

Übungen zur Vorlesung
Deklarative Programmierung: Sommersemester 2018

Nr. 2, Abgabe bis 30.4.2018 11:59 Uhr

Aufgabe 2.1: Programmfehler

4 Punkte

Prüfen Sie, ob die folgenden Ausdrücke eine Bedeutung in der BSL haben. Falls nicht, geben Sie den Typ (Syntax- oder Laufzeitfehler) und den Grund des Fehlers an.

- a) `(+ (string-length "KvP") (- 17 4))`
- b) `(* 2 (string-length (string-append "Die Zahl ist" 5)))`
- c) `(/ (image-width (circle 50 "solid" "red")) 2)`
- d) `(/ (+ 23 7) (- 16 (* 4 4)))`

Aufgabe 2.2: Konditionale Ausdrücke

4 Punkte

Programmieren Sie eine Funktion:

- a) `(max3 a b c)`, die das Maximum der drei übergebenen Zahlen zurückliefert. Verwenden Sie dazu einen verschachtelten `if`-Ausdruck.
- b) `(rome num)`, die die römische Darstellung der übergebenen Zahl zurückliefert. Sie müssen dabei lediglich die Zahlen 1-9 betrachten, für alle anderen Werte soll die Funktion den String "Nicht implementiert" liefern. Beispiel: `(rome 4) = "IV"`.

Aufgabe 2.3: DRY!

4 Punkte

Betrachten Sie die folgende Funktion, welche 5 Kreise in einer Szene platziert:

```
(define (draw color)
  (place-image (circle 20 "solid" color) 0 0
    (place-image (circle 20 "solid" color) 100 0
      (place-image (circle 20 "solid" color) 100 100
        (place-image (circle 20 "solid" color) 0 100
          (place-image (circle 20 "solid" color) 50 50
            (empty-scene 100 100))))))
)
```

Schreiben Sie die Funktion so um, dass sie keine *Redundanzen* oder *magic numbers* mehr enthält. Definieren Sie dazu geeignete Konstanten und ggf. Hilfsfunktionen.

Hinweis: Den Programmcode finden Sie in der Datei `dry.rkt`.

Aufgabe 2.4: Rocket reloaded

4 Punkte

In Kapitel 2.6.2 des Skripts haben wir die Funktion `create-rocket-scene-v7` implementiert. Übergeben wir diese an die `animate`-Funktion, landet die Rakete sicher auf dem Boden. Nun möchten wir die Funktion so erweitern, dass die Rakete in einer Dauerschleife vom Boden abhebt, bis zum oberen Rand der Szene fliegt und anschließend wieder landet. Der Parameter `height` muss also so abgebildet werden, dass er, unabhängig des übergebenen Werts, wiederholt von `HEIGHT` bis 0 und anschließend wieder bis `HEIGHT` zählt.

- a) Schreiben Sie dazu zunächst eine Funktion (`bounds height`), die die übergeben Höhe geeignet auf den Bereich `(-HEIGHT,HEIGHT]` abbildet.

Hinweis: Schauen Sie sich die Dokumentation der (`modulo n m`)-Funktion an (<https://docs.racket-lang.org/reference/generic-numbers.html>).

- b) Die Höhe der Rakete ergibt sich aus dem Absolutwert des Ergebnisses der `bounds`-Funktion. Erstellen Sie eine Funktion `create-rocket-scene-v8`, indem Sie den ursprünglichen Code entsprechend anpassen. Testen Sie Ihre Implementierung mithilfe von `animate`.

Hinweis: Den Programmcode zu `create-rocket-scene-v7` finden Sie in der Datei `rocket.rkt`.