

Übungen zur Vorlesung
Deklarative Programmierung: Sommersemester 2018

Nr. 9, Abgabe bis 2.7.2018 11:59 Uhr

Hinweis: Ändern Sie den Sprachlevel der DrRacket-Umgebung auf „Die Sprache Racket“. Ändern Sie die erste Zeile Ihres Programmes auf "#lang racket".

Aufgabe 9.1: Nickel Cadmium

4 Punkte

Die Fibonaccifolge ist wie folgt definiert:
$$fib(n) := \begin{cases} 1, & 1 \leq n \leq 2 \\ fib(n-1) + fib(n-2) & \text{sonst} \end{cases}$$

- a) Implementieren Sie eine Funktion (`fib n`), die das n-te Element der Folge nach der gegebenen Vorschrift berechnet.
- b) Die Berechnung kann erheblich beschleunigt werden, indem wir zwei Akkumulatoren einführen, die jeweils $fib(n-1)$ und $fib(n-2)$ enthalten. Implementieren Sie eine Funktion (`fib/fast n`), die das n-te Element der Folge mithilfe einer lokalen (akkumulatorgestützten) Funktion berechnet.
- c) Messen Sie die Laufzeit der beiden Implementierungen für verschiedene Werte von n. Verwenden Sie dazu die Funktion `time`. Wie erklären Sie sich die erheblichen Unterschiede in der Laufzeit?

Hinweis: Fügen sie "(require 2htdp/abstraction)" zu Ihrem Programm hinzu.

Aufgabe 9.2: ADT Security

4 Punkte

Viele Fachbereiche schreiben den Studierenden eine bestimmte Zitierweise vor, um in ihren Ausarbeitungen auf andere Forschungsartikel und Bücher zu verweisen. In dieser Aufgabe entwickeln wir ein Programm um Studierende beim Zitieren zu unterstützen. Folgende Richtlinien sollen umgesetzt werden:

Bücher: Autor. Titel. Verlag, Ort Jahr

Artikel aus Zeitschriften: Autor. Titel. Name der Zeitschrift, Jahr

Konferenzbeiträge: Autor. Titel. In Name der Konferenz. Verlag, Ort Jahr

- a) Nutzen Sie `define-type` um einen geeigneten algebraischen Datentyp für Referenzen zu erstellen.
- b) Implementieren Sie eine Funktion (`cite reference`); `Reference -> String`, welche die übergebene Referenz nach den Richtlinien zitiert. Nutzen Sie bei der Implementierung das `type-case`-Konstrukt.

Hinweis: Fügen sie "(require 2htdp/abstraction)" zu Ihrem Programm hinzu.

Aufgabe 9.3: This was not in my contract!

4 Punkte

Betrachten Sie die folgenden Funktionen:

```
; Fügt die Liste l2 hinter die List l1 an
(check-expect (lst-append '(1 2) '(4 3)) '(1 2 4 3))
(check-expect (lst-append '(1 2) '()) '(1 2))
(define (lst-append l1 l2)
  (match l1
    ['() (first l2)]
    [(cons x '()) (cons x l2)]
    [(cons x xs) (cons x (lst-append xs l2))]))

; Gibt das Resultat zurück, dass sich aus dem Vergleich der beiden
; Elemente in pair mittels dem Operator comparator ergibt.
(check-expect (f = '(1 2)) false)
(define (f comparator pair)
  (comparator (first pair) (second pair)))
```

- Geben Sie für die oben genannten Funktionen jeweils eine Implementierung mit einem Contract an. Schreiben Sie für jede Funktion einen Aufruf, bei dem der Contract verletzt wird. Wer ist verantwortlich für den Fehler?
Hinweis: Namen von Lebewesen und/oder Organisationen (real existierend oder erfunden) sind nicht als Antwort zulässig.
- Geben Sie für die oben genannten Funktionen jeweils eine formale Signatur an (also mit typed Racket).
Ändern Sie die erste Zeile ihres Programmes auf „#lang typed/racket“.
- Sofern der Typcheck der Funktion in Aufgabenteil b) fehlschlägt, geben Sie eine korrigierte Fassung der Funktion an.
- Geben Sie zu jeder Funktion einen Aufruf an, der die formale Signatur verletzt (darf derselbe wie in Aufgabenteil a) sein) und erläutern Sie den Fehler des Typcheckers.

Aufgabe 9.4: Oberst von Gatow mit dem Kerzenleuchter im Keller

4 Punkte

Gegeben seien 6 Personen, die verschiedene Sportarten betreiben und an verschiedenen Wochentagen Zeit haben.

- Anja spielt Tennis und hat montags und dienstags Zeit.
- Bernd spielt Golf und Tennis und hat dienstags Zeit.
- Claudia spielt Billard und Tennis und hat donnerstags Zeit.
- Daniel spielt Billard und hat mittwochs Zeit.
- Elke spielt Golf und hat freitags und samstags Zeit.
- Frank spielt Tennis und Billard und hat donnerstags Zeit.

Stellen sie diese Informationen als Prolog-Fakten dar und formulieren geeignete Anfragen:

- a) Welche Sportarten beherrscht Frank?
- b) Welche Sportarten spielen sowohl Anja als auch Claudia?
- c) Wer spielt Billard?
- d) Definieren sie ein Prädikat `spiel(Person,Person,Sportart,Zeitpunkt)`, das die Relation `Person x Person x Sportart x Zeitpunkt` mit obigen Prolog-Fakten realisiert. Welche Spiele kommen zustande?

Hinweis: Für die Bearbeitung dieser Aufgabe benötigen Sie tuprolog (<https://bitbucket.org/tuprologteam/tuprolog/downloads/2p-3.2.1.zip>).
Ungleichheit kann in Prolog mit `X \== Y` ausgedrückt werden.