Homework 6: Scheme

Put your Scheme code for the standard problems in `hw6.l`. (We will want to be able to see the `hw6.l` file in your directory.)

You should hand in a printout of your programs. (If you do extra-credit problems, please write the names of the file(s) where the source code for your solutions can be found on your print-out.)

Test your programs using the `scheme` command on Unix, which is the interpreter described in Appendix I.

1. (12 points) Do problem 1, parts (a), (c), and (e) on page 148–149 of Kamin's book. Note that you are to do this using mapcar, mapc, or combine, and the answers you define.
2. (extra credit only) Do problem 1, parts (b), (d), or (f) on page 148–149.
3. (8 points) Do problem 2 on page 149 (lex-order*).
4. (8 points) Define a function `iterate` such that `((iterate f) 0)` is the identity function and `((iterate f) k)` equals `(lambda (x) (f (((iterate f) (- k 1)) x)))` for `k` > 0. The argument to `iterate` is a function (`f`, which takes one argument), and `iterate` returns a function that takes a number (`k`), as an argument and returns a function. For example:

   ```
   (((iterate (lambda (x) (+ 1 x))) 3) 5)
   ```

   returns 8. As another example,

   ```
   (((iterate (lambda (x) (* x x))) 3) 5)
   ```

   returns $5^{(2^3)} = 390625$.
5. (10 points) Do problem 3 on page 149 of Kamin's book (infinite sets).
6. (10 points) Do problem 7 on page 150 of Kamin's book (dynamic scoping).
7. (extra credit only) Do problem 4 on page 149 of Kamin's book (continuations). Note that this exercise has a typo: the occurrence of `(gcd 3 5)` should be `(gcd 9 11)`.
8. (extra credit only) Do problem 5 on page 149 of Kamin's book (term rewriting).
9. (extra credit only) Do (one or more of) the problems 10, 12, or 15 in section 4.10.2 of Kamin's book. You will find the source for the scheme interpreter in

   ```
   /home/cs342/public/src/chap.4/scheme.p
   ```

   Make a copy using the Unix `cp` command and then edit your copy. Turn in differences (using the Unix command `diff`) as before.