

## LÖSUNG ZUR ÜBUNG 2.25 – ABELSON UND SUSSMAN, STRUKTUR UND INTERPRETATION VON COMPUTERPROGRAMMEN (SICP)

[HTTPS://GITHUB.COM/PZUEHLKE](https://github.com/pzuehlke)

*Übung 2.25:* Geben Sie Kombinationen von `car` und `cdr` an, die die 7 aus jeder der folgenden Listen herauspicken:

- (a) `liste1 = (1 3 (5 7) 9)`
- (b) `liste2 = ((7))`
- (c) `liste3 = (1 (2 (3 (4 (5 (6 7)))))`

*Lösung.*

- (a) `(car (cdr (car (cdr (cdr (liste))))))`. Denn es gilt:
  - `(cdr liste1) = (3 (5 7) 9)`
  - `(cdr (cdr liste1)) = ((5 7) 9)`
  - `(car (cdr (cdr liste1))) = (5 7)`
  - `(cdr (car (cdr (cdr liste1)))) = (7)`
  - `(car (cdr (car (cdr (cdr liste1)))) = 7`
- (b) `(car (car liste2))`
- (c) Man beachte zuerst, dass die Elemente dieser Liste nicht einfach die Zahlen  $1, 2, \dots, 7$  sind, sondern die Zahl 1 und eine andere Liste. Die letztere enthält die Zahl 2 und eine andere Liste, usw.. Wir brauchen fünf Anwendungen von `cadr` um (6 7) zu erreichen, und dann noch einen `cadr` um die 7 herauszupicken:

`(cadr (cadr (cadr (cadr (cadr (cadr liste)))))`