Functional Programming
Exercises Week 5
(for November 13, 2009)

1. Read Chapter 5 of the lecture notes.

2. Use the conventions to simplify the following λ-term

$$(\lambda x.(\lambda y.(\lambda z.(((x\ y)\ (y\ x))\ z))))$$

Use the conventions backwards to write the following λ-term in 'full-detail'

$$\lambda abcd.a\ b\ c\ d\ (d\ c\ b\ a)$$

3. A well-known λ-term (at least in scientific circles) is the so called *S-combinator*; defined by

$$S \stackrel{\mathsf{def}}{=} \lambda xyz.x\ z\ (y\ z)$$

Give its syntax tree and the set $\mathcal{S}\mathsf{ub}(S)$ of all its subterms.

4. For each λ-term t out of $\{\lambda x.x\ y, \lambda xy.z, \lambda x.x\ (y\ z)\}$ give the sets $\mathcal{V}\mathsf{ar}(t)$, $\mathcal{BV}\mathsf{ar}(t)$, and $\mathcal{FV}\mathsf{ar}(t)$—the *set of variables*, *bound variables*, and *free variables* in t, respectively.

5. Use the following type for λ-terms

```
type var  = Strng.t
type term = Var of var
          | App of (term * term)
          | Abs of (var  * term)
```

to implement the functions:

```
subterms : term -> term list
vars     : term -> var list
fvars    : term -> var list
bvars    : term -> var list
```

6. Consider the λ-term $S\ S\ S$ (recall that S stands for $\lambda xyz.x\ z\ (y\ z)$). Rewrite it to normal form (NF).

Warning: Make sure to avoid variable capture.