

```

1 #lang racket
2
3 (struct empty-list () #:transparent)
4
5 (define empty (empty-list))
6 empty
7
8 (struct cons-list (v prev) #:transparent)
9
10 (cons-list 5 empty)
11
12 ;(cons-list 5 5)
13
14 (define (make-list args)
15   (match args
16     ((list)(empty-list))
17     ((list i (? empty-list? x))(cons-list i x))
18     ((list i (? cons-list? x))(cons-list i x))
19     ((list i)(cons-list i (empty-list)))
20     (_ (printf "Error: not a valid list!\n"))))
21
22 (make-list (list ))
23 (make-list (list 1))
24 (make-list (list 2 empty))
25 (make-list (list 1 2 3 4))
26
27 (define (get-first-list lst)
28   (match lst
29     ((? empty-list? lst) (printf "Error: not
29 long enough!\n"))
30     ((cons-list v rest) v)))
31
32 (define alpha (make-list (list "c" (make-list
32 (list "b" (make-list (list "a"))))))))
33 alpha

```

```

34 (get-first-list alpha)
35
36 (define (get-rest-list lst)
37   (match lst
38     ((? empty-list? lst) (printf "Error: not
38 long enough!\n")))
39     ((cons-list v rest) rest)))
40
41 (get-rest-list alpha)
42
43 (define (get-item-index lst i)
44   (letrec ((helper (lambda (lst i j)
45                     (if (= i j)
46                         (get-first-list lst)
47                         (helper (get-rest-list
47 lst) i (+ j 1))))))
48     (helper lst i 0)))
49 (get-item-index alpha 1)
50
51 (define (length-list lst)
52   (cond ((empty-list? lst) 0)
53         ((cons-list? lst) (+ 1
54                               (length-list
54 (get-rest-list lst))))
55         (else (printf "Error: not a list!"))))
56
57 (length-list alpha)
58
59 (define (length-list-1 lst)
60   (letrec ((helper (lambda (l len)
61                     (if (empty-list? l)
62                         len
63                         (helper (get-rest-list
63 l)
64                               (+ 1 len))))))

```

```
65     (helper lst 0)))
66
67 (length-list-1 alpha)
68
69 (define (length-list-2 lst)
70   (match lst
71     ((empty-list) 0)
72     ((cons-list _ rest) (+ 1 (length-list-2
72 rest)))
73     (_ "Error: not a list!")))
74
75 (length-list-2 alpha)
```