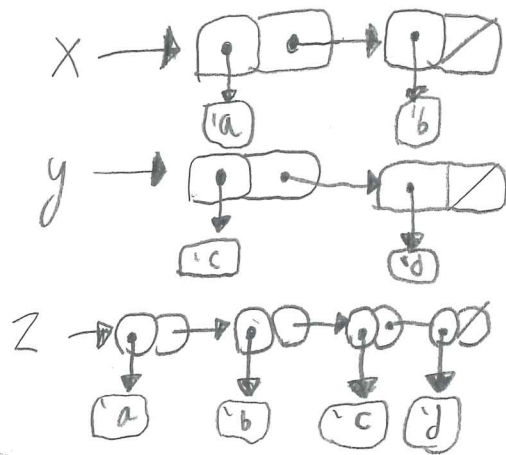


3.12

```

(def x (list 'a 'b))
(def y (list 'c 'd))
(def z (append x y))
z
(a b c d)
(cdr x)
(b)

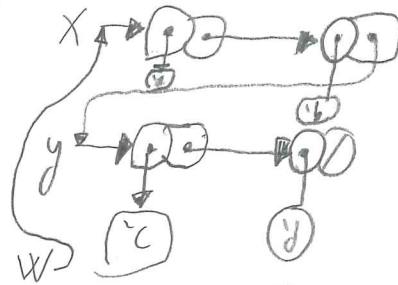
```



```

(define w (append! x y))
(w)
(cdr x)
(b c d)

```

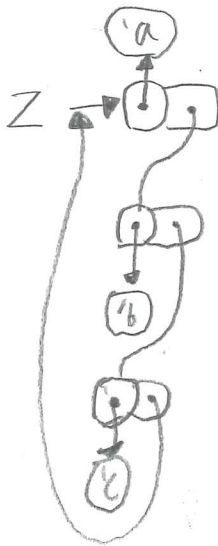


3.13

```

(define z (make-cycle (list 'a 'b 'c)))

```



If we try to compute

(last-pair z), we enter an infinite loop.

3.14

Mystery returns the reversed list of X (elements in reverse order, still () as ending element etc), and destructively updates X in the process.

