Elaboration „Thread 3 (Thread Race Condition)“

a) *Which problems could arise?*

/* in own words and more points, if the problem is explained with a possible outcome */

```
public class Counter {
/*********************/
    private int counter=0;

...
    public void incCounter() {
           try {
                  int temp = counter;
                  Thread.sleep((long) (Math.random() * 1000));
                  counter = temp + 1;
           } catch (Exception e) {…}
    }
}
public class Worker extends Thread {
/**********************************/
    private Counter counter=null;
    public Worker(Counter _counter) {
           counter = _counter;
    }
    public void run(){
           for (int i=0; i<5; i++) {
                  try {
                         System.out.println(getName() + " before " + counter.getCounter());
                         Thread.sleep((long)(Math.random()*1000));
                         counter.incCounter();
                         System.out.println(getName() + " after " + counter.getCounter());
                  }catch ...
           }
    }
}
}
public class ThreadMain {
/***********************/
    public static void main(String[] args) {
           Counter c = new Counter(20);
           Worker w1 = new Worker(c);
           Worker w2 = new Worker(c);
           w1.start();
           w2.start();
           System.out.println("Main...Worker started");
           try {
                  w1.join();
                  w2.join();
           } catch...
           System.out.println("Main...Worker finished");
           System.out.println("Main...counter=" + c.getCounter());
    }
}
```

b) *Why is the following code-fragment not secure?*

```
public class MyCounter {
   private static int counter;
   public synchronized void incCounter() {
      counter += 1;
   }
}
```

What could be the solution?

c) Why is the following code-fragment not secure?

```
/* moving object (changing permanently its position) is observed */

class GeoCoo {
 private float x,y;
 public GeoCoo (float x,float y) {
       this.x=x;
       this.y=y;
 }
 public void setGeoCoo (float x,float y) {
       this.x=x;
       this.y=y;
 }
 public GeoCoo getGeoCoo () {
       return new GeoCoo(x,y);
 }
}
```

What could be the solution?