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 (3 points)

We define a structure to represent courses as follows:

```
(define-struct course (dept number cap enroll))
```

The dept field is the department name (a string), number is the course number (a string, since it can contain letters), cap is the capacity (the maximum number of students who can enroll), and enroll is the number of students currently enrolled.

Define the function increase-capacity as described below:

```
;; increase-capacity:  course  number  ->  course
;; Return the input course with its capacity increased by number
(define increase-capacity
  (lambda (C n)
```

Problem 2 (6 points)

Fill in the blanks in the function described below, which takes a list of course structures as its input. All the parentheses are in the correct places and each blank should be filled by exactly one item: function name, constant, or other name.

```
;; count-full-courses:  list-of-course -> number
;; Return the number of full courses in the input list
(define count-full-courses
  (lambda (L)
    (cond

      ((________________ L) ________________)

      ((= (________________ (________________ L)) (________________ (________________ L)))

        (________________ 1 (________________ (________________ L))))

      (else (________________ (________________ L))))))
```

Problem 3 (5 points)

Complete the definition of the function below, using previous definitions where possible:

```
;; increase-all-caps:  list-of-course  number  ->  list-of-course
;; Return the input list with each course's capacity increased by the number
(define increase-all-caps
  (lambda (L n)
```

Problem 4 (6 points)

Complete the definition of the function below:

```
;; remove-empty-courses:  list-of-course  ->  list-of-course
;; Return the input list with all the courses with zero enrollment removed
(define remove-empty-courses
  (lambda (L)
```