J. Hoenicke
J. Christ

05.02.2013
Hand in solutions via email to
`christj@informatik.uni-freiburg.de`
until 12.02.2013 (only Java sources, KeY proofs, and PDFs accepted).
Paper submissions possible after the lecture.

Tutorials for "Formal methods for Java"
Exercise sheet 13

Note: The webstart version of KeY contains some GUI bugs when using Java 7. A working version can be downloaded from `http://www.key-project.org/download/releases/key165rc/KeY-1.6.5_cf8990d4ec6fef2d0a49662adb3ec509e023a0c3.tgz`. A webstart for this version is available from `http://www.key-project.org/download/releases/key165rc/webstart/KeY.jnlp`. Please use this version for all exercises on this sheet.

Exercise 1: Insertion Sort
On the webpage of the lecture you find a version of Insertion Sort that is fully annotated. Set the proof search strategy of KeY to

- Goal Chooser: "Default"
- Logical splitting: "Off"
- Loop treatment: "Invariant"
- Method treatment: "Expand"
- Quantifier treatment: "None"

This proof search strategy does not succeed to automatically prove total correctness even though the annotations are sufficient. Instead, this strategy leads to five goals.

(a) Give a description of the different loop invariants in prose. Instead of `i` use "the outer iterator", and instead of `j` use "the inner iterator". For example, the invariant `i >= 1 && i <= arr.length` can be described as "The outer iterator is always between 1 and the length of the array."

(b) Explain the goals, i.e., what is to be proven in each case. Since KeY is deterministic at this part you may enumerate the remaining goals.

(c) Use your knowledge from the previous exercise to prove the remaining goals.

Exercise 2: Insertion Sort for Empty Arrays
In the previous exercise we forced to array to be non-empty by adding the pre-condition `arr.length > 0`. If we remove this pre-condition the proof gets slightly more complicated.

(a) Identify the problems that might occur with empty arrays.

(b) How can we elegantly fix these problems with KeY without modifying the code or the annotations of the class. Of course, we remove the pre-condition `arr.length > 0`.