

# Advanced Aspects of Object-Oriented Programming (SS 2011)

## Practice Sheet 8

Date of Issue: 31.05.11  
Deadline: 07.06.11  
(until 10 a.m. as PDF via E-Mail)

### Exercise 1 Questions

- a) Add one or more questions to the document at <http://bit.ly/1Gbs2v>.

### Exercise 2 Introduction to JML

The *Java Modeling Language* allows to specify properties of Java software by using special annotations. The JML homepage (<http://www.jmlspecs.org>) provides tutorials, papers, and tools you can use to solve the following exercises.

- a) Summarize advantages of using a formal specification technique such as *JML* in comparison to informal approaches. Could you think of any drawbacks caused by the usage of formal specification techniques?
- b) Make yourself familiar with the conceptual framework provided by JML. Use the paper „Design by Contract with JML“ by Leavens and Cheon (<http://www.eecs.ucf.edu/~leavens/JML/jmldbc.pdf>) as a starting point.
- c) Download a version of the common JML tools (<http://www.eecs.ucf.edu/~leavens/JML/OldReleases/JML.5.5.tar.gz>), follow the included install instructions and make yourself familiar with their usage. Use the tools for the following exercises to check your specifications and to execute your annotated programs. *Note: The tools only support Java 1.4, so you cannot use generics and the syntax extensions of Java 5 and 6.*
- d) Specify the following class. Give pre- and postconditions for the constructor and methods and a non-trivial class and loop invariant.

```
class Container {
    int[] a;
    int n;

    Container( int[] input ){
        n = input.length;
        a = new int[n];
        System.arraycopy(input, 0, a, 0, n);
    }

    int extractMin() {
        int m = Integer.MAX_VALUE;
        int mindex = 0;
        for (int i = 0; i < n; i++) {
            if (a[i] < m) {
                mindex = i;
                m = a[i];
            }
        }
        n--;
        a[mindex] = a[n];
        return m;
    }
}
```

- e) It is obvious that the value `n` can only decrease over time. How can you specify this using JML?
- f) Specify the JDK class `java.io.ByteArrayInputStream` using *JML*.