J. Hoenicke
J. Christ

30.10.2012
Hand in solutions via email to
christj@informatik.uni-freiburg.de
until 06.11.2012 (only Java sources and PDFs accepted) or at the beginning of the lecture.

Tutorials for "Formal methods for Java"
Exercise sheet 2

Exercise 1: Operational semantics

(a) Give rules for the operational semantics of the `?:` and the `!` operator, i. e., specify how to evaluate `e ? st1 : st2` and `!e`.

(b) Consider the following Java class:

```
class C {
  int x = 0;
  public int m(boolean b) {
    return (b = !b) ? x + 1 : x;
  }
}
```

Use the rules defining the operational semantics of Java to compute the result of the method call: `c.m(true)`. Assume that `c` is an instance of class `C` which has just been initialized.

Exercise 2: Loops with continue

Java provides the **continue** statement that when executed within a loop causes the execution of the loop to immediately return to the loop head. Execution is then continued as if the loop head would have been reached in normal execution. For simplicity, we assume every loop is labeled, and every **continue** statement is followed by a label, i.e., a **while** loop has the form $l : \mathit{while}(e)s$ where l is the label of the loop.
We can model **continue** statements by extending the flow component of program states:

$$\mathit{Flow} ::= \mathit{Norm} | \mathit{Ret} | \mathit{Exc}\langle\langle \mathit{Address} \rangle\rangle | \mathit{Continue}\langle\langle \mathit{Label} \rangle\rangle \ .$$

Use this extension to define the operational semantics of **continue** l statements and **while** loops with continues.
Hint: You only need to define one axiom and one rule.

Exercise 3: Operational equivalence

We say that two Java statements c_1 and c_2 are operationally equivalent if

$$\forall \mathit{flow}, \mathit{heap}, \mathit{lcl}, \mathit{flow}', \mathit{heap}', \mathit{lcl}'.\ (\mathit{flow}, \mathit{heap}, \mathit{lcl}) \xrightarrow{c_1} (\mathit{flow}', \mathit{heap}', \mathit{lcl}') \iff (\mathit{flow}, \mathit{heap}, \mathit{lcl}) \xrightarrow{c_2} (\mathit{flow}', \mathit{heap}', \mathit{lcl}')$$

Are the following pairs of Java statements operationally equivalent? Give a proof or a counter-example.

(a) $y = x\texttt{++;}$ and $y = x\texttt{; }x\texttt{++;}$, where x and y are local variables.

(b) $\textbf{if}(e)\, c\, \textbf{else}\, c$ and c,
where e is a boolean expression and c a statement.

(Bonus) Try to find a counterexample to the equivalence of $e_1 < e_2$ and $-e_1 > -e_2$ where e_1 and e_2 are integer-valued expressions. Although we did not present a rule for negation, less, and greater you should assume the Java semantics.