Short Answer (4 Points)

1.) To spawn a new Thread of execution, we must call the ___start_____ method.

2.) A(n) __port number__ defines which networking application on a remote computer a socket is attempting to communicate with.

3.) A(n) ____blocking___ method call suspends the execution of the calling Thread until some external event occurs, allowing the Thread to continue.

4.) A(n) __race condition__ can occur when two Threads simultaneously access a shared resource.

True or False (5 Points - Please circle your answer)

5.) (True / False) - Only one Thread can be executing inside of a synchronized block on an object at any given time.

6.) (True / False) - The interrupt() method terminates a Thread.

Rationale: interrupt() only suggests that a Thread should terminate as soon as possible. It’s up to the programmer who writes the thread to determine the exact behavior when interrupt() is called.

7.) (True / False) - An object must be Serializable to be written to a stream.

Rationale: The class to which the object belongs must implement java.io.Serializable, or its sub-interface, java.io.Externizable. In either case, the object is-a Serializable.

8.) (True / False) - A Lock object can give a programmer greater control over protecting a critical section than a synchronized block because a Lock is bound by syntactical constraints.

Rationale: Synchronized blocks are constricted by syntactical constraints, not Lock objects.

9.) (True / False) - We can choose to write either human-readable or binary data over a network connection.

Rationale: A socket (our end of a network connection) gives us an java.io.InputStream and a java.io.OutputStream to read and write to the network connection, respectively. InputStreams and OutputStreams interact with data as bytes, or in binary format.

Short Answer (1 Point)

10.) Explain why it might be undesirable to use the deprecated stop() method to immediately stop the execution of a Thread.
Answer: The thread might need to clean up some resources that it is using, e.g., open streams or sockets that need to be closed in order to gracefully finish any work that was in the middle of being performed by the thread. Stopping a thread in its tracks might result in undefined behavior.