Problem Set 1: Introduction to Scheme

Review the sections 1., 2., 4.1., and 6.3.2. of the Revised Report on the Algorithmic Language Scheme.

Basics:

Problem 1

Translate the following algebraic formulas into Scheme's notation. Your answer may be handwritten, however you may want to check it with the Scheme interpreter.

- \((8 \times 7) - (10 + 5)\)
- \((5 \times (4 + (-5 - -3)))\)
- \((3 / (5 - (1 / 7)))\)

For example, translating \((7 - (4 - 5))\) into Scheme's notation yields \((- 7 (- 4 5)).\)

Solution:

\((- (* 8 7) (+ 10 5))\)
\((- (* 5 (+ 4 (- -5 -3)))\)
\((- (/ 3 (- 5 (/ 1 7))))\)

---

12

Problem 2

1. Using only the Scheme procedure list, and numbers and quoted symbols (such as '+, 'sentence, and 'np), write Scheme expressions to make the following lists. Check them with the Scheme interpreter.

- \((3 4 2)\)
- \((+ 1 (- 2 3))\)
- \((\text{sentence} (\text{np (noun fred)}) (\text{vp (verb sells)}) (\text{object it}))\)

For example, to make the list \((3 (5))\), one can use the expression \((\text{list} 3 \text{ (list} 5))\).
Solution:

(list 3 4 2) 2
(list ‘+ 1 (list ‘– 2 3)) 4
(list ‘sentence
  (list ‘np (list ‘noun ‘fred))
  (list ‘vp (list ‘verb ‘sells))
  (list ‘object ‘it)) 6
-- 12

2. Using only the Scheme procedure cons, and numbers and quoted symbols (such as ‘+, ‘sentence, and ‘np), and the empty list ()
write Scheme expressions to construct the lists given in 1.

For example, to construct the list (3 (5)), one can use the expression
(cons 3 (cons (cons 5 '()) '())).

Solution:

(cons 3 (cons 4 (cons 2 '()))) 4
(cons ‘+
  (cons 1 (cons (cons ‘–
    (cons 2 (cons 3 ‘()))) ‘()))) 6
(cons `sentence
  (cons
    (cons `np (cons (cons `noun (cons `fred `())) `()))
    (cons
      (cons `vp (cons (cons `verb (cons `sells `())) '()))
      (cons (cons `object (cons `it `())) '())))) 12
----
22
-----
34

Problem 3

Suppose we are writing code for a procedure in which the variable lst is bound to a list of numbers. Suppose further that lst has the value (3 6 4 2). For each of the following, write an expression that uses lst and makes the given list:
• (3 6 4 2 5)
• (6 4 2)
• (4 2)
• (2)
• (3 2)
• (6 4)

You can check your answer on the computer by first typing

(define lst (list 3 6 4 2))

Solution:

(append lst (list 5)) or (reverse (cons 5 (reverse lst)))  6
(cdr lst)  2
(cddr lst)  2
(cdddr lst)  2
(cons (car lst) (cdddr lst))  4
(list (cadr lst) (caddr lst))  4

-----  20

Problem 4

Consider the following definition:

(define capitals '((Germany Berlin) (Italy Rome)
    (Malaysia (Kuala Lumpur))
    ((Papua New Guinea) (Port Moresby))
    ((New Zealand) Wellington)
    (Australia Canberra)))

Determine the values of the following expressions:

• (car (cdr capitals))
• (cadr capitals)
• (cdar capitals)
• (cadar capitals)
• (cadr (caddr capitals))
• (car (cddddr capitals))
• (cadr (cddddr capitals))
• (cons 'Port (cons 'Moresby '()))
• (cons (cons 'Port (cons 'Moresby '())) '())
(cons (cons 'New (cons 'Zealand '())) (cons 'Wellington '()))

Solution:
(car (cdr capitals))
→ (italy rome)
(cadr capitals)
→ (italy rome)
(cdar capitals)
→ (berlin)
(cadar capitals)
→ berlin
(cadr (caddr capitals))
→ (kuala lumpur)
(car (cddddr capitals))
→ ((papua new guinea) (port moresby))
(cadr (cddddr capitals))
→ (australia canberra)
(cons 'Port (cons 'Moresby '()))
→ (port moresby)
(cons (cons 'Port (cons 'Moresby '())) '())
→ ((port moresby))
(cons (cons 'New (cons 'Zealand '())) (cons 'Wellington '()))
→ ((new zealand) wellington)

Problem 5

What is unusual about the following expression?

((lambda (x) (list x (list (quote quote) x)))
 (quote (lambda (x) (list x (list (quote quote) x)))))

Solution: The term evaluates to itself. 4

Total: 12 + (12 + 34) + 20 + 10 + 4 = 92