

Exercises

The Lambda Calculus

Exercise 1

Keeping in mind both alpha equivalence and how multiple heads are nested, choose an answer that is equivalent to the listed lambda term.

1. $\lambda xy.xz$
 - (a) $\lambda xz.xz$
 - (b) $\lambda mn.mz$
 - (c) $\lambda z(\lambda.x.xz)$
2. $\lambda xy.xxy$
 - (a) $\lambda mn.mnp$
 - (b) $\lambda x.(\lambda y.xy)$
 - (c) $\lambda a(\lambda b.aab)$
3. $\lambda xyz.zx$
 - (a) $\lambda x.(\lambda y.(\lambda z))$
 - (b) $\lambda tos.st$
 - (c) $\lambda mnp.mn$

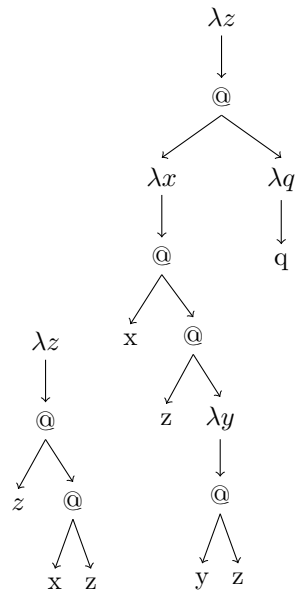
Exercise 2

Write out lambda calculus trees for the following expressions.

1. $\lambda x.\lambda y.xy$
2. $\lambda x..\lambda y.xy.\lambda z.yz$
3. $\lambda x.(\lambda y.xy)(\lambda z.yz)$

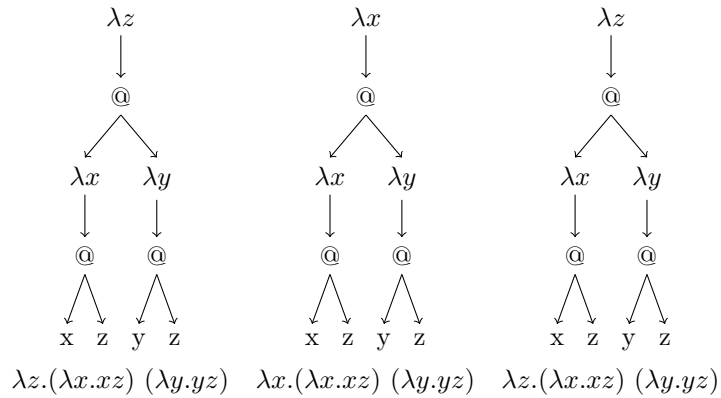
Exercise 3

Write out the equivalent lambda calculus expression for the following trees.



Exercise 4

Find the free variables. What are the free variables? To which lambdas are bound variables bound?



Exercise 5

Using β reduction etc., rewrite these expressions in normal form.

1. $(\lambda x.x)y$
2. $(\lambda x.xz)(\lambda y.y)$
3. $(\lambda x.x(\lambda x.y))(\lambda z.z)$
4. $(\lambda x.(\lambda y.x))y(\lambda z.z)$

5. $(\lambda x.xx)(\lambda x.xx)$

Exercise 6

Rewrite, using Haskell and evaluate the following:

1. $(\lambda x.x)2$
2. $(\lambda x.(x * 2))4$
3. $(\lambda x.(\lambda y.x * y))3\ 4$
4. $(\lambda x.\lambda y.(if\ x < y\ then\ -1\ else\ if\ x == y\ then\ 0\ else\ 1))\ 3\ 4$
 (**Note:** Use of if inside the lambda expression.)

Exercise 7

Write a lambda version of the following functions:

1. **abs**: which takes an Integer and returns the non-negative value.
 e.g. $abs\ -1 = 1$, $abs\ 4 = 4$.
2. **mymax**: which takes two numbers and returns the larger of the two
3. **mymin**: which takes two numbers and returns the smaller of the two