

**Präsenzübungen zur Vorlesung**  
**Deklarative Programmierung: Sommersemester 2022**  
Nr. 7

---

**Aufgabe 7.1:** Bedeutung von Datendefinitionen

- a) Was ist das Datenuniversum eines Programms?
- b) Was bedeutet eine Strukturdefinition für das Datenuniversum?
- c) Wozu benötigen wir Datendefinitionen? Was drücken diese aus?
- d) Finden Sie ein geeignetes Beispiel einer Strukturinstanz, die zwar im Datenuniversum des Programms enthalten ist, aber nicht der (zu erstellenden) Datendefinition entspricht.

**Aufgabe 7.2:** Refactoring algebraischer Strukturen

Betrachten Sie folgende Struktur:

```
(define-struct student-flat  
  (matno firstname lastname street zipcode city  
    phone mobil email bday bmon byear))
```

- a) Was ist der Nachteil einer solch flachen Struktur?
- b) Definieren Sie eine **isomorphe** Struktur (und ggf. Teilstrukturen), die diese Informationen „besser“ darstellt.
- c) Implementieren Sie eine **bijektive** Funktion, die die flache Struktur in die neue überführt.
- d) Implementieren Sie Funktionen, die diese Strukturen als String darstellen.

**Aufgabe 7.3:** Rekursion

Implementieren Sie die folgenden Funktionen für Listen ganzer Zahlen. Geben Sie zu jeder Funktion die Signatur mit korrektem Gebrauch von Typvariablen an. Geben Sie außerdem jeweils mindestens zwei Testfälle (mittels `check-expect`) an.

- a) `(list-size l)`, die die Länge der Liste berechnet.
- b) `(list-contains e l)`, die überprüft, ob sich `e` in der Liste `l` befindet.
- c) `(list-max l)`, die das größte Element der Liste findet und zurückliefert.
- d) `(list-tail l)`, die die übergebene List ohne das erste Element zurückgibt. Falls die Liste keine Elemente enthält, soll die leere List zurückgegeben werden.
- e) `(even-sum n)`, die die Summe aller geraden Zahlen zwischen 0 und der natürlichen Zahl `n` berechnet.
- f) Terminieren diese Funktionen immer? Begründen Sie Ihre Antwort.

### Aufgabe 7.4: Das „S“ steht für Hoffnung!

In dieser Aufgabe sollen Sie einige Datenstrukturen erzeugen.

- a) Übersetzen Sie das folgende Struct in eine S-Expression:

```
(make-transformographagizer "Oreobot "  
  (make-machine "Oreo-vending-machine")  
  (make-enemy "Soundwave by Dre")  
  (make-ally "Optimus Prime"))
```

- b) Erzeugen Sie eine S-Expression für Autos.  
Beachten Sie die Felder „Marke“, „Modell“ und „Farbe“.
- c) Erzeugen Sie Ihren Lieblingssuperhelden in Racket. Die Struktur für Superhelden soll den Namen „Held“ und mindestens den Superheldennamen sowie den bürgerlichen Namen enthalten.  
Geben Sie den erzeugten Feldern sinnvolle Namen.

### Aufgabe 7.5: Eins zwei Polizei – Mo-Do

*Bereits mit uns ist der Regen, / schüttelt die stille Luft.* – Salvatore Quasimodo

- a) Definieren Sie eine Struktur (`quadrangle`) mit zugehöriger Datendefinition (`Quadrangle`) zur Darstellung eines Vierecks. Zur Darstellung der vier Eckpunkte soll die Struktur `posn` verwendet werden. Implementieren Sie weiterhin eine Funktion (`define (create-square tl br) ...`), welche zwei gegenüberliegende Eckpunkte als Argument erhält (tl: top-left und br: bottom-right, jeweils `posn`-Instanzen) und eine entsprechende Instanz von `quadrangle` zurück gibt. (Denken Sie an das Entwurfsrezept.)
- b) Überlegen Sie sich eine Codierung als S-Expression für den Datentyp `Quadrangle`. Berücksichtigen Sie bei der Definition bereits Teilaufgabe d) und verwenden Sie eine Codierung, die für diese Teilaufgabe sinnvoll ist. Verwenden Sie zur Erstellung von Beispiel-Werten Quoting.
- c) Implementieren Sie die Funktion (`define (create-square2 tl br) ...`), welche das Viereck in der Darstellung als S-Expression aus der vorigen Teilaufgabe zurückgibt. Verwenden Sie hierzu Quasiquoting.
- d) Implementieren Sie eine weitere Funktion (`define (perimeter quadrangle) ...`), welche den Umfang des Vierecks (also die Summe der Länge aller vier Kanten) berechnet. Definieren Sie hierzu eine Hilfsfunktion, welche eine Liste von Punkten erhält und jeweils die Entfernungen dazwischen aufsummiert. Diese Hilfsfunktion soll neben der Liste noch einen Punkt als Argument erhalten. Für den letzten Punkt in der Liste soll dann die Entfernung zu diesem Argument berechnet werden.