CMPSC 461: Programming Language Concepts
Assignment 7. Due: Monday, Mar, 15th, 11:59PM

For this assignment, you need to submit your solution to **Gradescope**. All submitted work must be your own work. Submit written part as pdf or images to hw7 and submit hw7.rkt with definitions of functions required below to hw7-code.

Problem 1 [6pt] For the following term:

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

a) Calculate its free variables FV() and connect all bound variables to their definitions with lines. For example, the bound variables for this term, $\lambda x.\ x\ x\ y$ should be $\lambda \underline{x.\ x}\ \ x\ \ y$.

b) Do reduction on the term until no more β-reduction is possible. Show every steps.

Problem 2 [8pt] In hw6, you have defined a few functions on church numerals on paper. Please implement ISZERO, PRED and MINUS functions in hw7.rkt file that works on church numerals. We have already implemented some helper functions to help you get started with your implementation. Your implementation can only use (lambda ...), function application of the form (a b c) and predefined constants/constructs such as TRUE, FALSE, IF, PAIR, LEFT, RIGHT, SUCC, PLUS and ZERO.

The ENCODE and DECODE functions are strictly for testing. Your implementation should not involve either of these two functions.

a) (2pt) Define a function `ISZERO` so that given a church numeral $\underline{n}$, it returns `TRUE` (the encoding of true) if $\underline{n} = \underline{0}$; `FALSE` (the encoding of false) if $\underline{n} \neq \underline{0}$.

```
> (ISZERO ZERO)
#<procedure:TRUE>
> (ISZERO (ENCODE 100))
#<procedure:FALSE>
```

b) (4pt) Define a function `PRED` so that given a church numeral $\underline{n}$, the function returns its predecessor, assuming the predecessor of $\underline{0}$ is $\underline{0}$.

$$\texttt{PAIR} \triangleq \lambda x\ y\ f.f\ x\ y \qquad \texttt{LEFT} \triangleq \lambda p.p\ \lambda x\ y.x \qquad \texttt{RIGHT} \triangleq \lambda p.p\ \lambda x\ y.y$$

```
> (DECODE (PRED ZERO))
0
> (DECODE (PRED (ENCODE 461)))
460
```

c) (2pt) Use your encoding of `PRED` to define a subtraction function `MINUS`, so that `MINUS` $\underline{n_1}$ $\underline{n_2}$ returns $\underline{n_1 - n_2}$ when $n_1 \geq n_2$, and $\underline{0}$ otherwise.

```
> (DECODE (MINUS (ENCODE 461) (ENCODE 311)))
150
> (DECODE (MINUS (ENCODE 131) (ENCODE 461)))
0
```

Problem 3 [6pt] Implement two racket functions in hw7.rkt that manipulate lists.

a) (4pts) Write a function sequence that takes 3 arguments low, high, and fac, all assumed to be positive integers. Further assume factor is greater than 1. sequence produces a geometric sequence from low to high (including low and possibly high) where the sequence has a factor of fac between each two numbers. If low is greater than high, the sequence should be an empty list.

```
> (sequence 1 100 2)
'(1 2 4 8 16 32 64)
> (sequence 20 19 2)
'()
> (sequence 20 1000 3)
'(20 60 180 540)
```

b) (2pts) Write a function sum that returns the sum of all numbers of a list. It returns 0 if the list is empty.

```
> (sum '(1 2 3))
6
> (sum '())
0
> (sum '(1.1 2.2 3.3 4.4 5.5 100))
116.5
>
```