CS1102

Sample Midterm Exam 2

Name:

Problem	Points	Score
1	20	
2	30	
3	50	
	Total	

You have 50 minutes to complete the problems on the following pages. There should be sufficient space provided for your answers. You do not need to show templates, but you may receive partial credit if you do. You also do not need to show test cases or examples of data, but you may develop them if they will help you write the programs. You do not need to use map/filter unless a problem states otherwise (you may use them if you wish).

Your programs may contain only the following Scheme syntax:

define define-struct cond else lambda local let

and the following primitive operations:

empty? cons? cons first rest list append map filter length
number? $+ - * / = < > <= >=$ *zero?*
symbol? symbol=? string? string=? eq? equal?
boolean? **and or** *not*

and the functions introduced by **define-struct**.

You may, of course, use whatever constants are necessary.

1. (20 points) myWPI maintains data on people associated with each course, including their name, role in the course (ie: instructor, TA, or student), and whether they have permission to edit grades. The following data definition captures a person:

   ;; A *person* is a (*make-person string symbol boolean*)
   (**define-struct** *person* (*name role edit-grades?*))

   Write a function *any-students-edit?* that consumes a list of person and produces a boolean indicating whether any students are allowed to edit grades in the course.

   ;; any-students-edit? : list[person] → boolean
   ;; determine whether any student in list has permission to edit grades

(exam continues next page)

2. (30 points) Consider the following data defintion for rental cars:

```
;; A car is a (make-car string number number boolean)
(define-struct car (model mileage miles-since-service problems?))
```

Assume that *carlist* is a list of cars. For each of the following expressions, give an English description of what the expression does (i.e., a purpose statement like “produces a list of cars with high mileage” or “increases the mileage of all cars”)

(a) (10 points)

```
(map car-miles-since-service
     (filter (lambda (acar) (string=? (car-model acar) ”Prius”))
             carlist))
```

(b) (10 points)

```
(map (lambda (acar)
       (cond [(string=? (car-model acar) ”Jeep”)
              (make-car (car-model acar) (car-mileage acar) 0 false)]
             [else acar]))
     carlist)
```

(c) (10 points)

```
(empty? (filter (lambda (num) (< num 1000))
                (map car-mileage
                     (filter car-problems? carlist))))
```

(exam continues next page)

3. (50 points) A company maintains information on its employees and who each one supervises. The following data definition captures employees:

   ;; A *employee* is a (*make-employee string number list*[*employee*])
   (**define-struct** *employee* (*name salary supervises*))

   (a) (25 points) Write a program *give-raises* that takes an employee (ie, the tree starting from the boss) and gives all employees who earn less than 60,000 a 10% raise.

   ;; give-raises : employee → employee
   ;; returns tree where all employees who earn less than 60,000 get a 10% raise

(exam continues next page)

(b) (25 points) Write a program *none-below* that takes an employee and returns a list of names of employees who don’t supervise anyone else.

;; none-below : employee → list[string]
;; returns list of names of employees who don’t supervise anyone else

(end of exam)