Solution Lambda Calculus

Instructions:

Solutions of the exercises are to be delivered before Wednesday, the 11th of April at 10:15AM.
Solutions should be placed in a separate folder with the name "**Assignment05**".
Please submit answers to all the exercises in **one** text file.

Exercise 1 (2 points)

Consider the following λ-expressions. Indicate which occurrences of variables are bound and which ones are free in the expressions.

1. `(λ x . x) y (λ y . y x) x`
2. `((λ x . λ y . λ z . x y z ) (λ x.  y x) y) (λ x . z x)`

<u>**Answer:**</u>

```
b = bound
f = free

ex. 1.1.

(\ x . x) y (\ y . y x) x
       |  |        | |  |
       b  f        b f  f

ex. 1.2.

((\ x . \ y . \ z .  x y z) (\ x .  y x) y) (\ x . z x)
                     | | |              | |  |          | |
                     b b b              f b  f          f b
```

Exercise 2 (2 points)

Define boolean functions `and` and `or` in Lambda Calculus and show that `True and False = False` and `True or False = True` based on the definitions of `True` and `False` functions from the lecture hours.

Answer:

and $\equiv \lambda$ *x y . x y x*

// if the first argument is `False`*, the function should return* `False`, i.e., *the first argument;*

// if the first argument is `True`*, the function should return the second argument*

or $\equiv \lambda$ *x y . x x y*

// if the first argument is `True`*, the function should return* `True`, i.e., *the first argument;*

// if the first argument is `False`*, the function should return the second argument*

`True and False = False`

(λ x y. x y x)(λ x y. x)(λ x y. y) =

(λ x y. x)(λ x y. y)(λ x y. x) =

(λ x y. y) ≡ `False`

`True or False = True`

(λ x y. x x y)(λ x y. x)(λ x y. y) =

(λ x y. x)(λ x y. x)(λ x y. y) =

(λ x y. x) ≡ `True`

Exercise 3 (2 points)

Reduce the following λ-expressions to their normal form where possible.

1. `(λ x.  (λ z .  z y) x )(λ x .  x)`
2. `(λ x.  x x y)(λ x .  x x y)`

Answer:

a. (λ x. (λ z . z y) x)(λ x . x) = / β reduction */*
(λ z . z y)(λ x . x) = / β reduction */*
(λ x . x) y = / β reduction */*
y

b. (λ x. x x y)(λ x . x x y) = / β reduction */*
(λ x . x x y) (λ x . x x y) y = / β reduction */*
(λ x . x x y) (λ x . x x y) y y = / β reduction */*
... / no normal form */*