Assignment 2
Semantics, WS 2009/10

Prof. Dr. Gert Smolka, Dr. Jan Schwinghammer, Christian Doczkal
www.ps.uni-sb.de/courses/sem-ws09/

Hand in by 11.59am, Tuesday, November 3

You can find a Coq template for the exercises on the course web page.

This exercise sheet counts towards your exam qualification. Send your solutions in a file named `lastname.v` to `doczkal@ps.uni-sb.de`, and make sure that the entire file compiles without errors.

Exercise 2.1 (Inductive Subgoals) Consider the following statements and give the subgoals generated by the induction tactic. Do not forget the induction hypothesis. Make sure you can do this without using Coq.

a) $\forall X\,(xs : list\ X),\ length\ xs = length\,(rev\ xs)$ starting with $induction\ xs$

b) $\forall n\,m\,p : nat,\ n + (m + p) = m + (n + p)$ starting with $induction\ p$

c) $\forall n\,m\,p : nat,\ n + (m + p) = m + (n + p)$ starting with $induction\ n$

Exercise 2.2 (Primitive Recursion for Lists) In this exercise you will study primitive recursion for polymorhic lists, which is defined as follows.

```
Fixpoint foldl X Y (f: X->Y->Y) (a:Y) xs := match xs with
  | nil => a
  | x::xr => foldl X Y f (f x a) xr
end.
```

Solve the following problems:

a) Express *length* with *foldl* and prove the correctness.

b) Express *rev* with *foldl* and prove the correctness.

c) Express ++ with *foldl* and prove the correctness.

If you can do this without further help we are impressed. Otherwise we offer the following hints:

- For (a) and (b) one has to generalize the inductive claim so that the induction goes through.
- (c) follows from (b) with two standard lemmas ($xs ++ nil = xs$ and $rev(rev\ xs) = xs$).
- To make your job still easier, we provide a Coq file containing the necessary definitions and lemmas as well as the generalized claims for (a) and (b). Use the *ring_simplify* tactic for the arithmetic in (a). You load it with Require Import Arith.