

Name: Chengxuan Li

Student Number: 1631060

1.

NOR
 $(\lambda xy|xx(yy))(\lambda x|xy)(\lambda x|x)$
 $\rightarrow_{\beta} (\lambda x|xy)(\lambda x|xy)((\lambda x|x)(\lambda x|x))$
 $\rightarrow_{\beta} (\lambda x|xy)y((\lambda x|x)(\lambda x|x))$
 $\rightarrow_{\beta} yy((\lambda x|x)(\lambda x|x))$
 $\rightarrow_{\beta} yy(\lambda x|x)$

AOR
 $(\lambda xy|xx(yy))(\lambda x|xy)(\lambda x|x)$
 $\rightarrow_{\beta} (\lambda x|xy)(\lambda x|xy)((\lambda x|x)(\lambda x|x))$
 $\rightarrow_{\beta} (\lambda x|xy)(\lambda x|xy)(\lambda x|x)$
 $\rightarrow_{\beta} (\lambda x|xy)y(\lambda x|x)$
 $\rightarrow_{\beta} yy(\lambda x|x)$

2. (a)

$NOT = (\lambda x|xFT)$
 $OR = (\lambda xy|xTy)$
 $T = (\lambda xy|x)$
 $F = (\lambda xy|y)$
 $x \subset y \equiv \neg x \vee y$
 $IMP = (\lambda xy|NOTxTy)$
 $\rightarrow_{\alpha} (\lambda xy|(\lambda x|xFT)xTy)$
 $\rightarrow_{\beta} (\lambda xy|xFTTy)$
 $\rightarrow_{\alpha} (\lambda xy|x(\lambda xy|y)(\lambda xy|x)(\lambda xy|x)y)$
 $\rightarrow_{\alpha} (\lambda xu|x(\lambda xy|y)(\lambda xy|x)(\lambda xy|x)u)$

2. (b)

$IMP \ T \ F$
 $\rightarrow_{\alpha} (\lambda xu|x(\lambda xy|y)(\lambda xy|x)(\lambda xy|x)u) \ T \ F$
 $\rightarrow_{\beta} T(\lambda xy|y)(\lambda xy|x)(\lambda xy|x)F$
 $\rightarrow_{\alpha} (\lambda xy|x)(\lambda xy|y)(\lambda xy|x)(\lambda xy|x)(\lambda xy|y)$
 $\rightarrow_{\beta} (\lambda xy|y)(\lambda xy|x)(\lambda xy|y)$
 $\rightarrow_{\beta} (\lambda xy|y)$
 $\rightarrow_{\alpha} F$
 $IMP \ F \ T$
 $\rightarrow_{\alpha} (\lambda xu|x(\lambda xy|y)(\lambda xy|x)(\lambda xy|x)u) \ F \ T$
 $\rightarrow_{\beta} F(\lambda xy|y)(\lambda xy|x)(\lambda xy|x)T$
 $\rightarrow_{\alpha} (\lambda xy|y)(\lambda xy|y)(\lambda xy|x)(\lambda xy|x)(\lambda xy|x)$
 $\rightarrow_{\beta} (\lambda xy|x)(\lambda xy|x)(\lambda xy|x)$
 $\rightarrow_{\beta} (\lambda xy|x)$
 $\rightarrow_{\alpha} T$

3.

```

( ( (lambda (x y) (lambda (z) (if (> x y) (+ x z) (+ x y) ) ) ) 4 5) 10)
- E1 = ( (lambda (x y) (lambda (z) (if (> x y) (+ x z) (+ x y) ) ) ) 4 5)
- (E1 10) is a function application. Evaluate (E1 10) in CT0:
  - Evaluate 10 in CT0:
    - 10 is constant. It evaluates itself.
  - E1 is a function application. Evaluate E1 in CT0:
    - F1 = (lambda (x y) (lambda (z) (if (> x y) (+ x z) (+ x y) ) ) )
    - Evaluate 4 and 5 in CT0
      - 4 and 5 are constants. They evaluate themselves.
    - F1 is a lambda function. F1 is evaluated to a closure [F1, CT0]:
      - Obtain argument list (x y) and function body from closure [F1, CT0]
      - Evaluated argument 4 and 5
      - New variable binding: x->4, y->5
      - Extend context: CT1 = {x->4, y->5} U CT0
      - Evaluate body of F1 in CT1 = {x->4, y->5} U CT0:
        - F2 = (lambda (z) (if (> x y) (+ x z) (+ x y) ) )
        - F2 is a lambda function. F2 is evaluated to a closure [F2, CT1]:
          - There are no arguments needed to be evaluated => No evaluated arguments need to apply
        - Evaluation of F1 complete, return [F2, CT1]
    - E1 is evaluation is completed, return [F2, CT1]
- ([F2, CT1], 10 ) is a function application
  - Obtain argument list (z) and function body from closure [F2, CT1]
  - Evaluated argument 10
  - New variable binding: z->10
  - Extend context: CT2 = {x->4, y->5, z->10} U CT0
  - Evaluate body of F2 in CT2 = {x->4, y->5, z->10} U CT0:
    - (if (> x y) (+ x z) (+ x y))
    - (if (> 4 5) (+ 4 10) (+ 4 5)), so 9
- The result of evaluating this expression is 9
- last context is CT2 = {x->4, y->5, z->10} U CT0

```

Figure 1: solution