Problem Sheet 1.3: Introduction to Lambda Calculus

For these exercises, you do not need to follow the definition of capture-avoidance **exactly** — it is very mechanical, and thus tedious. You only need to rename an abstraction λx that would otherwise capture a variable (in the substitution rule above, when x is a free variable in M). Renaming an abstraction, from $\lambda x.M$ to $\lambda z.(M[z/x])$, simply means renaming all variables x bound by λx, and λx itself, to z.

Exercise 1: For each of the following terms, (1) say if it's a **variable**, **abstraction**, or **application**, (2) identify the free variables (encircle them), and (3) find the binding occurrence for each bound variable (connect them with an arrow).

a) x

b) $\lambda x.x$

c) $(\lambda a.z)a$

d) $\lambda a.za$

e) $(\lambda n.n)z$

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

g) $(\lambda t.((\lambda t.(\lambda t.t)t)t))t$

Exercise 2: Perform the following (capture-avoiding) substitutions.

a) $(\lambda x.xy)[\lambda z.z/y]$

b) $(\lambda x.xy)[\lambda z.zx/y]$

c) $(f(\lambda x.yx)yx)[fy/x]$

d) $(\lambda f.f(\lambda x.yx)yx)[fy/x]$

Exercise 3: Identify all the redexes in the following terms, and compute the λ-terms that result from β-reducing each redex.

a) $(\lambda x.\lambda y.x)yx$

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

c) $(\lambda x.\lambda y.yx)(\lambda x.xy)$

d) $(\lambda x.xx)((\lambda y.y)(\lambda x.x))$

e) $(\lambda x.xx)(\lambda y.y)(\lambda x.x)$

f) $(\lambda x.xx)(\lambda x.xx)((\lambda y.y)(\lambda x.x))$

Exercise 4:

a) Find distinct terms M, N, P such that $M \to_\beta P$ and $N \to_\beta P$.

b) Find distinct terms M, N, P, Q such that $M \to_\beta N$, $M \to_\beta P$ and $M \to_\beta Q$.