CSE340 SPRING 2021 - HOMEWORK 3

Due: Monday Thursday March 18, 2021 by 11:59 PM on Canvas/Gradescope

All submissions should be typed, no exceptions.

1. **(Lambda calculus: binding)**. For each of the following expressions. determine for each variable x the 𝜆x. it is bound to. I have numbered the variables and the abstractions. The variables are numbered using Arabic numerals and the abstractions are numbered using roman numerals and you should specify which variable numbers are bound to which abstraction number. For example, if variables 4 and 7 are bound to abstraction I, your answer should be of the form I ➙ 4 7 to indicate that abstraction I has variables 4, and 7 bound by it. If variable 3 is free, then your answer should have the form free ➙ 3.

**Example** ( 𝜆y. x y 𝜆x. y ) x

I 1 2 II 3 4

**Answer**  free ➙ 1 4

I ➙ 2 3

II ➙

Your answers need not be colored

* 1. 𝜆y. x y z

I 1 2 3

free ➙ 1 3

I ➙ 2

* 1. 𝜆x. x y (𝜆z. y ) ( 𝜆y. z ) x

I 1 2 II 3 III 4 5

free ➙ 2 3 4

I ➙ 1 5

II ➙

III ➙

* 1. 𝜆y. x y 𝜆x. x 𝜆y. x y 𝜆x. y y x

I 1 2 II 3 III 4 5 IV 6 7 8

free ➙ 1

I ➙ 2

II ➙ 3 4

III ➙ 5 6 7

IV ➙ 8

* 1. ( ( 𝜆x. y 𝜆y. x ) x ( 𝜆x. 𝜆x. x x 𝜆x. x ) x ) x

I 1 II 2 3 III IV 4 5 V 6 7 8

free ➙ 1 3 7 8

I ➙ 2

II ➙

III ➙

IV ➙ 4 5

V ➙ 6

* 1. ( 𝜆x. y 𝜆y. x x ( 𝜆x. 𝜆x. (x x) 𝜆y. x ) x ) x

I 1 II 2 3 III IV 4 5 V 6 7 8

free ➙ 1 8

I ➙ 2 3 7

II ➙

III ➙

IV ➙ 4 5 6

V ➙

1. (**Reducible Expressions**) For each of the following, identify all the reducible expressions by highlighting the 𝜆x. , the t and the t’ of the reducible expressions. If there is more than one redex in the given expression, you should identify all the redexes, each one on a separate line. If there is no reducible expression, you should say so in your answer.

**Example 1** ( 𝜆x. 𝜆x. x x )

**Answer** no redex

**Example 2** ( ( 𝜆x. x x ) ( 𝜆x. x x ) ) ( ( 𝜆x. x x ) ( 𝜆x. x x ) )

**Answer (** ( 𝜆x. x x ) ( 𝜆x. x x ) ) **(** ( 𝜆x. x x ) ( 𝜆x. x x ) )

**(** ( 𝜆x. x x ) ( 𝜆x. x x ) ) **(** ( 𝜆x. x x ) ( 𝜆x. x x ) )

* 1. ( x x ) (𝜆x. x x) x ( 𝜆x. (𝜆x. x) x )

( ( ( x x ) (𝜆x. x x) ) x ) ( 𝜆x. (𝜆x. x) x )

* 1. 𝜆x. ( 𝜆x. ( 𝜆x. x ) x ) x

( 𝜆x. ( 𝜆x. ( 𝜆x. x ) x ) x )

( 𝜆x. ( 𝜆x. ( 𝜆x. x ) x ) x )

* 1. ( 𝜆x. y ) z 𝜆x. 𝜆x. ( 𝜆x. x ) w (𝜆x. x) x

( ( 𝜆x. y ) z ) ( 𝜆x. ( 𝜆x. ( ( ( ( 𝜆x. x ) w ) (𝜆x. x) ) x ) ) )

( ( 𝜆x. y ) z ) ( 𝜆x. ( 𝜆x. ( ( ( ( 𝜆x. x ) w ) (𝜆x. x) ) x ) ) )

* 1. ( 𝜆x. ( 𝜆y. z y ) 𝜆z. x ) y

( 𝜆x. ( ( 𝜆y. ( z y ) ) ( 𝜆z. x ) ) ) y

( 𝜆x. ( ( 𝜆y. ( z y ) ) ( 𝜆z. x ) ) ) y

* 1. 𝜆x. ( 𝜆y. x ) ( 𝜆z. x y ) 𝜆x. (𝜆x. w) ( y z )

( 𝜆x. ( ( ( 𝜆y. x ) ( 𝜆z. ( x y ) ) ) ( 𝜆x. ( (𝜆x. w) ( y z ) ) ) ) )

( 𝜆x. ( ( ( 𝜆y. x ) ( 𝜆z. ( x y ) ) ) ( 𝜆x. ( (𝜆x. w) ( y z ) ) ) ) )

1. (**Alpha Renaming**) For each of the following redexes, do alpha renaming if it is needed before reducing the redex. If renaming is not needed, you should say so. You should highlight the renamed variable(s) in your answer. This question is only about renaming. You should not do the beta reduction.

**Example 1** ( 𝜆x. x x ) x

**Answer** no renaming needed

**Example 2** ( 𝜆x. 𝜆y. 𝜆z. x y 𝜆w. z ) ( x y z w)

**Answer** ( 𝜆x. 𝜆u. 𝜆v. x u 𝜆w. v ) ( x y z w)

**Example 3** ( 𝜆x. 𝜆y. 𝜆z. x y 𝜆w. z ) ( x z ) ( z w)

**Answer** ( 𝜆x. 𝜆y. 𝜆v. x y 𝜆w. v ) ( x z ) ( z w)

* 1. (𝜆x. (𝜆y. 𝜆z. y ) z ) ( x z )

no renaming needed

* 1. (𝜆x. (𝜆y. x) (𝜆z. x y z) ) ( y z )

(𝜆x. ( (𝜆w. x) (𝜆u. x y u) ) ) ( y z )

* 1. (𝜆x. 𝜆y. x 𝜆z. x y 𝜆x. x z ) ( y z )

(𝜆x. ( 𝜆w. ( x ( 𝜆u. ( x w ( 𝜆x. ( x u ) ) ) ) ) ) ) ( y z )

* 1. 𝜆y. (𝜆x. 𝜆y. x 𝜆x. 𝜆z. x y z ) (𝜆z. y z )

𝜆y. (𝜆x. ( 𝜆w. ( x ( 𝜆x. ( 𝜆z. ( x w z ) ) ) ) ) ) (𝜆z. y z )

* 1. (𝜆y. x 𝜆z. x y z ) (𝜆y. 𝜆z. y z )

no renaming needed

1. (**Beta reductions**) For each of the following expressions, identify the redexes and, for each redex, do only one beta reduction step to reduce the redex. If renaming is needed you should do renaming first before doing the beta reduction.

**Example 1** ( 𝜆x. x x ) x

**Answer** ( 𝜆x. x x ) x →ᵝ x x

**Example 2** ( 𝜆x. 𝜆y. 𝜆z. x y 𝜆w. z ) ( x y z w)

**Answer** ( 𝜆x. 𝜆y. 𝜆z. x y 𝜆w. z ) ( x y z w) →𝛼 ( 𝜆x. 𝜆u. 𝜆v. x u 𝜆w. v ) ( x y z w)

( 𝜆x. 𝜆u. 𝜆v. x u 𝜆w. v ) ( x y z w) →ᵝ 𝜆u. 𝜆v. ( x y z w) u 𝜆w. v

**Example 3** (𝜆x. (𝜆y. x ) y ) z

**Answer** // This expression has two redexes. You should show one beta reduction for each redex separately not one // after the other

(𝜆x. (𝜆y. x ) y ) z →ᵝ (𝜆y. z ) y // beta reduction for first redex

(𝜆x. (𝜆y. x ) y ) z →ᵝ (𝜆x. x ) z // beta reduction for second redex

// The examples above clearly show that you should answer by highlighting the redexes and also highlight the result of the

// reduction (and the renaming if applicable). You should follow the same format as the examples below

* 1. ( (𝜆x. z ) ( x z ) ) ( 𝜆x. (𝜆x. z ) ( x z ) )

( (𝜆x. z ) ( x z ) ) ( 𝜆x. (𝜆x. z ) ( x z ) ) →ᵝ ( z ) ( 𝜆x. (𝜆x. z ) ( x z ) )

( (𝜆x. z ) ( x z ) ) ( 𝜆x. (𝜆x. z ) ( x z ) ) →ᵝ ( (𝜆x. z ) ( x z ) ) ( 𝜆x. z )

* 1. (𝜆x. (𝜆y. x) (𝜆z. x y z) ) ( y z )

(𝜆x. (𝜆y. x) (𝜆z. x y z) ) ( y z ) →ᵝ (𝜆x. (x) ) ( y z )

(𝜆x. (𝜆y. x) (𝜆z. x y z) ) ( y z ) →𝛼 (𝜆x. (𝜆u. x) (𝜆w. x y w) ) ( y z )

(𝜆x. (𝜆u. x) (𝜆w. x y w) ) ( y z ) →ᵝ ( (𝜆u. ( y z ) ) (𝜆w. ( y z ) y w) )

* 1. (𝜆x. 𝜆y. x 𝜆z. x y 𝜆x. x z ) ( (𝜆z. y z ) y z )

(𝜆x. 𝜆y. x 𝜆z. x y 𝜆x. x z ) ( (𝜆z. y z ) y z ) →ᵝ (𝜆x. 𝜆y. x 𝜆z. x y 𝜆x. x z ) ( (y y) z )

(𝜆x. 𝜆y. x 𝜆z. x y 𝜆x. x z ) ( (𝜆z. y z ) y z ) →𝛼 (𝜆x. 𝜆u. x 𝜆w. x u 𝜆x. x w) ( (𝜆z. y z ) y z )

(𝜆x. 𝜆u. x 𝜆w. x u 𝜆x. x w) ( (𝜆z. y z ) y z ) →ᵝ (𝜆u. ( (𝜆z. y z ) y z ) 𝜆w. ( (𝜆z. y z ) y z ) u 𝜆x. x w)

* 1. 𝜆y. (𝜆x. 𝜆y. x 𝜆x. 𝜆z. x y z ) ( (𝜆z. y z ) (𝜆z. y z ) )

𝜆y. (𝜆x. 𝜆y. x 𝜆x. 𝜆z. x y z ) ( (𝜆z. y z ) (𝜆z. y z ) ) →ᵝ 𝜆y. (𝜆x. 𝜆y. x 𝜆x. 𝜆z. x y z ) ( (y (𝜆z. y z ))))

𝜆y. (𝜆x. 𝜆y. x 𝜆x. 𝜆z. x y z ) ( (𝜆z. y z ) (𝜆z. y z ) ) →𝛼 𝜆y. (𝜆x. 𝜆w. x 𝜆x. 𝜆z. x w z ) ( (𝜆z. y z ) (𝜆z. y z ) )

𝜆y. (𝜆x. 𝜆w. x 𝜆x. 𝜆z. x w z ) ( (𝜆z. y z ) (𝜆z. y z ) ) →ᵝ 𝜆y. (𝜆w. ( (𝜆z. y z ) (𝜆z. y z ) ) 𝜆x. 𝜆z. x w z )

* 1. (𝜆y. x 𝜆z. x y z ) (𝜆y. 𝜆z. y z )

(𝜆y. x 𝜆z. x y z ) (𝜆y. 𝜆z. y z ) →ᵝ(x 𝜆z. x (𝜆y. 𝜆z. y z ) z)

1. (**Call by value**) repeat problem 4 but do beta reductions according to call by value.

// This is asking you to repeat problem 4. The answer to this question should have the same format as the answer to problem 4.

* 1. ( (𝜆x. z ) ( x z ) ) ( 𝜆x. (𝜆x. z ) ( x z ) )

There are two redexes:

1. ( (𝜆x. z ) ( x z ) ) ( 𝜆x. (𝜆x. z ) ( x z ) )
2. ( (𝜆x. z ) ( x z ) ) ( 𝜆x. (𝜆x. z ) ( x z ) )

Redex 2 is under an abstraction, so it cannot be reduced under call by value evaluation strategy. Redex 1 is not under an abstraction and its argument (x z) is a value, so only redex 1 can be reduced under call by value evaluation strategy.

( (𝜆x. z ) ( x z ) ) ( 𝜆x. (𝜆x. z ) ( x z ) ) →ᵝ ( z ) ( 𝜆x. (𝜆x. z ) ( x z ) )

* 1. (𝜆x. (𝜆y. x) (𝜆z. x y z) ) ( y z )

There are two redexes:

1. (𝜆x. (𝜆y. x) (𝜆z. x y z) ) ( y z )
2. (𝜆x. (𝜆y. x) (𝜆z. x y z) ) ( y z )

Redex 1 is under an abstraction, so it cannot be reduced under call by value evaluation strategy. Redex 2 is not under an abstraction and its argument (y z) is a value, so only redex 1 can be reduced under call by value evaluation strategy.

(𝜆x. (𝜆y. x) (𝜆z. x y z) ) ( y z ) →𝛼 (𝜆x. (𝜆u. x) (𝜆w. x y w) ) ( y z )

(𝜆x. (𝜆u. x) (𝜆w. x y w) ) ( y z ) →ᵝ ( (𝜆u. ( y z ) ) (𝜆w. ( y z ) y w) )

* 1. (𝜆x. 𝜆y. x 𝜆z. x y 𝜆x. x z ) ( (𝜆z. y z ) y z )

There are two redexes:

1. (𝜆x. 𝜆y. x 𝜆z. x y 𝜆x. x z ) ( (𝜆z. y z ) y z )
2. (𝜆x. 𝜆y. x 𝜆z. x y 𝜆x. x z ) ( (𝜆z. y z ) y z )

Both redexes are not under abstraction. The argument of Redex 2 is not a value, hence it cannot be reduced under Call by value evaluation strategy. The argument of Redex 1 is a value, so only Redex 1 can be reduced.

(𝜆x. 𝜆y. x 𝜆z. x y 𝜆x. x z ) ( (𝜆z. y z ) y z ) →ᵝ (𝜆x. 𝜆y. x 𝜆z. x y 𝜆x. x z ) ( (y y) z )

* 1. 𝜆y. (𝜆x. 𝜆y. x 𝜆x. 𝜆z. x y z ) ( (𝜆z. y z ) (𝜆z. y z ) )

There are two redexes:

1. 𝜆y. (𝜆x. 𝜆y. x 𝜆x. 𝜆z. x y z ) ( (𝜆z. y z ) (𝜆z. y z ) )
2. 𝜆y. (𝜆x. 𝜆y. x 𝜆x. 𝜆z. x y z ) ( (𝜆z. y z ) (𝜆z. y z ) )

Both redexes are under abstraction. Hence, neither redex can be reduced under call by value evaluation strategy.

* 1. (𝜆y. x 𝜆z. x y z ) (𝜆y. 𝜆z. y z )

There is one redex:

(𝜆y. x 𝜆z. x y z ) (𝜆y. 𝜆z. y z )

This redex is not under abstraction and its argument is a value. Hence, it can be reduced under call by value evaluation strategy.

(𝜆y. x 𝜆z. x y z ) (𝜆y. 𝜆z. y z ) →ᵝ(x 𝜆z. x (𝜆y. 𝜆z. y z ) z)

1. (**Normal Order**) repeat problem 4 but do beta reductions according to normal order evaluation

// This is asking you to repeat problem 4. The answer to this question should have the same format as the answer to problem 4.

* 1. ( (𝜆x. z ) ( x z ) ) ( 𝜆x. (𝜆x. z ) ( x z ) )

There are two redexes:

1. ( (𝜆x. z ) ( x z ) ) ( 𝜆x. (𝜆x. z ) ( x z ) )
2. ( (𝜆x. z ) ( x z ) ) ( 𝜆x. (𝜆x. z ) ( x z ) )

Redex 1 is the left outermost and should be reduxed first. We get the following:

( (𝜆x. z ) ( x z ) ) ( 𝜆x. (𝜆x. z ) ( x z ) ) →ᵝ ( z ) ( 𝜆x. (𝜆x. z ) ( x z ) )

* 1. (𝜆x. (𝜆y. x) (𝜆z. x y z) ) ( y z )

There are two redexes:

1. (𝜆x. (𝜆y. x) (𝜆z. x y z) ) ( y z )
2. (𝜆x. (𝜆y. x) (𝜆z. x y z) ) ( y z )

Redex 2 is the left outermost redex and should be reduxed first. We get the following:

(𝜆x. (𝜆y. x) (𝜆z. x y z) ) ( y z ) →𝛼 (𝜆x. (𝜆u. x) (𝜆w. x y w) ) ( y z )

(𝜆x. (𝜆u. x) (𝜆w. x y w) ) ( y z ) →ᵝ ( (𝜆u. ( y z ) ) (𝜆w. ( y z ) y w) )

* 1. (𝜆x. 𝜆y. x 𝜆z. x y 𝜆x. x z ) ( (𝜆z. y z ) y z )

There are two redexes:

1. (𝜆x. 𝜆y. x 𝜆z. x y 𝜆x. x z ) ( (𝜆z. y z ) y z )
2. (𝜆x. 𝜆y. x 𝜆z. x y 𝜆x. x z ) ( (𝜆z. y z ) y z )

Redex 2 is the left outermost redex and should be reduced first. We get the following:

(𝜆x. 𝜆y. x 𝜆z. x y 𝜆x. x z ) ( (𝜆z. y z ) y z ) →𝛼 (𝜆x. 𝜆u. x 𝜆w. x u 𝜆x. x w) ( (𝜆z. y z ) y z )

(𝜆x. 𝜆u. x 𝜆w. x u 𝜆x. x w) ( (𝜆z. y z ) y z ) →ᵝ (𝜆u. ( (𝜆z. y z ) y z ) 𝜆w. ( (𝜆z. y z ) y z ) u 𝜆x. x w)

* 1. 𝜆y. (𝜆x. 𝜆y. x 𝜆x. 𝜆z. x y z ) ( (𝜆z. y z ) (𝜆z. y z ) )

There are two redexes:

1. 𝜆y. (𝜆x. 𝜆y. x 𝜆x. 𝜆z. x y z ) ( (𝜆z. y z ) (𝜆z. y z ) )
2. 𝜆y. (𝜆x. 𝜆y. x 𝜆x. 𝜆z. x y z ) ( (𝜆z. y z ) (𝜆z. y z ) )

Redex 2 is the left outermost redex. We get the following:

𝜆y. (𝜆x. 𝜆y. x 𝜆x. 𝜆z. x y z ) ( (𝜆z. y z ) (𝜆z. y z ) ) →𝛼 𝜆y. (𝜆x. 𝜆w. x 𝜆x. 𝜆z. x w z ) ( (𝜆z. y z ) (𝜆z. y z ) )

𝜆y. (𝜆x. 𝜆w. x 𝜆x. 𝜆z. x w z ) ( (𝜆z. y z ) (𝜆z. y z ) ) →ᵝ 𝜆y. (𝜆w. ( (𝜆z. y z ) (𝜆z. y z ) ) 𝜆x. 𝜆z. x w z )

* 1. (𝜆y. x 𝜆z. x y z ) (𝜆y. 𝜆z. y z )

There is one redex:

(𝜆y. x 𝜆z. x y z ) (𝜆y. 𝜆z. y z )

We get the following:

(𝜆y. x 𝜆z. x y z ) (𝜆y. 𝜆z. y z ) →ᵝ(x 𝜆z. x (𝜆y. 𝜆z. y z ) z)

1. (**Evaluating Expressions**) Evaluate each of the following expressions
   1. tru fls tru fls tru fls tru →

(tru fls tru) fls tru fls tru →

(fls fls tru) fls tru →

tru fls tru →

fls

* 1. tru fls tru fls tru fls tru fls tru fls tru fls tru →

(tru fls tru) fls tru fls tru fls tru fls tru fls tru →

(fls fls tru) fls tru fls tru fls tru fls tru →

(tru fls tru) fls tru fls tru fls tru →

(fls fls tru) fls tru fls tru →

(tru fls tru) fls tru →

fls fls tru →

tru

* 1. 2 (plus 2) 3 →

(plus 2)((plus 2) 3) →

(plus 2) 5 →

7

* 1. fst ( snd ( pair (pair 1 2) (pair (3 4)) ) ) →

fst ( (pair (3 4) ) →

(𝜆p. p tru) (𝜆s. 𝜆b. b (3 4) s) →

(𝜆s. 𝜆b. b (3 4) s) tru →

𝜆b. b (3 4) tru →

pair (3 4) tru

* 1. a = pair 1 2

b = pair 3 4

c = pair a b

d = pair b a

e = pair c d

f = pair d c

fstsnd = 𝜆q. fst (snd q)

calculate 2 fstsnd (pair e f)

2 fstsnd (pair e f)

fstsnd (fstsnd (pair e f))

fstsnd (fst (snd (pair e f)))

fstsnd (fst f)

fstsnd (fst (pair d c))

fstsnd (d)

fstsnd (pair b a)

fst (snd (pair b a))

fst (a)

fst (pair 1 2)

1