J. Hoenicke
A. Nutz

24.5.2017
Please hand in your solution until
31.5.2017, via email to
`nutz@informatik.uni-freiburg.de`.

Tutorials for "Formal methods for Java"
Exercise sheet 5

Exercise 1: Prove safety of array accesses
Consider the file `Sort.java` from the lecture homepage. When run in ESC mode, OpenJML reports that array bounds might be violated in the method `sort`. Annotate the method with `requires` and `loop_invariant` annotations such that OpenJML(esc) reports no more errors (and thus has proven that no array is accessed outside of its bounds).

Exercise 2: Prove correctness
Consider the file `Max.java` from the lecture homepage.
First add `requires` and `loop_invariant` annotations such that OpenJML(esc) proves that the method `max` fulfills the given `ensures` clause.
The given ensures clause does not enforce that the result is in the input array. To fix this, add a second `ensures` clause to the method (you need an `\exists` quantifier). If necessary, adapt your annotations such that the proof still goes through.

Notes:

- Probably the easiest way to run OpenJML(esc) is via the webinterface on `http://rise4fun.com/OpenJMLESC`.
- If you want to run your OpenJML installation in ESC mode, you can do it with a command line like:

```
java -jar <dir>/openjml.jar -esc -exec <solver-executable> <file>.java
```

- You need to have an SMT solver installed. For a precompiled version of the Z3 SMT solver go to `https://z3.codeplex.com/releases`, click "planned" and chose a link suitable for your operating system.
- Warning: It is currently unclear if, and under which circumstances, the command-line version of OpenJML(esc) can be made to run and return sensible results. So maybe it works for you, then that's fine; otherwise switch to the webinterface.