

Übungsaufgaben

Aufgabe 1

Schreibe eine Funktion, die die Länge einer Liste ermittelt.

```
(länge '(1 2 3 4 5)) → 5
```

Aufgabe 2

Schreibe eine Funktion, die zählt, wie oft ein Element in einer Liste auftaucht.

```
(zähle 3 '(4 3 2 1 3)) → 2
```

Aufgabe 3

Schreibe eine Funktion, die zwei sortierte Listen nimmt und diese zu einer sortierten Liste verbindet.

```
(merge '(2 3 4) '(1 5)) → '(1 2 3 4 5)
```

Aufgabe 4

Schreibe eine Funktion, die zwei Listen zurück gibt. Eine soll dabei alle übergebenen geraden Zahlen, eine alle ungeraden Zahlen enthalten.

```
(teile '(1 2 3 4 5 6)) → '((2 4 6) (1 3 5))
```

Aufgabe 5

Schreibe eine Funktion, die für die Zahlen in einer Liste den Mittelwert berechnet.

```
(mittelwert-1 '(2 2 4 4)) → 3
```

Aufgabe 6

Schreibe eine Funktion, die für eine Liste von Listen Komponentenweise den Mittelwert berechnet. Dabei sollen alle Unterlisten die gleiche Länge besitzen.

```
(mittelwert-2 '((10 5 3) (6 3 3) (2 7 -3))) → '(6 5 1)
```

Aufgabe 7

Schreibe die Funktion `append` selbst und beschreibe, wo das Problem ist, wenn etwas an eine lange Liste hängen möchte.

```
(append '(1 2) '(3 4 5 6)) → '(1 2 3 4 5 6)
```

Aufgabe 8

Schreibe eine Funktion, die für eine Zahl ihre Binärdarstellung berechnet.

```
(binär 42) → '(1 0 1 0 1 0)
```

Aufgabe 9

Schreibe eine Funktion, die prüft, ob es sich bei einer Liste um ein Palindrom handelt. Dabei sollte die Laufzeit in $O(n)$ liegen.

```
(palindrom? '(1 2 5 2 1)) → #t
```