Assignment 3
Semantics, WS 2011-2012

Prof. Dr. Gert Smolka, Dr. Chad Brown
www.ps.uni-saarland.de/courses/sem-ws11/

Exercise 3.1 Prove the following.

```
Goal forall X:Prop, ~X <-> ~~~X.
```

Exercise 3.2 Prove the following.

a)

```
Goal False <-> forall Z:Prop, Z.
```

b)

```
Goal forall X:Prop, ~X <-> forall Z:Prop, X -> Z.
```

c)

```
Goal forall X:Type, forall x y:X, x = y <-> forall p:X -> Prop, p x -> p y.
```

d)

```
Goal forall X Y:Prop, X /\ Y <-> forall Z:Prop, (X -> Y -> Z) -> Z.
```

e)

```
Goal forall X Y:Prop, X \/ Y <-> forall Z:Prop, (X -> Z) -> (Y -> Z) -> Z.
```

Exercise 3.3 Prove the following.

a)

```
Goal forall X Y:Prop, ~(X \/ Y) <-> ~X /\ ~Y.
```

b)

```
Goal forall X Y Z:Prop, (X \/ (Y /\ Z)) <-> (X \/ Y) /\ (X \/ Z).
```

Exercise 3.4 Prove the following. (This exercise may be tough.)

```
Goal (forall X:Prop, ~~X -> X) -> (forall X:Prop, X \/ ~X).
```

Exercise 3.5 Prove the following.

a)

```
Goal forall p:nat -> Prop, forall x:nat, p 0 -> (forall x:nat, p x -> p (S x)) -> p x.
```

b)

```
Goal forall X:Type, forall p: list X -> Prop, forall xs:list X, p nil ->
 (forall x:X, forall xs: list X, p xs -> p (x :: xs)) -> p xs.
```

Exercise 3.6 Extend the compiler correctness development with an operator for subtraction.

Exercise 3.7 (Challenge) Write a decompilation function that recovers an expression from the program it compiles to and prove the correctness of your function.

Exercise 3.8 Consider the following alternative definition of a compiler.

```
Fixpoint compile' (e : exp) : prog :=
  match e with
    | Const n => iConst n :: nil
    | Binop b e1 e2 => compile' e1 ++ compile' e2 ++ iBinop b :: nil
  end.
```

Consider the binary operators for addition (+), multiplication (*) and subtraction (-). What is the maximum set of these three operators for which this compiler is correct?