Coverage: This homework covers material in §1.1—§2.7 of Dybvig’s The Scheme Programming Language.

What to turn in.

For questions 1 & 2: Verify test cases as in Lab 1 to demonstrate the correctness of your answers.

For question 3: Your pictures (hand drawings are fine)

For questions 4 & 5: A copy of the source code and verified test cases. Include contract and purpose statements for this procedures (see the homework and lab policies) for examples.

How to turn it in. See: http://www.cis.syr.edu/~royer/cis352/info/hw-policy.html.

Problems: Due noon Friday, January 25

Problem 1 (20 points) For each of the following lists, give an expression (using only the symbols a, b, c, and d; the procedure cons; and the empty list ()) that evaluates to it.

For example, (a b c) can be constructed as (cons 'a (cons 'b (cons 'c ()))). (Note: For the purpose of this questions, the answer (cons 'a '(b c)) would be considered incorrect, because it fails the above criteria.)

a. (a (b) c d)
b. ((a b) c d)
c. (a (b (c)) d)
d. (a (b (c (d))))
e. (((a) b) (c (d)))

Problem 2 (20 points) Extract the symbol a from each of the following lists, using only car, cdr, and their hybrid forms (e.g., caddr). For example, a can be extracted from (b (a c)) using (car (car (cdr (b (a c)))))) or the abbreviated (caadr (b (a c))).

a. (b c d a)
b. ( (b c) (a d))
c. (c ((a)) b)
d. (b (d (a c)))
e. (d (c (b) a))

Problem 3 (18 points) Suppose we have the following definitions:

(define thing-one '(1 (a . 2) (((z)))) )
(define thing-two '(((b 3) #t) w (x y (z)) (4 . 6)) )

Draw pictures (as in lecture) of the internal list structures for both thing-one and thing-two. (See Exercise 2.2.5 in Dybvig for an example of such a picture.)

Problem 4 (24 points) A Pythagorean triple is a collection of three positive integers (say, m, n, and p) such that m^2 + n^2 = p^2. For example, 3, 4 and 5 form a Pythagorean triple (since 32 + 42 = 52), as do 65, 72, and 97.

a. (7 points) Write a Scheme procedure allPositive? that takes three integers and returns #t if all three are positive; it should return #f otherwise.

b. (17 points) Write a Scheme procedure pythTriple? that takes three integers (x, y, and z) and determines whether or not x, y, and z form a Pythagorean triple.
Your procedure should work correctly regardless of the order of the arguments or their sign: for example, \((\text{pythTriple} \ 3 \ 4 \ 5)\) and \((\text{pythTriple} \ 4 \ 5 \ 3)\) should both return \(\#t\), whereas \((\text{pythTriple} \ 3 \ -4 \ 5)\) should return \(\#f\).

**Problem 5 (18 points)** Use `cond` to write a Scheme procedure `identify` that takes a single argument and determines whether it is:

- a number,
- a symbol,
- a boolean,
- a list,
- a dotted pair (i.e., improper list),
- a procedure,
- or none of the above.

In each case, it should return a string indicating the datums type. For example:

```scheme
> (identify 6/7)
"number"
> (identify car)
"procedure"
> (identify car)
"symbol"
> (identify (3 . 5))
"dotted pair"
> (identify #\a);; #\a is a char
"unknown"
```