

# Homework Solutions: Lambda Calculus

## Learning Objectives:

1. Understand evaluation order
2. Understand church encoding
3. Learn to perform  $\beta$ -reduction

## Instructions:

- Total points: 47 pt
- Early deadline: Mar 31 (Wed) at 11:59 PM; Regular deadline: Apr 2 (Fri) at 11:59 PM (you can continue working on the homework till TA starts to grade the homework).
- Submit one pdf file on Canvas under Assignments, Homework 6 submission. You are encouraged to use latex. But we will accept a scanned copy as well.

## Questions:

1. (9 pt) [ $\beta$ -reduction] Perform  $\beta$ -reduction for the following  $\lambda$  expressions.

(a) (3 pt)  $((\lambda(x)(\lambda(y)(x\ y)))(\lambda(a)\ a)\ a))((\lambda(b)\ b)\ b))$

(b) (3 pt)  $((\lambda(x)(\lambda(y)(y)))(\lambda(z)\ z)\ a))\ b)$

(c) (3 pt)  $((\lambda(x)(x\ x))(\lambda(y)\ y))\ x)$

**Sol.**

(a) (3pt)

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

(b) (3pt)

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

(c) (3pt)

$$(((\lambda(x)(x\ x))(\lambda(y)\ y))\ x) \quad (1)$$

$$= (((\lambda(y)y)(\lambda(y)\ y))\ x) \quad (2)$$

$$= ((\lambda(y)\ y)\ x) \quad (3)$$

$$= x \quad (4)$$

2. (4 pt) [Evaluation order] The goal of this problem is to help you understand the evaluation order of lambda calculus.

In the following, show the steps of  $\beta$ -reduction for the lambda expression using two types of evaluation orders

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

**Sol.**

(a) (3pt)

$$((\lambda(x)p)((\lambda(y)(y\ y))(\lambda(z)(z\ z)))) \quad (1)$$

$$= p \quad (2)$$

(b) (3pt)

$$((\lambda(x)p)((\lambda(y)(y\ y))(\lambda(z)(z\ z)))) \quad (1)$$

$$= ((\lambda(x)p)((\lambda(z)(z\ z))\ (\lambda(z)(z\ z)))) \quad (2)$$

$$= ((\lambda(x)p)((\lambda(z)(z\ z))\ (\lambda(z)(z\ z)))) \quad (3)$$

$$= ((\lambda(x)p)((\lambda(z)(z\ z))\ (\lambda(z)(z\ z)))) \quad (4)$$

$$= \dots \quad (5)$$

3. (7 pt) [Church Encoding] Encode the following logic Boolean operations using the encoding of *true*, *false*, *ite*, *not* and *or* provided in the lecture.

(a) (3 pt) *and*  $a\ b$ (b) (4 pt)  $a \rightarrow b$ **Sol.**

```
(a)      (ite a
          (ite b true false)
          false)
(b)
```

```

      (ite a
          (ite b true false )
          true
      )

```

4. (16 pt) [Church Encoding and understanding the semantics of lambda expressions] Using *zero*, *one* and *two* as well as *succ*, *true* and *false* provided in the lecture, answer the following two questions:

- (a) (4 pt) What is the result of  $((\lambda(z)((one\ f)\ z))\ (succ\ zero))$  ?
- (b) Suppose we define *unknown*:  $(\lambda(x)(\lambda(y)(\lambda(z)\ z)))$  and *g*:  $(\lambda(n)((n\ unknown)\ true))$ , what is the result of:
- (4 pt)  $(g\ zero)$
  - (3 pt)  $(g\ one)$
  - (2 pt)  $(g\ two)$
  - (3 pt) What mathematical/logical operation is computed by *g*?

**Sol.**

- (a) (4 pt)
- $$\begin{aligned}
 & ((\lambda(z)((one\ f)\ z))\ (succ\ zero)) = \\
 & ((\lambda(z)((\lambda(f)(\lambda(x)(f\ x)))\ f)\ z))\ (succ\ zero)) = \\
 & ((\lambda(z)((\lambda(x)(f\ x))\ z))\ (succ\ zero)) = \\
 & ((\lambda(z)(f\ z))\ (succ\ zero)) = \\
 & ((\lambda(z)(f\ z))\ ((\lambda(n)(\lambda(f)(\lambda(x)(f\ (n\ f)\ x))))\ zero)) = \\
 & ((\lambda(z)(f\ z))\ (\lambda(f)(\lambda(x)(f\ ((zero\ f)\ x))))) = \\
 & ((\lambda(z)(f\ z))\ (\lambda(f)(\lambda(x)(f\ ((\lambda(f)(\lambda(x)x))\ f)\ x))))) = \\
 & ((\lambda(z)(f\ z))\ (\lambda(f)(\lambda(x)(f\ ((\lambda(x)x)\ x))))) = \\
 & ((\lambda(z)(f\ z))\ (\lambda(f)(\lambda(x)(f\ x)))) = \\
 & ((\lambda(z)(f\ z))\ one) = \\
 & (f\ one)
 \end{aligned}$$
- (b) i. (4 pt)
- $$\begin{aligned}
 & (g\ zero) = \\
 & ((\lambda(n)((n\ unknown)\ true))\ zero) = \\
 & ((zero\ unknown)\ true) = \\
 & (((\lambda(f)(\lambda(x)x))\ unknown)\ true) = \\
 & ((\lambda(x)x)\ true) = \\
 & true
 \end{aligned}$$
- ii. (3 pt)
- $$\begin{aligned}
 & (g\ one) = \\
 & ((\lambda(n)((n\ unknown)\ true))\ one) = \\
 & ((one\ unknown)\ true) = \\
 & (((\lambda(f)(\lambda(x)(f\ x)))\ unknown)\ true) = \\
 & ((\lambda(x)(unknown\ x))\ true) =
 \end{aligned}$$

$((\lambda(x)((\lambda(x)(\lambda(y)(\lambda(z) z))) x)) \text{true}) =$   
 $((\lambda(x)(\lambda(y)(\lambda(z) z))) \text{true}) =$   
 $((\lambda(x) \text{false}) \text{true}) =$   
 $\text{false}$

iii. (2 pt)

$(g \text{ two}) =$   
 $((\lambda(n)((n \text{ unknown}) \text{true})) \text{two}) =$   
 $((\text{two unknown}) \text{true}) =$   
 $((\lambda(f)(\lambda(x)(f (f x)))) \text{unknown}) \text{true}) =$   
 $((\lambda(x)(\text{unknown} (\text{unknown } x))) \text{true}) =$   
 $(\text{unknown} (\text{unknown true})) =$   
 $(\text{unknown} ((\lambda(x)(\lambda(y)(\lambda(z) z))) \text{true})) =$   
 $(\text{unknown} (\lambda(y)(\lambda(z) z))) =$   
 $(\text{unknown false}) =$   
 $((\lambda(x)(\lambda(y)(\lambda(z) z))) \text{false}) =$   
 $(\lambda(y)(\lambda(z) z)) =$   
 $\text{false}$

iv. (3 pt) The computation performed is: testing if g is 0.

5. (11 pt) Given:

$\text{data: } (\lambda(x) (\lambda(y) (\lambda(z) ((z x) y))))$   
 $\text{op1: } (\lambda(p) (p (\lambda(x) (\lambda(y) x))))$   
 $\text{op2: } (\lambda(p) (p (\lambda(x) (\lambda(y) y))))$   
 $\text{true: } (\lambda(x) (\lambda(y) x))$   
 $\text{false: } (\lambda(x) (\lambda(y) y))$

- (a) (4 pt) What is the result of  $(\text{op1 } ((\text{data } a) b))$ ?
- (b) (4 pt) What is the result of  $(\text{op2 } ((\text{data } a) b))$ ?
- (c) (3 pt) What computation do  $\text{op1}$  and  $\text{op2}$  perform?

**Sol.**

$((\text{data } a) b) = (((\lambda(x) (\lambda(y) (\lambda(z) ((z x) y)))) a) b) = (\lambda(z) ((z a) b))$

$\text{op1: } (\lambda(p) (p (\lambda(x) (\lambda(y) x)))) = (\lambda(p) (p \text{true}))$

$\text{op2: } (\lambda(p) (p (\lambda(x) (\lambda(y) y)))) = (\lambda(p) (p \text{false}))$

- (a)  $(\text{op1 } ((\text{data } a) b)) = ((\lambda(p) (p \text{true}))(\lambda(z) ((z a) b))) =$   
 $((\lambda(z) ((z a) b)) \text{true}) =$   
 $((\text{true } a) b) =$   
 $((\lambda(x) (\lambda(y) x)) a) b =$   
 $a$

$$\begin{aligned}
 \text{(b) } (op2 ((data\ a)\ b)) &= ((\lambda(p)\ (p\ false))(\lambda(z)\ ((z\ a)\ b))) = \\
 &((\lambda(z)\ ((z\ a)\ b))false) = \\
 &((false\ a)\ b) = \\
 &(((\lambda(x)\ (\lambda(y)\ y))\ a)\ b) = \\
 &b
 \end{aligned}$$

(c) `op1` and `op2` return the first and second element respectively.