# Parallel and MultiThreaded Programming

# CSYE 7215

# Homework 2

# Due: January 25, 2020

Put all your java, compiled class files and documentation into a zip file named Homework2.zip and submit it via the dropbox on the blackboard before the END of due date. Put your name on all .java files. There will be a short quiz on this assignment.

1. Your program HelloWorld references three objects, Class1, Class2, and Class3. What happens when you start your program? Describe your answers in terms of Process, JVM, JRE, ClassLoader, Stack Thread, Stack Frame, how does your program gets started?

2. Define a Student class with instance variables name, id, midterm, project, and final-exam. Name is a string whereas others are all integers. Also add a static variable nextId which is an integer and statically initialized to 1. In each of them, the id should be assigned to the next available id given by nextId. The default constructor should set the name of the student object to “StudentX” where X is the next id.

a) Your program is to create 32 Student Threads each to be identified with name-Thread-nextId. The default constructor for each thread calls a method to randomly generate grades for midterm, project, and final-exam ranging between 50 to 100 inclusive. You need to consider 1 second wait-time between each score generation for midterm, project, and final-exam. Each student thread writes the grade scores to “Grades” file in this format: name, nextId, ThreadId, midterm, project, finalExam. All student threads share this file and you need to protect it.

b) Create GraderThread that checks “Grades” file periodically up to 30 seconds to retrieve submitted grades by all student threads. How do you protect the file? The GraderThread reads the file (format described above) and validates name, id, threadId, scores for all 25 student threads submitted scores. For any missing grade, the student will receives zero score. The GraderThread does calculateGrade() (30% midterm + 30% project + 40% final) and returns a letter grade like “A”, “B”, “C”, “D” or “F”, based on the overall score.

c) In your StudentThread and GraderThread, you wrote several methods to handle processing of various functions in your program. Provide details on stack frames as it relates to local variable array, operand stack, and constant pool. Note: you need to show an example with data.

d) Create GradesDriver class to create 32 Student Threads and GraderThread to test your program.

Notes: You need to consider a number of protection mechanisms to protect the “Grades” file to avoid threads over-stepping each other. You need to think about what data structures to use to hold the FinalGrade (ie: use HashMap for key/value as threadId/FinalGrade). You need to think about how GraderThread to notify each student with finalGrade. Do you want to update the file with new column “FinalGrade” and have student threads get the final grade by reading the file? How does that going to work? You need to think how this mechanism can work. OR do you want to do another Wait/Notify where the GraderThread will be the notifier to each student thread? All in all, you need to be creative to solve this problem.

3. Does Thread synchronization works correctly with the following code? Why or

Why not? If Not, how do you fix it?

//example of java synchronized method

class Table{

synchronized void printTable(int n){//synchronized method

for(int i=1;i<=5;i++){

System.out.println(n\*i);

try{

Thread.sleep(400);

}catch(Exception e){System.out.println(e);}

}

}

}

class MyThread1 extends Thread{

Table t;

MyThread1(Table t){

this.t=t;

}

public void run(){

t.printTable(5);

}

}

class MyThread2 extends Thread{

Table t;

MyThread2(Table t){

this.t=t;

}

public void run(){

t.printTable(100);

} }

public class TestSynchronization2{

public static void main(String args[]){

Table obj1 = new Table();//only one object

Table obj2 = new Table();//only one object

MyThread1 t1=new MyThread1(obj1);

MyThread2 t2=new MyThread2(obj2);

t1.start(); t2.start(); }}

4. Consider the following diagram, Explain why there is race condition?

