0 2 0 10]

Client 0 making request:

[1 0 15 0 0 0 0 0 0 11]

Safe Seq:

[0 0 0 0 0 0 0 0 0]

Printing Allocation:

[1 0 15 0 0 0 0 0 0 11]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

Client 0 request 0 granted

Client 8 making request:

[1 0 3 2 0 2 0 2 0 1]

Client 8 must wait

Client 7 making request:

[5 0 0 1 0 14 3 0 1 4]

Client 7 must wait

Client 4 making request:

[6 0 3 3 0 5 1 1 2 1]

Client 4 must wait

Client 6 making request:

[6 0 1 1 0 0 1 0 0 17]

Not enough resources

Client 5 making request:

[4 0 0 2 0 2 2 2 1 0]

Client 5 must wait

Client 3 making request:

[4 0 4 2 0 0 1 1 0 1]

Client 3 must wait

Client 2 making request:

[7 0 0 3 0 1 2 1 0 1]

Safe Seq:

[0 2 0 0 0 0 0 0 0]

Printing Allocation:

[1 0 15 0 0 0 0 0 0 11]

[0 0 0 0 0 0 0 0 0 0]

[7 0 0 3 0 1 2 1 0 1]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

Client 2 request 0 granted

Client 1 making request:

[0 0 2 0 0 14 0 0 1 7]

Client 1 must wait

Client 7 making request:

[11 0 5 3 0 4 1 0 0 6]

Not enough resources

Client 4 making request:

[10 0 6 1 0 4 1 0 1 1]

Not enough resources

Client 6 making request:

[0 0 5 1 0 0 0 0 1 7]

Not enough resources

Client 2 making request:

[0 0 0 8 0 1 0 0 1 0]

Safe Seq:

[0 2 2 0 0 0 0 0 0]

Printing Allocation:

[1 0 15 0 0 0 0 0 0 11]

[0 0 0 0 0 0 0 0 0 0]

[7 0 0 11 0 2 2 1 1 1]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

Client 2 request 1 granted

Client 3 making request:

[4 0 3 13 0 0 1 0 0 0]

Not enough resources

Client 5 making request:

[0 0 1 5 0 0 0 0 1 0]

Client 5 must wait

Client 0 making request:

[0 0 0 0 0 1 1 0 0 4]

Safe Seq:

[0 2 2 0 0 0 0 0 0]

Printing Allocation:

[1 0 15 0 0 1 1 0 0 15]

[0 0 0 0 0 0 0 0 0 0]

[7 0 0 11 0 2 2 1 1 1]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

Client 0 request 1 granted

Client 8 making request:

[11 0 0 1 0 1 0 2 0 2]

Not enough resources

Client 7 making request:

[6 0 2 3 0 0 2 0 1 5]

Not enough resources

Client 3 making request:

[0 0 3 5 0 1 2 2 1 0]

Not enough resources

Client 5 making request:

[2 0 2 0 0 1 0 0 1 1]

Client 5 must wait

Client 1 making request:

[9 0 1 1 0 0 1 1 1 18]

Not enough resources

Client 2 making request:

[0 0 0 1 0 1 0 0 0 0]

Safe Seq:

[0 2 2 0 2 0 0 0 0]

Printing Allocation:

[1 0 15 0 0 1 1 0 0 15]

[0 0 0 0 0 0 0 0 0 0]

[7 0 0 12 0 3 2 1 1 1]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

Client 2 request 2 granted

Client 6 making request:

[1 0 7 3 0 0 2 1 1 11]

Not enough resources

Client 3 releasing resources:

[0 0 0 0 0 0 0 0 0 0]

Client 4 making request:

[8 0 7 1 0 3 1 0 2 1]

Not enough resources

Client 7 releasing resources:

[0 0 0 0 0 0 0 0 0 0]

Client 8 making request:

[4 0 3 0 0 7 0 2 0 4]

Client 8 must wait

Client 0 making request:

[2 0 3 0 0 0 0 0 0 2]

Safe Seq:

[0 2 2 0 2 0 0 0 0]

Printing Allocation:

[3 0 18 0 0 1 1 0 0 17]

[0 0 0 0 0 0 0 0 0 0]

[7 0 0 12 0 3 2 1 1 1]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

Client 0 request 2 granted

Client 2 releasing resources:

[7 0 0 12 0 3 2 1 1 1]

Client 1 making request:

[8 0 2 5 0 3 1 1 1 6]

Not enough resources

Client 6 releasing resources:

[0 0 0 0 0 0 0 0 0 0]

Client 8 releasing resources:

[0 0 0 0 0 0 0 0 0 0]

Client 5 releasing resources:

[0 0 0 0 0 0 0 0 0 0]

Client 4 releasing resources:

[0 0 0 0 0 0 0 0 0 0]

Client 1 releasing resources:

[0 0 0 0 0 0 0 0 0 0]

Client 0 releasing resources:

[3 0 18 0 0 1 1 0 0 17]

Final Available Vector:

Printing Initial/Final Available:

[13 0 19 17 0 18 4 5 2 20]

Final Allocation Matrix:

Printing Allocation:

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]

[0 0 0 0 0 0 0 0 0 0]