SENG2200/6220 –Programming Languages & Paradigms
Computer Lab for Week 11, Semester 1, 2019

1. Write a function **max** which takes two numbers as input arguments and return the maximum number. Write the definition of this function in two ways.

```
(define (max a b)
  (if (> a b) a b)
)

(define max (lambda (a b)
  (if (> a b) a b)
  ))
```

2. Based on Q1 code, use **recursion** to find the maximum number of a list.
```
(define (cmp a b)
  (if (> a b) a b)
  )
(define (max numlist)
  (if (< (length numlist) 2)
      (car numlist)
      (cmp (car numlist) (max (cdr numlist))))
))
```

3. What are differences between functions **let**, **let*** and **letrec**. Give examples to justify your answers.
   a. let: does not guarantee the order of argument evaluation
   b. let*: guarantees the order of argument evaluation, but the variable is visible to evaluations after it.
   c. letrec: variables are visible to all argument evaluations and the body. It is usually used in mutual recursions.
   d. Examples: see lecture slides.

4. What is the output of the following Scheme programs?

```
a. ((lambda (a b c . z) (list a b c z)) 1)
Syntax Error - expects 3 arguments, had 1

b. ((lambda (a b c . z) (list a b c z)) 1 2 3)
(1 2 3 ())

c. ((lambda (a b c . z) (list a b c z)) 1 2 3 4)
(1 2 3 (4))

d. ((lambda (a b c . z) (list a b c z)) 1 2 3 4 5)
(1 2 3 (4 5))

e. ((lambda s (reverse s)) 1 2)
(2 1)

f. ((lambda (s t) (+ s t)) 1 2)
3

g. ((lambda (s t) (quote (+ s t))) 1 2)
(+ s t)

h. ((lambda (s t)
      (quasiquote (unquote (+ s t)))) 1 2)
3

i. (apply + '(1 2 3 4))
10

j. (map + '(1 2 3 4) '(5 6 7 8))
(6 8 10 12)
```

5. Write Scheme code to implement factorial function.

```scheme
(define (factorial n)
   (if (< n 1)
        1
        (* n (factorial (- n 1)))
   ))
```

6. Convert Q5 into a tail recursive function

```scheme
(define (factorial n (prod 1))
   (if (< n 1)
      prod
      (factorial (- n 1) (* prod n))
   ))
```