1. (10 pts) Below is code for the main loop of a network server application. Fill in the missing parts. You may assume that the WorkerThread class is a subclass of Thread. port is an int.

```java
try {
    _ServerSocket_ sock = new _ServerSocket_ (port);

    while (_true_)
    {
        _Socket_ conn = sock.accept ();
        WorkerThread worker = new WorkerThread (conn);
        worker._start_ ();
    }
} catch (java.io.IOException e) {
    ... handle exception ...
}
```

public class ServerSocket // some methods omitted
{
    ServerSocket();
    ServerSocket(int port);
    Socket accept();
    void bind(SocketAddress endpoint);
    void close();
    InetAddress getInetAddress();
    int getLocalPort();
    boolean isBound();
    boolean isClosed();
}

public class Socket // some methods omitted
{
    Socket()
    Socket(String host, int port);
    void bind(SocketAddress bindpoint);
    void close();
    InetAddress getInetAddress();
    InputStream getInputStream();
    InetAddress getLocalAddress();
    int getLocalPort();
    OutputStream getOutputStream();
    int getPort();
    boolean isBound();
    boolean isClosed();
    boolean isConnected();
}
public class Thread // some methods omitted
{
    void destroy()
    String getName()
    int getPriority()
    void interrupt()
    boolean isAlive()
    boolean isDaemon()
    boolean isInterrupted()
    void join()
    void join(long millis)
    void join(long millis, int nanos)
    void run()
    void setDaemon(boolean on)
    void setName(String name)
    void setPriority(int newPriority)
    static void sleep(long millis)
    void start()
    static void yield()
}

2. (5 pts) Match the type of thread to the best method to be used to stop the thread:

a. A network client that reads stock market data continuously from a remote server.

1

b. A thread that reads large image files into a graphical editor.

1

c. A thread (not a timer) that checks the contents of a log file every five minutes and alerts
   the user of new contents.

2

d. A thread that scrolls messages across the message bar at the bottom of a JFrame

3

Termination methods

1. Close the input/output stream or network connection that the thread is using.
2. Call the thread’s interrupt() method.
3. Set a termination variable that controls the thread’s inner loop.
4. Call the thread’s stop() method.
3. (5 pts) A thread may become non-runnable (but not dead) in three different ways. Indicate briefly, for each situation, what needs to happen for the thread to become runnable again.

a. The thread is asleep.

End of sleep time reached.

b. The thread is in a wait state.

Another thread calls notify on the object that the thread is waiting on.

c. The thread is blocked because it is attempting to call a synchronized method on an object that is in use by another thread.

The other thread completes its method call and unblocks object.