# Midterm exam 1 (sample #2)

CS61a (sp11)

### Problem 1 (What will Scheme print?)

What will Scheme print in response to the following expressions? If an expression produces an error message, you may just write “error”; you don’t have to provide the exact text of the message. **Also, draw a box and pointer diagram for the value produced by each expression.**

(list (cons 2 3) 4)

**-> ((2 . 3) 4)**



(append (cons '(a) '(b)) '(c))

**-> ((a) b c)**



(cdadar '((e (f) g) h))

**-> ()**

### Problem 2 (Backwards what-will-Scheme-print.)

Fill in the blank in this expression:

( **cadadr** '(G (H I) J))

so that the value of the expression is the letter I.

### Problem 3 (Project 2 data abstraction)

**(define (make-segment start end)**

**(attach-tag 'segment (cons start end)))**

**(define (start-segment s)**

**(car (contents s)))**

**(define (end-segment s)**

**(cdr (contents s)))**

**(define (midpoint obj)**

**(let ((type (type-tag obj)))**

**(cond**

**((eq? type 'segment)**

**(make-vector**

**(/ (+ (x-coord (start-segment obj))**

**(x-coord (end-segment obj))) 2)**

**(/ (+ (y-coord (start-segment obj))**

**(y-coord (end-segment obj))) 2)))**

**((eq? type 'frame)**

**(midpoint (make-segment (add-vector (origin-frame obj)**

**(edge1-frame obj))**

**(add-vector (origin-frame obj)**

**(edge2-frame obj)))))**

**(else #f))))**

### Problem 4 (Tree recursion)

**(define (max-path tree)**

**(if (null? (children tree))**

**(datum tree)**

**(+ (datum tree)**

**(biggest (map max-path (children tree))))))**

### Problem 5 (Scheme-1 interpreter)

I think the error in one line 11:

**(map eval-1 (cdr exp))))**

the student forgot to eval arguments, so he(she) wrote:

**(cdr exp)))**

### Problem 6 (Data directed programming)

**(define (transition fsm state letter)**

**(cond ((null? fsm) 0)**

**((and (= (caar fsm) state)**

**(eq? (cadar fsm) letter))**

**(caddar fsm))**

**(else (transition (cdr fsm) state letter))))**

**(define (process fsm state wd)**

**(if (empty? wd)**

**state**

**(process fsm (transition fsm state (first wd)) (bf wd))))**

### Problem 7 (Object oriented programming)

(a) Check the cases in which the first class shown should be the PARENT of the other class:

n cell phone / keypad

y building / office building

n stapler / staple

n arm / arm bone

y bone / arm bone

n person / arm bone

(b) For each of the following indicate CLASS VARIABLE or INSTANCE VARIABLE:

number of taxis in city - CLASS

taxi - INSTANCE

fridge - CLASS

number of milk cartons in fridge - INSTANCE