Problem Set 7

(10 Points) Cohort Question 1:

Given WorkerThread.java which performs operations on a given Map, design an experiment to show the performance difference between Collections.synchronizedMap and ConcurrrentHashMap.

(10 Points) Cohort Question 2:

Analyze class GDesktopProb.java which implements the skeleton of Google Desktop, and fix potential problems with BlockingQueue.

(10 Points) Cohort Question 3:

Use semaphore to implement a BoundedHashSet. The class skeleton is given below. Hint: you can initialize the semaphore to the desired maximum size of the set.

(10 Points) Cohort Question 4:

Given an (large) array of strings (of grades), write a multi-threaded program, using CountDownLatch, to check whether the array contains 7 “F”. Stop all threads as soon as possible.

(10 Points) Cohort Question 5:

Fix the minor problem on the previous slide using ConcurrentMap.putIfAbsent(). Notice that you can’t apply client-side locking on ConcurrentMap.

(10 Points) Cohort Question 6:

Write a program to simulate the following. During exam, the examiner (a thread) who waits patiently for a known number of students (one thread for each student) to get ready; the exam starts when all students are ready, and at the same time for all. Afterwards, the examiner patiently waits for all students to hand in their scripts. Students are free to leave once they hand in the exam. Once the last student hands in the exam (or the time limit expires), the examiner stops waiting and leaves with the scripts.