Phuong Ho

011056693

CECS277

Lab 7

**Question 1:**

import java.util.concurrent.locks.Condition;

import java.util.concurrent.locks.Lock;

import java.util.concurrent.locks.ReentrantLock;

public class BankAccount {

public static final double MAX\_BALANCE = 100000;

private double balance;

private Lock balanceChangeLock;

private Condition sufficientFundsCondition;

private Condition lessThanMaxBalanceCondition;

public BankAccount(){

balance = 0;

balanceChangeLock = new ReentrantLock();

sufficientFundsCondition = balanceChangeLock.newCondition();

lessThanMaxBalanceCondition = balanceChangeLock.newCondition();

}

public void deposit(double amount) throws InterruptedException{

balanceChangeLock.lock();

try{

while (balance + amount > MAX\_BALANCE)

lessThanMaxBalanceCondition.await();

System.out.print("Depositting " + amount);

double newBalance = balance + amount;

System.out.println(", new balance is " + newBalance);

balance = newBalance;

sufficientFundsCondition.signalAll();

}

finally{

balanceChangeLock.unlock();

}

}

public void withdraw(double amount) throws InterruptedException{

balanceChangeLock.lock();

try{

while (balance < amount)

lessThanMaxBalanceCondition.await();

System.out.print("Withdrawing " + amount);

double newBalance = balance - amount;

System.out.println(", new balance is " + newBalance);

balance = newBalance;

lessThanMaxBalanceCondition.signalAll();

}

finally{

balanceChangeLock.unlock();

}

}

public double getBalance(){

return balance;

}

}

**public** **class** DepositRunnable **implements** Runnable{

**private** **static** **final** **int** *DELAY* = 1;

**private** BankAccount account;

**private** **double** amount;

**private** **int** count;

**public** DepositRunnable(BankAccount anAccount, **double** anAmount, **int** aCount){

account = anAccount;

amount = anAmount;

count = aCount;

}

**public** **void** run(){

**try** {

**for** (**int** i = 1; i <= count; i++){

account.deposit(amount);

Thread.*sleep*(*DELAY*);

}

}

**catch** (InterruptedException exception){}

}

}

**public** **class** WithdrawRunnable **implements** Runnable{

**private** **static** **final** **int** *DELAY* = 1;

**private** BankAccount account;

**private** **double** amount;

**private** **int** count;

**public** WithdrawRunnable(BankAccount anAccount, **double** anAmount, **int** aCount){

account = anAccount;

amount = anAmount;

count = aCount;

}

**public** **void** run(){

**try** {

**for** (**int** i = 1; i <= count; i++){

account.withdraw(amount);

Thread.*sleep*(*DELAY*);

}

}

**catch** (InterruptedException exception){}

}

}

**public** **class** BankAccountThreadRunner {

**public** **static** **void** main(String[] args){

BankAccount account = **new** BankAccount();

**final** **double** AMOUNT = 10000;

**final** **int** REPETITIONS = 10;

**final** **int** DEPOSIT\_THREADS = 10;

**final** **int** WITHDRAW\_THREADS = 2;

**for** (**int** i = 0; i < DEPOSIT\_THREADS; i++){

DepositRunnable d = **new** DepositRunnable(account, AMOUNT, REPETITIONS);

Thread t = **new** Thread(d);

t.start();

}

**for** (**int** i = 0; i < WITHDRAW\_THREADS; i++){

WithdrawRunnable d = **new** WithdrawRunnable(account, AMOUNT, REPETITIONS \* DEPOSIT\_THREADS);

Thread t = **new** Thread(d);

t.start();

}

}

}

Output:

Depositting 10000.0, new balance is 10000.0

Depositting 10000.0, new balance is 20000.0

Depositting 10000.0, new balance is 30000.0

Depositting 10000.0, new balance is 40000.0

Depositting 10000.0, new balance is 50000.0

Withdrawing 10000.0, new balance is 40000.0

Depositting 10000.0, new balance is 50000.0

Depositting 10000.0, new balance is 60000.0

Depositting 10000.0, new balance is 70000.0

Depositting 10000.0, new balance is 80000.0

Depositting 10000.0, new balance is 90000.0

Withdrawing 10000.0, new balance is 80000.0

Depositting 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Withdrawing 10000.0, new balance is 80000.0

Depositting 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Withdrawing 10000.0, new balance is 80000.0

Depositting 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Withdrawing 10000.0, new balance is 80000.0

Depositting 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Withdrawing 10000.0, new balance is 80000.0

Depositting 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Withdrawing 10000.0, new balance is 80000.0

Depositting 10000.0, new balance is 90000.0

Withdrawing 10000.0, new balance is 80000.0

Depositting 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Withdrawing 10000.0, new balance is 80000.0

Withdrawing 10000.0, new balance is 70000.0

Depositting 10000.0, new balance is 80000.0

Depositting 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Withdrawing 10000.0, new balance is 80000.0

Depositting 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Depositting 10000.0, new balance is 100000.0

Withdrawing 10000.0, new balance is 90000.0

Withdrawing 10000.0, new balance is 80000.0

Withdrawing 10000.0, new balance is 70000.0

Withdrawing 10000.0, new balance is 60000.0

Withdrawing 10000.0, new balance is 50000.0

Withdrawing 10000.0, new balance is 40000.0

Withdrawing 10000.0, new balance is 30000.0

Withdrawing 10000.0, new balance is 20000.0

Withdrawing 10000.0, new balance is 10000.0

Withdrawing 10000.0, new balance is 0.0

**Question 2:**

**import** java.io.IOException;

**import** java.io.InputStream;

**import** java.io.OutputStream;

**import** java.io.PrintWriter;

**import** java.net.Socket;

**import** java.util.Scanner;

**public** **class** NumberClient {

**public** **static** **void** main(String[] args) **throws** IOException {

**final** **int** PORT = 8899;

Socket s;

**while** (**true**){

Scanner console = **new** Scanner(System.*in*);

**int** n;

System.*out*.print("\nPlease enter a number: ");

n = console.nextInt();

s = **new** Socket("localHost", PORT);

InputStream inStream = s.getInputStream();

OutputStream outStream = s.getOutputStream();

Scanner in = **new** Scanner(inStream);

PrintWriter out = **new** PrintWriter(outStream);

out.println(n);

out.flush();

**double** response = in.nextDouble();

System.*out*.println("The squareroot of " + n + " is: " + response);

}

}

}

import java.io.IOException;

import java.net.ServerSocket;

import java.net.Socket;

public class SquareRootServer {

public static void main(String[] args) throws IOException {

final int PORT = 8899;

ServerSocket server = new ServerSocket(PORT);

System.out.println("Waiting for clients to connect...");

while(true){

Socket s = server.accept();

SquareRootService service = new SquareRootService(s);

service.doService();

s.close();

}

}

}

**import** java.io.IOException;

**import** java.io.InputStream;

**import** java.io.OutputStream;

**import** java.io.PrintWriter;

**import** java.net.Socket;

**import** java.util.Scanner;

**public** **class** SquareRootService {

**private** Socket s;

**public** SquareRootService(Socket aSocket){

s = aSocket;

}

**public** **void** doService() **throws** IOException{

InputStream ip = s.getInputStream();

Scanner in = **new** Scanner(ip);

OutputStream op = s.getOutputStream();

PrintWriter out = **new** PrintWriter(op);

**double** response = Math.*sqrt*(in.nextDouble());

System.*out*.println("Sending: " + response);

out.println(response);

out.flush();

}

}



