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%)*

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

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%)*

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

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

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%)

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%)

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

   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%)

   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%)

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%)

   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%)
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%)

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

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

   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)