

Übungsblatt 2

Prof. Dr. Frank Deinzer, Frank Ebner
Hochschule für angewandte Wissenschaften Würzburg-Schweinfurt
Fakultät für Informatik & Wirtschaftsinformatik FHWS

Algorithmen und Datenstrukturen I
Wintersemester 2019/2020

Vorbemerkung:

Bitte loggen sie sich zur Abgabe der Übungen mit Ihren „Active Directory“ Daten in das Abgabesystem unter

<https://algo.welearn.de>

ein. Testen sie ihre Übung vorher mit DrRacket. Die Aufgaben (Code und Ausgabe) werden automatisiert überprüft, Punkte gibt es also nur für 100% korrekte Lösungen mit korrekten Prozedurnamen und Variablen die eine korrekte Ausgabe liefern. Unter Quellcode können sie ihre Ausdrücke und Prozeduren definieren. Unter Eingabe können sie ihre Testdaten eingeben. Mit „speichern & testen“ werden Quellcode und Eingabe ausgeführt, die evaluierte Ausgabe erscheint im Ausgabefenster. Die Ergebnisse werden in einer Datenbank gespeichert und nach Ablauf des Abgabetermins überprüft.

Hinter der Aufgabennummer finden Sie in Klammern eine Angabe der Schwierigkeitsstufe der jeweiligen Aufgabe, wobei * = leicht, ** = mittel und *** = schwer, gilt.

Aufgaben vom Typ „Tafelübung“ werden nicht über das System abgegeben, sondern sollen von Ihnen schriftlich (Stift und Papier) bearbeitet werden.

Algorithmik//Aufgabenbearbeitung

Aufgabe 1

speichern & testen Quelltext speichern

Quellcode:

```
(define (mysum x y) (+ x y))
```

Eingabe:

```
(mysum 3 4)
```

Ausgabe:

```
7
```

Aufgabe 7 (6 Punkte) (**)

Es gibt folgenden Algorithmus zur Berechnung der Quadratwurzel einer natürlichen Zahl:

1. Summiere die ungeraden Zahlen ab 1 bis die Summe gleich dem Parameter ist.
2. Zähle die Summanden. Diese Zahl ist die gesuchte Quadratwurzel

| Ungerade Zahlen | Summe der ungeraden Zahlen | Anzahl von Summanden | Parameter |
|-----------------|----------------------------|----------------------|--------------|
| 1 | 1 | 1 | 1 (1^2) |
| 3 | 4 (1+3) | 2 (1,3) | 4 (2^2) |
| 5 | 9 (1+3+5) | 3 (1,3,5) | 9 (3^2) |
| 7 | 16 (1+3+5+7) | 4 (1,3,5,7) | 16 (4^2) |
| 9 | 25 (1+3+5+7+9) | 5 (1,3,5,7,9) | 25 (5^2) |
| 11 | 36 (1+3+5+7+9+11) | 6 (1,3,5,7,9,11) | 36 (6^2) |

Übungsblatt 2

Prof. Dr. Frank Deinzer, Frank Ebner
Hochschule für angewandte Wissenschaften Würzburg-Schweinfurt
Fakultät für Informatik & Wirtschaftsinformatik FHWS

Algorithmen und Datenstrukturen I
Wintersemester 2019/2020

Geben Sie eine korrekte Prozedur `nat-wurzel` an, die diesen Algorithmus implementiert.

```
(define (nat-wurzel x)
  ...)
```

Testen Sie Ihre Prozedur mit einer ausreichend großen Anzahl an Beispielen.

Aufgabe 8 (6 Punkte) (***)

Schreiben Sie eine Prozedur `zahl-umdrehen`, die die Ziffernreihenfolge einer eingegebenen natürlichen Zahl umdreht und ausgibt:

```
(define (zahl-umdrehen x)
  ...)
```

| Eingegebene Zahl | Ausgabe: Umgedrehte Zahl |
|------------------|--|
| 123 | 321 |
| 4497821 | 1287944 |
| 597050 | 50795 (erste Null muss nicht angezeigt werden) |

Testen Sie Ihre Prozedur mit einer ausreichend großen Anzahl an Beispielen.

Hinweis: In Scheme gibt es vordefinierte Prozeduren zur ganzzahligen Division und zur Berechnung des Rests einer ganzzahligen Division:

```
32 : 5 = 6 Rest 2      ➔      > (quotient 32 5)
                                6
                                > (remainder 32 5)
                                2
```

Aufgabe 9 (4 Punkte) (**)

Definieren Sie ein Prädikat `aufsteigendes-produkt?`, das für 4 Zahlen `a b c d` testet, ob die Zahlen in aufsteigender Reihenfolge angegeben sind und ob das Produkt aus `a`, `b` und `c` den Wert von `d` ergibt.

```
(define (aufsteigendes-produkt? a b c d)
  ... )
```

Beispiele:

Übungsblatt 2

Prof. Dr. Frank Deinzer, Frank Ebner
Hochschule für angewandte Wissenschaften Würzburg-Schweinfurt
Fakultät für Informatik & Wirtschaftsinformatik FHWS

Algorithmen und Datenstrukturen I
Wintersemester 2019/2020

| | | |
|-----------------------------------|---|----|
| (aufsteigendes-produkt? 1 2 3 6) | → | #t |
| (aufsteigendes-produkt? 2 1 3 6) | → | #f |
| (aufsteigendes-produkt? 2 3 5 11) | → | #f |

Aufgabe 10 (9 Punkte) (*)

Definieren Sie für jede der drei booleschen Funktionen eine Scheme-Prozedur:

$$f1(a, b) = \overline{(a \vee b)} \wedge (a \vee b) \wedge a \wedge \overline{b}$$

$$f2(a, b, c) = a \vee (a \wedge b \wedge \overline{c}) \vee (\overline{a} \wedge c) \vee (\overline{a} \wedge \overline{b} \wedge c)$$

$$f3(a, b, c, d) = (\overline{a} \oplus b) \wedge (a \vee \overline{b} \vee c) \wedge (\overline{d} \vee \overline{c} \vee \overline{b} \vee \overline{a})$$

```
(define (f1 a b)... )  
(define (f2 a b c) ...)  
(define (f3 a b c d) ...)
```

Tafelübung T2

- a) Bearbeiten Sie bitte Aufgabe 1.6 im Buch „Structure and Interpretation of Computer Programs“ (<http://mitpress.mit.edu/sicp/full-text/book/book.html>).
- b) Sind die folgenden vier Aussagen richtig oder falsch? Begründen Sie Ihre Antwort!

$$\forall x \in \mathbb{N}_0 : x^2 > 0$$

$$\forall x \in \mathbb{R} : x > -x$$

$$\exists x \in \{3, 5, 9, 17\} : \frac{x}{2} \in \mathbb{N}$$

$$\exists x \in \{3, 34, 21, 15, 16\} : x = 2k, k \in \mathbb{N}_0$$

Abgabe der Übung, also Eingabe ins System unter <https://algo.welearn.de> bis Montag, 04.11.2019, 23:59.