Homework: Lambda Calculus Solutions

Learning Objectives:

1. Learn to perform β-reduction
2. Understand evaluation order
3. Understand church encoding

Instructions:

- Total points: 75 pt
- Early deadline: March 29 (Wed) at 11:59 PM; Regular deadline: March 31 (Fri) at 11:59 PM (you can continue working on the homework till TA starts to grade the homework).
- How to submit:
  - Submit your document to Canvas under Assignments, Homework 6.
  - Please provide the complete solutions in one PDF file.
  - You can write your solutions in latex or word and then convert it to PDF; or you can submit a scanned document with legible handwritten solutions.

Questions:

1. (6 pt) [Understanding λ expressions] Mark all the free variables in the following λ expressions:
   (a) (3 pt) $(\lambda x.xz)\ \lambda y.w\lambda w.wyzx$
   (b) (3 pt) $\lambda x.xy\ \lambda x.yx$

   Solution: Free variables marked in red.

   (a) ((λx.(x z))) (λy.(w (λw.(((w y) z) x))))
   (b) (λx.(x y) λx.(y x))

2. (15 pt) [β-reduction] Perform β-reduction for the following λ expressions.
   (a) (3 pt) $\lambda z.z\ \lambda y.(y\ y)\ (\lambda x.x\ a)$
   (b) (3 pt) $\lambda x.\lambda y.(x\ y\ y)\ (\lambda a.a)\ b$
   (c) (3 pt) $\lambda x.(x\ x)\ \lambda y.(y\ x)\ z$
   (d) (3 pt) $\lambda x.(x\ x)\ \lambda y.y\ \lambda y.y$
   (e) (3 pt) $\lambda x.m\ \lambda y.y\ ((\lambda v.v\ a)\ (\lambda w.wb))$

Solution:

(a) (3pt)

$$
\begin{aligned}
& \lambda z.z\ \lambda y.(y\ y)\ (\lambda x.x\ a) && (1)\\
= \ & \lambda y.(y\ y)\ (\lambda x.x\ a) && (2)\\
= \ & (\lambda x.x\ a)\ (\lambda x.x\ a) && (3)\\
= \ & a\ (\lambda x.x\ a) && (4)\\
= \ & a\ a && (5)
\end{aligned}
$$

(b) (3 pt)

$$
\begin{aligned}
& \lambda x.\lambda y.(x\ y\ y)\ (\lambda a.a)\ b && (1)\\
= \ & \lambda y.(\lambda a.a\ y\ y)\ b && (2)\\
= \ & \lambda a.a\ b\ b && (3)\\
= \ & b\ b && (4)
\end{aligned}
$$

(c) (3pt)

$$
\begin{aligned}
& \lambda x.(x\ x)\ \lambda y.(y\ x)\ z && (1)\\
= \ & \lambda y.(y\ x)\ \lambda y.(y\ x)\ z && (2)\\
= \ & (\lambda y.(y\ x)\ x)\ z && (3)\\
= \ & x\ x\ z && (4)
\end{aligned}
$$

(d) (3pt)

$$
\begin{aligned}
& \lambda x.(x\ x)\ \lambda y.y\ \lambda y.y && (1)\\
= \ & (\lambda y.y\ \lambda y.y)\ \lambda y.y && (2)\\
= \ & \lambda y.y\ \lambda y.y && (3)\\
= \ & \lambda y.y && (4)
\end{aligned}
$$

(e) (3pt)

$$
\begin{aligned}
& \lambda x.m\ \lambda y.y\ ((\lambda v.v\ a)\ (\lambda w.wb)) && (1)\\
= \ & m\ ((\lambda v.v\ a)\ (\lambda w.wb)) && (2)\\
= \ & m\ (a\ (\lambda w.w\ b)) && (3)
\end{aligned}
$$

OR

$$
\begin{aligned}
& \lambda x.m\ \lambda y.y\ ((\lambda v.v\ a)\ (\lambda w.wb)) && (1)\\
= \ & \lambda x.m\ ((\lambda v.v\ a)\ (\lambda w.wb)) && (2)\\
= \ & \lambda x.m\ (a\ (\lambda w.wb)) && (3)\\
= \ & m && (4)
\end{aligned}
$$

3. (6 pt) [Evaluation order] The goal of this problem is to help you understand the evaluation order of lambda calculus. Show the steps of β-reduction for the following lambda expression using two different evaluation orders.

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

Solution:

(a) Lazy Evaluation

$$\begin{aligned} & \lambda x.y\ (\lambda y.(y\ y)\ \lambda z.(z\ z\ z)) && (1)\\ = \ & y && (2)\end{aligned}$$

(b) Non-Lazy Evaluation

$$\begin{aligned} & \lambda x.y\ (\lambda y.(y\ y)\ \lambda z.(z\ z\ z)) && (1)\\ = \ & \lambda x.y\ (\lambda z.(z\ z\ z)\ \lambda z.(z\ z\ z)) && (2)\\ = \ & \lambda x.y\ (\lambda z.(z\ z\ z)\ \lambda z.(z\ z\ z)\ \lambda z.(z\ z\ z)\ \lambda z.(z\ z\ z)) && (3)\\ & && (4)\end{aligned}$$

This keeps on expanding.

4. (12 pt) [Boolean Operations] Using *true*, *false*, *ite*, *not* and *or* defined in the lecture slides:

(a) (4 pt) Encode the logic Boolean operation $a \oplus b$.

(b) (4 pt) Prove `not (not true) = true`

(c) (4 pt) Prove `(((ite false) x) y) = y`

Solution:

(a) From the Table in slide 47, we can get: If `a` is true, then return `not b`; If `a` is false, then return `b`'s value.

$$\begin{aligned} & \text{ite a (not b) b} && (1)\\ = \ & \text{(((ite a) (not b)) b)} && (2)\\ = \ & \text{(((ite a) ((((ite x) false) true) b)) b)} && (3)\\ = \ & \text{(((ite a) (((ite b) false) true)) b)} && (4)\end{aligned}$$

(b) (4 pt)

$$\begin{aligned} & \text{not (not true)} && (1)\\ = \ & \text{(((ite x) false) true) (not true)} && (2)\\ = \ & \text{(((ite (not true)) false) true)} && (3)\\ = \ & \text{(((}\lambda\text{c.}\lambda\text{t.}\lambda\text{e.((c t) e) (not true)) false) true)} && (4)\\ = \ & \text{(((}\lambda\text{t.}\lambda\text{e.(((not true) t) e)) false) true)} && (5)\\ = \ & \text{((}\lambda\text{e.(((not true) false) e)) true)} && (6)\\ = \ & \text{(((not true) false) true)} && (7)\\ = \ & \text{((((((ite x) false) true) true) false) true)} && (8)\\ = \ & \text{(((((ite true) false) true) false) true)} && (9)\\ = \ & \text{(((((}\lambda\text{c.}\lambda\text{t.}\lambda\text{e.((c t) e) true) false) true) false) true)} && (10)\\ = \ & \text{(((((}\lambda\text{t.}\lambda\text{e.((true t) e)) false) true) false) true)} && (11)\\ = \ & \text{((((}\lambda\text{e.((true false) e)) true) false) true)} && (12)\\ = \ & \text{((((true false) true) false) true)} && (13)\\ & && (14)\end{aligned}$$

$$
\begin{aligned}
&= ((((\lambda x.\lambda y.x \text{ false}) \text{ true}) \text{ false}) \text{ true}) && (15)\\
&= (((\lambda y.\text{false true}) \text{ false}) \text{ true}) && (16)\\
&= ((\text{false false}) \text{ true}) && (17)\\
&= ((\lambda x.\lambda y.y \text{ false}) \text{ true}) && (18)\\
&= (\lambda y.y \text{ true}) && (19)\\
&= \text{true} && (20)
\end{aligned}
$$

(c) (4pt)

$$
\begin{aligned}
& (((\text{ite false}) \text{ x}) \text{ y}) && (1)\\
&= (((\lambda t.\lambda e.((\text{false t}) \text{ e})) \text{ x}) \text{ y}) && (2)\\
&= ((\lambda e.((\text{false x}) \text{ e})) \text{ y}) && (3)\\
&= (((\text{false x}) \text{ y})) && (4)\\
&= (((\lambda x.\lambda y.y \text{ x}) \text{ y})) && (5)\\
&= (\lambda y.y \text{ y}) && (6)\\
&= \text{y} && (7)
\end{aligned}
$$

5. (20 pt) [Church Encoding] Given:

$$
\begin{aligned}
zero &: \lambda f.\lambda y.y\\
one &: \lambda f.\lambda y.(f\ y)\\
two &: \lambda f.\lambda y.(f\ (f\ y))\\
three &: \lambda f.\lambda y.(f\ (f\ (f\ y)))\\
succ &: \lambda p.\lambda q.\lambda r.(q\ ((p\ q)\ r))\\
false &: \lambda a.\lambda b.b\\
true &: \lambda a.\lambda b.a\\
unknown &: \lambda m.\lambda n.\lambda o.n\\
g &: \lambda s.((s\ unknown)\ false)
\end{aligned}
$$

(a) (4 pt) What is the result of $(\lambda z.((two\ f)\ z))(succ\ zero)$?

(b) (4 pt) What is the result of $g\ zero$?

(c) (4 pt) What is the result of $g\ one$?

(d) (4 pt) What is the result of $g\ two$?

(e) (4 pt) What mathematical/logical operation does g perform?

Solution:

(a) (f (f one))

$$
\begin{aligned}
& (\lambda\ \text{z}.((\text{two f})\ \text{z}))\ (\text{succ zero}) && (1)\\
= & ((\text{two f})\ (\text{succ zero})) && (2)\\
= & ((\lambda\ \text{p}.\lambda\ \text{y}.(\text{p}\ (\text{p y}))\ \text{f})\ (\text{succ zero})) && (3)\\
= & ((\lambda\ \text{f}.\lambda\ \text{y}.(\text{f}\ (\text{f y}))\ \text{f})\ (\text{succ zero})) && (4)\\
= & (\lambda\ \text{y}.(\text{f}\ (\text{f y}))\ (\text{succ zero})) && (5)\\
= & (\text{f}\ (\text{f}\ (\text{succ zero}))) && (6)\\
= & (\text{f}\ (\text{f}\ (\lambda p.\lambda q.\lambda r.(q\ ((p\ q)\ r))\ \text{zero}))) && (7)\\
= & (\text{f}\ (\text{f}\ (\lambda q.\lambda r.(q\ ((\text{zero}\ q)\ r))))) && (8)\\
= & (\text{f}\ (\text{f}\ (\lambda q.\lambda r.(q\ ((\lambda f.\lambda y.y\ q)\ r))))) && (9)\\
= & (\text{f}\ (\text{f}\ (\lambda q.\lambda r.(q\ (\lambda y.y\ \ r))))) && (10)\\
= & (\text{f}\ (\text{f}\ (\lambda q.\lambda r.(q\ r)))) && (11)\\
= & (\text{f}\ (\text{f}\ (\lambda f.\lambda y.(f\ y)))) && (12)\\
= & (\text{f}\ (\text{f one})) && (13)
\end{aligned}
$$

(b) False.

$$
\begin{aligned}
& \text{g zero} && (1)\\
= & \lambda\text{s}.((\text{s unknown})\ \text{false})\ \text{zero} && (2)\\
= & ((\text{zero unknown})\ \text{false}) && (3)\\
= & ((\lambda\text{f}.\lambda\text{y}.\text{y unknown})\ \text{false}) && (4)\\
= & (\lambda\text{y}.\text{y false}) && (5)\\
= & \text{false} && (6)
\end{aligned}
$$

(c) True.

$$
\begin{aligned}
& \text{g one} && (1)\\
= & \lambda\text{s}.((\text{s unknown})\ \text{false})\ \text{one} && (2)\\
= & ((\text{one unknown})\ \text{false}) && (3)\\
= & (\text{unknown false}) && (4)\\
= & (\lambda\text{m}.\lambda\text{n}.\lambda\text{o}.\text{n false}) && (5)\\
= & (\lambda\text{n}.\lambda\text{o}.\text{n}) && (6)\\
= & (\lambda\text{a}.\lambda\text{b}.\text{a}) && (7)\\
= & \text{true} && (8)
\end{aligned}
$$

(d) True.

$$
\begin{aligned}
& \text{g two} && (1)\\
= \;& \lambda\text{s.((s unknown) false) two} && (2)\\
= \;& \text{((two unknown) false)} && (3)\\
= \;& \text{(unknown (unknown false))} && (4)\\
= \;& (\lambda\text{m.}\lambda\text{n.}\lambda\text{o.n (unknown false))} && (5)\\
= \;& (\lambda\text{n.}\lambda\text{o.n)} && (6)\\
= \;& (\lambda\text{a.}\lambda\text{b.a)} && (7)\\
= \;& \text{true} && (8)
\end{aligned}
$$

(e) `g` returns `unknown` applied natural number (`n`) times. `unknown` called on any argument is going to return true. But if the `n` is `zero`, meaning we don't apply the function, then `g` returns the false. Hence, g is checking if its argument is not equal to zero.

6. (16 pt) [Church Encoding] Given:

$$
\begin{aligned}
\mathit{zero} &: \lambda f.\lambda y.y\\
\mathit{one} &: \lambda f.\lambda y.(f\ y)\\
\mathit{two} &: \lambda f.\lambda y.(f\ (f\ y))\\
\mathit{three} &: \lambda f.\lambda y.(f\ (f\ (f\ y)))\\
\mathit{four} &: \lambda f.\lambda y.(f\ (f\ (f\ (f\ y))))\\
\mathit{false} &: \lambda a.\lambda b.b\\
\mathit{true} &: \lambda a.\lambda b.a\\
\mathit{unknown} &: \lambda m.\lambda n.\lambda o.((m\ n)\ ((m\ n)\ o))
\end{aligned}
$$

(a) (4 pt) What is the result of *unknown one*?

(b) (4 pt) What is the result of *unknown two*?

(c) (4 pt) What is the result of *unknown zero*?

(d) (4 pt) What mathematical/logical operation does *unknown* perform?

Solution:

(a) Two.

$$
\begin{aligned}
& \text{unknown one} && (1)\\
= \;& \lambda\text{m.}\lambda\text{n.}\lambda\text{o.((m n) ((m n) o)) one} && (2)\\
= \;& \lambda\text{n.}\lambda\text{o.((one n) ((one n) o))} && (3)\\
= \;& \lambda\text{n.}\lambda\text{o.(n ((one n) o))} && (4)\\
= \;& \lambda\text{n.}\lambda\text{o.(n (n o))} && (5)\\
= \;& \lambda\text{f.}\lambda\text{y.(f (f y))} && (6)\\
= \;& \text{two} && (7)
\end{aligned}
$$

(b) Four.

$$
\begin{aligned}
& \text{unknown two} && (1)\\
=\ & \lambda \text{m}.\lambda \text{n}.\lambda \text{o}.((\text{m}\ \ \text{n})\ \ ((\text{m}\ \ \text{n})\ \ \text{o}))\ \text{two} && (2)\\
=\ & \lambda \text{n}.\lambda \text{o}.((\text{two}\ \ \text{n})\ \ ((\text{two}\ \ \text{n})\ \ \text{o})) && (3)\\
=\ & \lambda \text{n}.\lambda \text{o}.(\text{n}\ (\text{n}\ ((\text{two}\ \ \text{n})\ \ \text{o}))) && (4)\\
=\ & \lambda \text{n}.\lambda \text{o}.(\text{n}\ (\text{n}\ (\text{n}\ (\text{n}\ \text{o})))) && (5)\\
=\ & \lambda \text{f}.\lambda \text{y}.(\text{f}\ (\text{f}\ (\text{f}\ (\text{f}\ \text{y})))) && (6)\\
=\ & \text{four} && (7)
\end{aligned}
$$

(c) Zero.

$$
\begin{aligned}
& \text{unknown zero} && (1)\\
=\ & \lambda \text{m}.\lambda \text{n}.\lambda \text{o}.((\text{m}\ \ \text{n})\ \ ((\text{m}\ \ \text{n})\ \ \text{o}))\ \text{zero} && (2)\\
=\ & \lambda \text{n}.\lambda \text{o}.((\text{zero}\ \ \text{n})\ \ ((\text{zero}\ \ \text{n})\ \ \text{o})) && (3)\\
=\ & \lambda \text{n}.\lambda \text{o}.((\text{zero}\ \ \text{n})\ \ \text{o}) && (4)\\
=\ & \lambda \text{n}.\lambda \text{o}.\text{o} && (5)\\
=\ & \lambda \text{f}.\lambda \text{y}.\text{y} && (6)\\
=\ & \text{zero} && (7)
\end{aligned}
$$

(d) `unknown` is applying the function from natural number (`n`) $2 * n$ times to the element. Hence, `unknown` is multiplication by 2.