Exercise 1

Object-oriented programming
8 Points

a) Name two possibilities to react to exceptions in Java. (2 points)

b) The following piece of code contains a logical mistake. Explain it and propose a solution. (2 points)

```java
public void doSomething() {
    try {
        someObject.process();
    } catch (Exception ex1) {
        throw new RuntimeException(ex1);
    } catch (IOException ex2) {
        System.err.println(ex2);
    }
}
```

c) Give a short and concise explanation for the concept of generic programming. Name two advantages of using generics in Java.

Exercise 2

Parallel programming
15 Points

a) Name and describe two problems that can arise when executing two threads concurrently.

b) Create a class Locker, which has two methods lock and unlock. Objects of this class are used to limit the number of threads that can enter a certain block of a program at the same time. The constructor of the class takes the maximum number of allowed threads. lock is called before the block is entered. If there are fewer threads than allowed already in the block, the current thread may enter the block. If not, it will be blocked and has to wait until another thread leaves the block. Upon leaving the
block, unlock must be called. The following piece of code demonstrates the usage of the Locker:

```java
class Test {
    private Locker locker = new Locker(3);
    public void doSomething() {
        locker.lock();
        try {
            // do something
        } finally {
            locker.unlock();
        }
    }
}
```

You must not use classes from the java.util.concurrent package (e.g. ReentrantLock). Instead you should use Java’s default synchronisation mechanisms. For simplification you must not for threads that call unlock without having called lock beforehand.