

```
1 Buat program menghitung panjang list, tanpa
... pre-defined.
2 > (panjang '(2 3 5))
3 3
4 (define (panjang x)
5   (if (empty? x) 0
6       (+ 1 (panjang (cdr x))))
7   )
8 )
9
10 (panjang '(2 3 5))
11 = (+ 1 (panjang '(3 5)))
12 = (+ 1 (+ 1 (panjang '(5))))
13 = (+ 1 (+ 1 (+ 1 (panjang '()))))
14 = (+ 1 (+ 1 (+ 1 0)))
15 = (+ 1 (+ 1 1))
16 = (+ 1 2)
17 = 3
18
19 Buat program gabung dua list x dan y, tanpa
... menggunakan "append"
20 > (gabung '(1 2) '(3 5))
21 (1 2 3 5)
22
23   (gabung '(2) '(3 5)) = (2 3 5)
24   (gabung (cdr x) y)
25
26   (cons 1 '(2 3 5)) = (1 2 3 5)
27   (cons (car x) (gabung (cdr x) y))
28
29   Jika x kosong, hasilnya adalah y
30
```

```
31 (define (gabung x y)
32   (if (empty? x) y
33       (cons (car x) (gabung (cdr x) y))
34   )
35 )
36
37 (gabung '(1 2) '(3 5))
38 = (cons 1 (gabung '(2) '(3 5)))
39 = (cons 1 (cons 2 (gabung '() '(3 5))))
40 = (cons 1 (cons 2 '(3 5)))
41 = (cons 1 '(2 3 5))
42 = (1 2 3 5)
43
44 Buat program mengambil sublist mulai elemen
... ke-a sebanyak n.
45 > (sublist '(2 3 5 7 8) 2 3)
46 (3 5 7)
47 > (sublist '(2 3 5 7 8) 2 0)
48 ()
49
50 (define (sublist x a n)
51   (if (> a 1)
52       (sublist (cdr x) (- a 1) n)
53       (if (= n 0) '()
54           (cons (car x) (sublist (cdr x) a (-
... n 1))))
55   )
56 )
57 )
58
59 (sublist '(3 5 7 8) 1 3) = (3 5 7)
60
```

```
61 (sublist '(5 7 8) 1 2) = (5 7)
62 (sublist (cdr x) a (- n 1))
63
64 (cons 3 '(5 7)) = (3 5 7)
65 (cons (car x) (sublist (cdr x) a (- n 1)))
66
67 (sublist '(2 3 5 7 8) 2 3)
68 = (sublist '(3 5 7 8) 1 3)
69 = (cons 3 (sublist '(5 7 8) 1 2))
70 = (cons 3 (cons 5 (sublist '(7 8) 1 1)))
71 = (cons 3 (cons 5 (cons 7 (sublist '(8) 1 0))))
72 = (cons 3 (cons 5 (cons 7 '())))
73 = (cons 3 (cons 5 '(7)))
74 = (cons 3 '(5 7))
75 = '(3 5 7)
76
```