Problem Set 6

(10 Points) Cohort Exercise 1:

Name your solution: MyStackThreadSafeComplete.java

(10 Points) Cohort Question 2:

Name your solution: FirstExampleFixed.java

(10 Points) Cohort Question 3:

Name your solution: SafeStack.java

(10 Points) Cohort Question 4:

Name your solution: TackerFixed.java

(10 Points) Cohort Question 5:

Name your solution: DiningPhilFixed1.java and DiningPhilFixed2.java

(10 Points) Cohort Question 6:

Name your solution: GDesktop.java

(10 Points) Cohort Question 7:

Name your solution: CacheV3.java

(20 Points) Homework Question 1:

As a rising star in a bank's IT department, you have been given the job of creating a new bank account class called SynchronizedAccount. This class must have methods to support the following operations: deposit, withdraw, and check balance. Each method should print a status message to the screen on completion. Also, the method for withdraw should return false and do nothing if there are insufficient funds. Because the latest system is multi-threaded, these methods must be designed so that the bookkeeping is consistent even if many threads are accessing a single account. Additionally, to maximize concurrency, SynchronizedAccount should be synchronized differently for read and write accesses. Any number of threads should simultaneously be able to check the balance on an account, but only one thread can deposit or withdraw at a time.

Name your solution: SynchronizedAccount.java