CS 360 Lab 1

Name:

Lab 1 tasks (all in Scheme)

Part 1 (3 points)
Access Lab 1 code you tested in the preparation for Lab 1. Review specification examples provided inside files *concat.scm, length.scm, numints.scm, order.scm*.

(i) Load the file *member-insert.scm*
Run tests of the *member, insert* functions.
Provide specifications for the *member* and *insert* functions.

(ii) Load the file *maxmin.scm*
Run tests of the *maxmin* function.
Provide specifications for the *maxmin* function.

(iii) Load the file *msort.scm*
Run tests of the *msort* function.
Provide specifications for the *msort* function.

Show the results to the TA: ________ (initials)
You may open another session (keeping your current session active and available for reviewing) and proceed with the further work on Lab 1 if the TA is currently not available.

Part 2 (3 points)
Implement in Scheme the following functions. Run several tests on each of them.

(i) Non-tail and tail recursive implementation of $n!$

(ii) Non-tail and tail recursive implementation of 2^n

(iii) Apply composition formula *(define (compose g f) (lambda (x) (g (f x))))* (from section 6 of Intro to Scheme) in order to construct $2^{n!}$ (for both, non-tail and tail recursive implementations of functions of (i), (ii)).

Show the results to the TA: ________ (initials)
You may open another session (keeping your current session active and available for reviewing) and proceed with the further work on Lab 1 if the TA is currently not available.

Part 3 (4 points)
Implement in Scheme the following functions. Run several tests on each of them.

(i) The Matlab language supports a convenient notation for specifying ranges of numbers. The notation **start:step:end** denotes the range of integers start, start+step, start+2*step,...,start+n*step, where n is the largest integer such that start+n*step $\leq$ end and start+(n+1)*step > end. Note that the range may be empty if start > end. Write a scheme function **(range (start step end))**, which returns the list of integers equal to **start:step:end**.
Example: (range '(0 2 7)) => (0 2 4 6), (range '(2 2 0)) => ()

(ii) The Maple computer algeba system has a command **seq(f, i = m..n, step)**, which returns the sequence fm,...fn, where fi is the expression f with all occurrences of the symbol i replaced by the numeric value of i in the sequence of integers from m to n. Implement a scheme function **(seq f (start step end))**, and produces a list of values (f(start),f(start+step),...,f(start+n*step)), where n is the largest integer such that start+n*step $\leq$ end and start+(n+1)*step > end.
Example: (seq (lambda (x) (* x x)) '(0 2 7)) => (0 4 16 36)

Show the results to the TA: ________ (initials)
You may open another session (keeping your current session active and available for reviewing) and proceed with the further work on Lab 1 if the TA is currently not available.

Part 4 (extra credit, 3 points)
Implement in Scheme a recursive function computing binomial coefficients (slide 13 of Week 1 Part 2 displays the C code).

Show the results to the TA: ________ (initials)