

# Übung 4

## Algorithmen und Programmierung

Jonas Henschel, Jens Pönisch, Dominik Gorgosch, Bastian Felix Bachmann, Arvid Horn, Billy Naumann

**Hinweis:** Bearbeiten Sie nicht alle Aufgaben in der Reihenfolge wie sie hier sind, dass könnte dazu führen, dass Sie nicht alle Aufgaben schaffen. Es ist besser aus allen Bereichen mind. eine Aufgabe zu bearbeiten und dann zu entscheiden, wo Sie Übungsbedarf haben.

### Aufgabe 1 (Typen)

- a) Schreiben Sie eine *rekursive* Funktion, die nacheinander alle kleingedruckten Buchstaben des Alphabetes von a bis z ausgibt.
- b) Diskussion: Überlegen Sie wie Sie bestimmen könnten, ob eine Zahl durch eine andere Zahl ohne Rest geteilt werden kann. Dabei sind beide Zahlen positive Zahlen ungleich 0 **mit Nachkommaanteil**, z.B. 1.5. Sie müssen hier noch keine Funktion schreiben.
- c) Setzen Sie die Überlegungen zum Teilen ohne Rest mithilfe einer *rekursiven* Funktion um. Testen Sie ihre Funktion für die folgenden Eingabewerte: (6, 1.5), (2.8, 1.4), (120.66, 40.22), (8, 3.3), (12.4, 8.6).

(Hinweis: Da die `readInteger()` Funktion nur ganze Zahlen einlesen kann, rufen Sie diese Funktion innerhalb der `main()` Funktion mit expliziten Werten statt Variablen auf, z.B. `teilbar(6, 1.5);` anstatt `teilbar(a,b);`)

### Aufgabe 2 (Nutzung von Pointern)

- a) Erläutern Sie die Funktionsweise der bekannten Funktion `readInteger`.

```
1  int readInteger() {  
2      int i;  
3      scanf("%d", &i);  
4      return i;  
5  }
```

- b) Was gibt der folgende Programmabschnitt aus?

```
1  int *b, a = 13, c;  
2  b = &c;  
3  int *d = b;  
4  c = a;  
5  a = a + *d;
```

```

6  *b = *b + 4;
7  c = c + 1;
8  d = &a;
9  printf("%d %d %d %d\n", a, *b, c, *d);

```

- c) Schreiben Sie eine Funktion **swap**, die die Werte zweier ganzzahliger Variablen vertauscht.
- d) Schreiben Sie eine Funktion **sort**, die drei übergebene ganzzahlige Variablen von klein nach groß sortiert. Hinweis: Sie können und sollten die Funktion **swap** aus der vorherigen Teilaufgabe nutzen.
- e) Schreiben Sie eine rekursive Funktion **sum**, die sowohl Summe aller geraden Zahlen von  $1..n$  als auch die Summe aller ungeraden Zahlen von  $1..n$  berechnet und der rufenden Funktion zur Verfügung stellt.

### Aufgabe 3 (Statische Arrays)

- a) Schreiben Sie eine rekursive Funktion, welche ein übergebenes ganzzahliges Array der Reihe nach mit Nutzereingaben befüllt. Sie können davon ausgehen, dass das Array maximal 100 Elemente enthält. Die Anzahl der einzugebenden Elemente wird als Parameter übergeben.
- b) Was müsste an Ihrer Lösung geändert werden, damit die Nutzereingaben das Array verkehrtherum befüllen? (Sprich: die erste Nutzereingabe befüllt das letzte Element, usw.)
- c) Schreiben Sie eine rekursive Funktion, welche die Summe aller Elemente des befüllten Arrays zurückgibt.

### Aufgabe 4 (Dynamische Zeichenketten)

- a) Schreiben Sie eine rekursive Funktion, die eine übergebene positive ganze Zahl in eine Zeichenkette, die den Ziffern der Reihe nach entspricht, umwandelt. Die Zeichenkette soll von der Funktion zurückgegeben werden.
- b) Schreiben Sie eine rekursive Funktion, die eine Nutzereingabe in eine Zeichenkette speichert. Die Zeichenkette soll von der Funktion zurückgegeben werden. Die Eingabe soll durch das Drücken von Enter beendet werden.
- c) Schreiben Sie eine rekursive Funktion, die eine Zeichenkette in eine positive ganze Zahl umwandelt. Sie können davon ausgehen, dass in der Zeichenkette nur Zeichen zwischen '0' und '9' vorkommen.

### Aufgabe 5 (Zusatzaufgaben)

- a) Schreiben Sie eine rekursive Funktion, welche die ersten  $n$  Primzahlen in ein übergebenes Feld schreibt. Die Reihenfolge ist dabei egal. Es kann  $n \leq 19$  angenommen werden. Sie dürfen rekursive Hilfsfunktionen schreiben.
- b) Gegeben sei ein Array von natürlichen Zahlen zwischen 0 und 9. Schreiben Sie eine rekursive Funktion, welche dieses Array in eine ganze Zahl umwandelt. Dabei steht jede Zahl in dem Array für die Ziffer des Ergebnisses an der Stelle, die dem Index entspricht.

Beispiele:

- $\{0, 1, 2, 3, 4\} \rightarrow 1234$
- $\{1, 0, 0, 0, 1\} \rightarrow 10001$
- $\{5, 6, 1, 0, 3, 0\} \rightarrow 561030$