Semantics Exercises in Isabelle

Exercise Session 1

1. **Expression Optimisation (*)**
   Write a function that optimises arithmetic expressions by evaluating constants as in the lecture, but additionally evaluate addition with 0, e.g. $(3 + 2) + x + 0$ should become $5 + x$. Prove that your optimisation is semantics preserving.

   Additionally, do the same for boolean expressions, optimising conjunction with $True$ and $False$.

   You can find a skeleton Isabelle file for this exercise in `IMP/Fold_Exp.thy`.

2. **Extending the Language (*/**)**
   Extend the IMP language by the construct `REPEAT c UNTIL b`. Update all proofs that have been covered in the lecture so far (up to equivalence of small step and big step semantics).

Exercise Session 2

1. **Alternative typed expression evaluation (*)**
   Instead of an inductive definition, write `taval` in theory `IMP/Typies_Exp.thy` as a function from expression and state to value option. Do the same for boolean expressions. Prove that your definition is equivalent to the inductive definition from the lecture.

2. **Extending the Language (**)**
   Complete the `REPEAT c UNTIL b` language extension for the rest of the theories covered in the lecture, but leave out the second direction of the compiler proof.

More Exercises

1. **Constant folding (***)**
   Extend constant folding of arithmetic expressions into statements, in particular into variable assignments. Use a table to statically keep track of variables with known current value. Use this table in optimising arithmetic expression. Update the table appropriately when analysing statements. Prove that your transformation is semantics preserving, i.e. that the simplified statement is semantically equivalent to the original program. For an added challenge, extend constant folding and the proof to boolean expressions in statements.

2. **Extended Compiler (***)**
   Complete the second direction of the compiler proof for the `REPEAT c UNTIL b` language extension.

The Isabelle IMP theories are available from
http://www4.in.tum.de/~nipkow/MOD2011/IMP.zip