FOURTH QUIZ

You have 15 minutes from the start of class to complete this quiz. Read the problems with care; work with deliberate speed. Don’t give us more than we ask for. The usual instructions apply. Good luck!

Problem 1 (4 points)

Fill in the blanks in the function described below. Note that `even?` is a predefined function in Scheme; it returns true if its argument is an even number.

```scheme
;; count-evens: list-of-numbers -> number
;; Return the number of even numbers in the input list
(define count-evens
  (lambda (L)
    (cond
      ((_______________ L) _______________
      ((empty? L) 0)
      ((even? (______________ L)) (______________ 1 (______________ (______________ L))))
      ((even? (first L)) (+ 1 (count-evens (rest L))))
      (else (________________ (________________ L))))))))
```

SCORING:  1/2 point per blank.

Problem 2 (5 points)

Complete the definition of the function below according to the contract, purpose, and examples shown.

```scheme
;; double-all: list-of-numbers -> list-of-numbers
;; Return the input list with all its items doubled
(define double-all
  (lambda (L)
    (cond                      ;; SCORING:  1 point for first cond clause, testing empty? and returning empty
      ((empty? L) empty)            ;;  Second cond clause:
      (else (cons (* 2 (first L)) (double-all (rest L))))))) ;;  1 pt for multiplying (first L) by 2

(check-expect (double-all empty) empty)
(check-expect (double-all (list 1 2 3 4 5)) (list 2 4 6 8 10))
```

Informatics 41 • Fall 2008 • David G. Kay & Rich Pattis • UC Irvine • Your name ___________________
Problem 3 (9 points)

We define a book as a structure (define-struct book (author title year wholesale retail)) where author and title are strings and year, wholesale, and retail are numbers (representing the year the book was published, the book’s wholesale price, and its retail price).

(a) (3 points) Define the function book-prices-valid? as described below.

;; book-prices-valid?: book -> boolean
;; Return true if the book’s retail price is greater than its wholesale price
(define book-prices-valid? (lambda (B)
   (> (book-retail B) (book-wholesale B)))))

(b) (6 points) Define the function keep-valid-books as described below. Where applicable, use functions you have already defined rather than duplicating code.

;; keep-valid-books: list-of-book -> list-of-book
;; Return a list containing all the books on the input list whose prices are valid
(define keep-valid-books (lambda (L)
   (cond ((empty? L) empty)
         ((book-prices-valid? (first L)) (cons (first L) (keep-valid-books (rest L))))
         (else (keep-valid-books (rest L))))))

Problem 4 (2 points)

In class we discussed the relative advantages and disadvantages of command-line user interfaces and graphical user interfaces (GUIs).

(a) Give one advantage of a command-line user interface over a GUI.

(b) Give one advantage of a GUI over a command-line interface (other than “it looks nicer”).