

MASSACHUSETTS INSTITUTE OF
TECHNOLOGY
Department of Electrical Engineering and
Computer Science
6.001—Structure and Interpretation of
Computer Programs
Fall Semester, 2001

Quiz I

NAME:

Section Number:

Tutor's Name:

PART	Value	Grade	Grader
1	16		
2	24		
3	26		
4	14		
5	20		
Total	100		

If you wish to write a note on any problem, put a check in the box next to the problem number, and use the space provided at the end of the answer sheet. *Notes written outside the answer boxes on this page will be ignored.*

PART 1 (16 points)

- 1.(2) Value:3
 Type:Number
- 2.(2) Value:error – not a procedure
 Type:
- 3.(2) Value:primitive
 Type:number, ..., number \mapsto number
- 4.(2) Value: -1
 Type:Number
- 5.(2) Value: compound
 Type: number, number \mapsto number
- 6.(2) Value: compound
 Type: number \mapsto (number \mapsto number)
- 7.(2) Value:4
 Type: number
- 8.(2) Value: (27 9)
 Type:List(number, number)

PART 2 (24 points)

9.(6)

```
(cond ((null? rest) best)
      ((< best (car rest))
       (aux best (cdr rest)))
      (else (aux (car rest) (cdr rest))))
```

10.(6)

```
(define (remove elt lst)
  (cond ((null? lst) nil)
        ((= elt (car lst)) (cdr lst))
        (else
         (cons (car lst)
               (remove elt (cdr lst))))))
```

- 11.(4) Time: B
 Space: A
- 12.(4) Time: B
 Space: B
- 13.(4) Time: D
 Space: B

PART 3 (26 points)

14.(8)

```
(define (reverse lst)
  (if (or (null? lst)
        (not (pair? lst)))
      lst
      (append (reverse (cdr lst))
              (list (car lst)))))
```

15.(8)

```
(define (ireverse lst)
  (define (help sofar todo)
    (if (null? todo)
        sofar
        (help (cons (car todo) sofar)
              (cdr todo))))
  (help nil lst))
```

- 16.(2) A
 17.(2) E
 18.(2) A
 19.(2) B
 20.(2) C

PART 4 (14 points)

- 21.(3) 49
 22.(3) 15
 23.(4) 27
 24.(4) 175

PART 5 (20 points)

25.(5)

```
(lambda (x) (not (lessthan (extract x) (extract (car lst)))))
```

26.(5)

```
(lambda (x) (lessthan (extract x) (extract (car lst))))
```

27.(4)

```
(general-sort test-vectors < xcoord)
```

28.(6)

```
(general-sort test-vectors
  (lambda (v1 v2)
    (cond ((< (xcoord v1) (xcoord v2)) true)
          ((= (xcoord v1) (xcoord v2)) (< (ycoord v2) (ycoord v1)))
          (else false)))
  (lambda (x) x))
```