Discussion Section 11 (11/20/2008)

## 1. How many possible results? Some of them?

```
public class Test {
    public static void main(String[] args) throws
            InterruptedException {
        Thread thread = new MyThread();
        thread.start();
        System.out.println("A");
        System.out.println("C");
        thread.join();
    }

    static class MyThread extends Thread {
        public void run() {
            System.out.println("B");
            System.out.println("D");
        }
    }
}
```

## 2. How many possible results? Some of them?

```
public class Test {
    private static Object lock = new Object();

    public static void main(String[] args) throws
            InterruptedException {
        Thread thread = new MyThread();
        synchronized (lock) {
            thread.start();
            System.out.println("A");
            lock.wait();
        }
        System.out.println("C");
        thread.join();
    }

    static class MyThread extends Thread {
        public void run() {
            synchronized (lock) {
               System.out.println("B");
               lock.notify();
               System.out.println("D");
            }
        }
    }
}
```

## 3. Issues on Threads

### a. What threads can do?

**b. Difference between run() and start() in Thread?**

**c. I always see the same result, what’s the worry?**

**d. Tradeoff between thread-safe data structure and not thread-safe?**

| **Classes** | **Thread-Safeness** |
|---|---|
| Hashtable | ✓ |
| Vector | ✓ |
| HashMap, LinkedHashMap, TreeMap | |
| HashSet, LinkedHashSet, TreeSet | |
| ArrayList, LinkedList | |

(Watch out for “Note that this implementation is not synchronized” in JavaDoc.)

**e. How to use all these: synchronized, wait, notify, join, sleep, yield, interrupt?**

**f. Share data between threads?**

**g. Send/Receive signals between threads? (Isn’t polling a great idea?)**

**h. How to debug an exponential number of combinations?**

**i. Performance vs. maintainability tradeoff**