CSCI 201L Final – Written – SOLUTION
Spring 2015
12% of course grade

1. **Garbage Collection** – The garbage collector in Java helps with memory management. Java provides a method called `System.gc()` that programmers can invoke.
   a. Explain what the `System.gc()` method does. **(0.5%)**

System.gc() makes a suggestion to the JVM to perform a garbage collection. It is no guarantee that it does it since the garbage collector runs in a separate thread. Some instances of the JVM may increase the priority of the garbage collector thread when System.gc.() is invoked.

   b. Why does `System.gc()` not cause the garbage collector to be run immediately? **(0.5%)**

Since the garbage collector is running in a separate thread, it will not execute immediately, which would require preempting the currently executing thread from the CPU.

2. **Inner Classes** – Anonymous inner classes are very commonly used with event-driven programming in Java. There is no requirement to use them though.
   a. Give two reasons why using anonymous inner classes are beneficial. **(0.5% + 0.5%)**

1. The implementation of is in-line with the rest of the code, making it easier to read.
2. If you are only going to use a class one time, there is no need to create an explicit class that will only be instantiated once, decreasing memory usage.
3. With small classes, the code will be more maintainable.
4. The scope of an anonymous inner class is hidden from other methods or classes.

   b. Give one situation when using anonymous inner classes is not recommended. **(0.5%)**

1. If the class needs to be used more than once, an anonymous inner class would require copying and pasting code.
2. If an anonymous inner class is very large, the readability of the code will decrease.

3. **Networking Theory** – Given the following IP address and subnet mask, answer the following questions.

IP Address – 178.93.209.160
Subnet Mask – 255.255.255.224

IP – 1011 0010 0101 1101 1101 0001 1010 0000
Mask – 1111 1111 1111 1111 1111 1111 1110 0000

   a. What is the subnet address? Provide this in the dotted IP notation, not in binary. **(0.5%)**

178.93.209.160

   b. How many hosts can be on the subnetwork? **(0.5%)**

32 (though if they say 30, give them credit since the first host is the network address and the last host is the broadcast address)

   c. What is the range of IP addresses for this subnet? In other words, what is the starting IP address and ending IP address of this subnet? Provide this in the dotted IP notation, not in binary. **(1.0%)**

0.5% for the start address - 178.93.209.160
0.5% for the end address - 178.93.209.191

4. **JDBC –** To connect to a database in Java, we use a JDBC driver. Consider the differences between using a driver to connect instead of connecting ourselves via a Socket.

   a. Give one advantage to using a JDBC driver instead of a direct connection when communicating with a database. **(0.5%)**

1. A JDBC driver takes care of all of the standard database communication.
2. JDBC drivers allow us to pass SQL code directly to the database without being concerned with sockets and communication.
3. JDBC drivers reduce the amount of code we need to write to communicate with a database.
4. JDBC drivers abstract the networking from the database code.

   b. Give one reason why a programmer would want to use a database instead of files for storing data. **(0.5%)**

1. Fast lookups
2. Searching is already implemented for us
3. Format is standardized
4. Operating system independent

5. **Locks and Conditions –** Answer the following questions about locks and conditions.
    a. The type of lock we used in Java was called a ReentrantLock. Explain what a ReentrantLock is and how it is different from a lock that is no reentrant. **(1.0%)**

With a ReentrantLock, after a thread has acquired a lock, it will be able to enter other sections of code that require the same lock without having to acquire it again. Otherwise, this would cause deadlock.

    b. A condition must be associated with a lock, and before calling any methods on a condition, the lock associated with that condition must first be obtained. Explain why a lock must be obtained before a method on a condition associated with the lock can be called. **(1.0%)**

When the await() method on a condition is called, the lock associated with that condition is released. Before being able to signal threads that are waiting on a condition, the lock must be obtained because it doesn't make sense to signal on a condition associated with a lock without having the lock. Why would one thread be able to signal another thread waiting on a condition when it doesn't have anything to do with that condition or lock?

    c. Describe a problem that could arise if a method on a condition was able to be called without first having the lock associated with it. **(1.0%)**

This means that threads could wait on any condition that was created, regardless of whether they have obtained any locks. This could easily lead to deadlock.

6. **Semaphores –** In the factory code, we used semaphores for ensuring that only a specific number of resources could be acquired before a worker would have to wait for more resources. This seems like it is essentially a counter.
    a. Explain the difference between using a counter and using a semaphore for solving this problem. **(1.0%)**

Semaphores have the waiting mechanism already built in. They have an internal counter, but once that counter reaches 0 and another thread tries to obtain a permit, the thread will have to wait. A counter will not give us the same benefit unless we wanted to implement the semaphore functionality manually.

    b. Is it possible to use locks and conditions to get the same behavior as a semaphore? If so, explain how. If not, explain why not. **(1.0%)**

Locks and conditions could be used to implement semaphores. You could either have an array of locks, one for each permit, or an array of conditions based on the number of permits with one lock.

7. **Distributed Programming –** Write the code for connecting to an RMI server…just kidding. RMI, CORBA, and Web Services all allow remote procedure calls to be made from a client to a server. Answer the following questions:

   a. Give one advantage to using RMI over CORBA and Web Services. **(0.5%)**

If both the client and server are written in Java, RMI allows us to serialize Java objects to transmit back and forth.

   b. Give one advantage to using Web Services over RMI and CORBA. **(0.5%)**

Web Services do not require any additional ports to be open through a firewall since it operates on a web/application server.

   c. Give one advantage to using CORBA over RMI and Web Services. **(0.5%)**

CORBA is the oldest distributed paradigm of the three so there is more legacy code that uses CORBA over RMI and Web Services.

   d. Did you think the section on distributed programming was beneficial? Do you think I should keep that section in the course in the future? Why or why not? **(0.5% extra credit)**

Any answer here will earn full credit.