

# Algorithmen und Programmierung

## Übungsblatt 1

### WS 2022/23

Dr. Felix Jonathan Boes

Benedikt Bastin, Ellen Bundschuh, Anna Höpfner, Gina Muuss, Adrian Oeyen, Felix Roth,  
Thore Wolf

Ausgabe: 10.10.2022

Abgabe: keine Abgabe; Präsentation in der Übung

*Die erfolgreiche Teilnahme an den Übungen ist verpflichtend um die Klausurzulassung zu erhalten. Dazu gehört die Bearbeitung und erfolgreiche Vorstellung von regelmäßig erscheinenden Übungsblättern in Kleingruppen von bis zu drei Studierenden. Jedes Gruppenmitglied muss die eingereichten Lösungen von mindestens 70% der Übungsblätter erfolgreich präsentieren.*

**Aufgabe 1** (Notwendige Anmeldungen). Die folgenden Dienste müssen Sie erfolgreich einrichten und nutzen um an den Übungen teilnehmen zu können. Es liegt in Ihrer Verantwortung die Einrichtung rechtzeitig abzuschließen.

- Jede Woche finden Präsenztutorien in Kleingruppen der Größe drei statt. Jede Kleingruppe bekommt einen festen, dreißigminütigen Zeitraum zugeteilt. Zur Vergabe der Zeiträume verwenden wir das Tutorienvergabesystem (TVS)<sup>1</sup>. Dabei können Wünsche angegeben werden. Diese Wünsche können auf Grund der hohen Teilnehmerzahl nicht immer berücksichtigt werden. Es ist organisatorisch nicht möglich allen Teilnehmern einen Wunschtermin zuzuordnen. Melden Sie sich bis inklusive **Donnerstag der ersten Vorlesungswoche** über TVS zu den Übungen an. Die finale Zuteilung zu den Übungen wird am Montag der zweiten Vorlesungswoche bekannt gegeben.
- Die meisten Dienste der Informatik und des B-IT werden von der gemeinsamen Systemgruppe (GSG)<sup>2</sup> betrieben. Für die Nutzung der Informatikdienste wird ein Informatikaccount verwendet. Richten Sie sich Ihren Informatikaccount ein<sup>3</sup>.
- Melden Sie sich mit Ihrem Informatikaccount an einem Computer der Computerpools<sup>4</sup> an.
- Die Übungen werden mithilfe der zentralen gitlab-Instanz<sup>5</sup> organisiert. Melden Sie sich bei der zentralen gitlab-Instanz mit Ihrem Informatikaccount an. Die erfolgreiche Anmeldung ist nötig um ab der kommenden Woche an den Übungen teilzunehmen.

---

<sup>1</sup>Siehe [tvs.cs.uni-bonn.de](https://tvs.cs.uni-bonn.de)

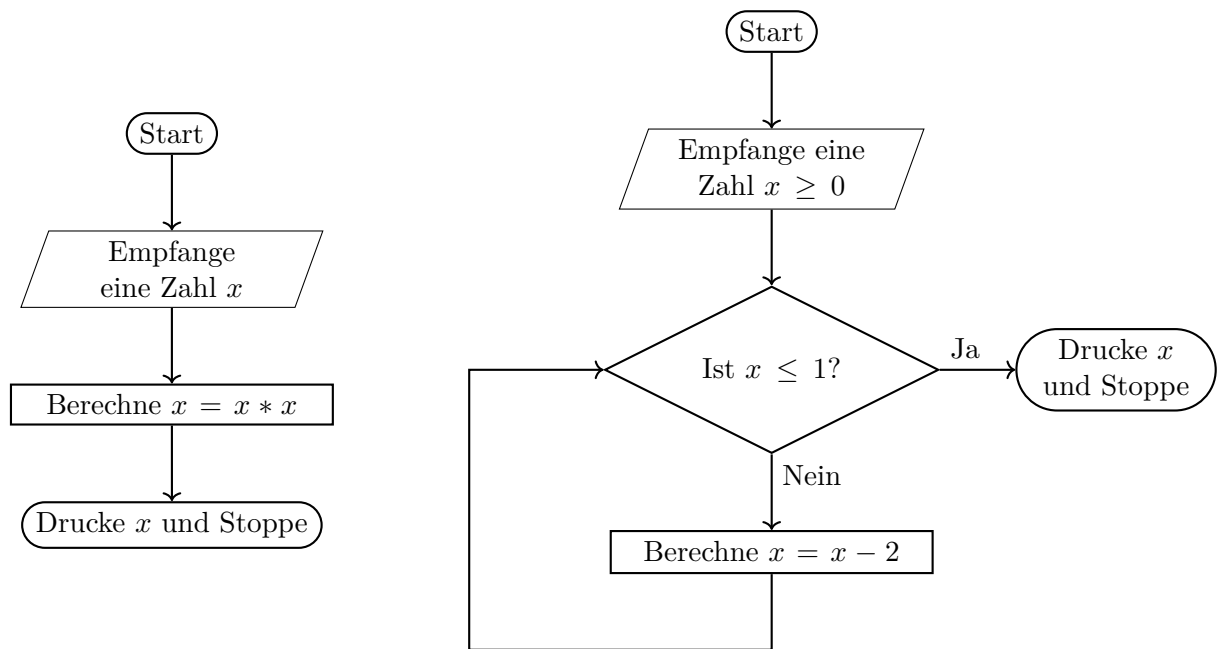
<sup>2</sup>Siehe <https://gsg.informatik.uni-bonn.de>

<sup>3</sup>Siehe <https://gsg.informatik.uni-bonn.de/doku.php?id=de:accounts:students:manage>

<sup>4</sup>Siehe <https://gsg.informatik.uni-bonn.de/doku.php?id=de:pool>

<sup>5</sup>Siehe [gitlab.informatik.uni-bonn.de](https://gitlab.informatik.uni-bonn.de)

**Aufgabe 2** (Was beschreiben diese Ablaufdiagramme?). Studieren Sie die folgenden Ablaufdiagramme<sup>6</sup>. Führen Sie diese Diagramme für einige Zahlenwerte (per Hand) aus. Welche Probleme werden durch die jeweiligen Programme gelöst? Begründen Sie Ihre Antwort nachvollziehbar und stellen Sie Ihre Lösung in der ersten stattfindenden Übung vor.



**Aufgabe 3** (Was beschreiben diese Programme?). Studieren Sie die folgenden Programme<sup>7</sup>. Führen Sie diese Programme für einige Zahlenwerte (per Hand) aus. Welche Probleme werden durch die jeweiligen Programme gelöst? Begründen Sie Ihre Antwort nachvollziehbar und stellen Sie Ihre Lösung in der ersten stattfindenden Übung vor. Wir beginnen mit einem Beispiel:

Listing 1: Programm\_1

```

1 Input: natürliche Zahl x
2 y = x
3 y = y + 17
4 y = y * 3
5 y = y modulo 2
6 drucke(y)

```

**Behauptung:** Programm\_1 druckt 0 aus immer wenn  $x$  ungerade ist und 1 immer wenn  $x$  gerade ist.

**Begründung:** In Zeile 2 ist  $y$  gerade genau dann wenn  $x$  gerade ist. In Zeile 3 ist  $y$  gerade genau dann wenn  $x$  ungerade ist. In Zeile 4 ist  $y$  gerade genau dann wenn  $x$  ungerade ist. In Zeile 5 ist  $y = 0$  genau dann wenn  $y$  gerade ist, also genau dann wenn  $x$  ungerade ist. Außerdem ist  $y = 1$  genau dann wenn  $y$  ungerade ist, also genau dann wenn  $x$  gerade ist. In Zeile 6 wird somit 0 ausgedruckt immer wenn  $x$  ungerade ist und 1 ausgedruckt immer wenn  $x$  gerade ist.

<sup>6</sup>In der Informatik verwendet man Variablen zur Benennung von Werten. Dabei werden Variablen immer wieder Werte zugewiesen. Die Zuweisung wird manchmal durch das Symbol  $:=$  und öfter durch  $=$  beschrieben.

<sup>7</sup>In der Informatik verwendet man Variablen zur Benennung von Werten. Dabei werden Variablen immer wieder Werte zugewiesen. Die Zuweisung wird manchmal durch das Symbol  $:=$  und öfter durch  $=$  beschrieben.

Listing 2: Programm\_2

```
1 Input:  Text T der aus genau n Zeichen T[0], T[1], ... besteht.
2 i = 0
3 s = 0
4 Erhöhe s um 1 falls T[i] ein Punkt, Ausrufezeichen oder Fragezeichen ist.
5 i = i+1
6 Springe zu Zeile 4 falls i < n ist; anderenfalls drucke(s).
```

Listing 3: Programm\_3

```
1 Input:  Liste von n+1 Zahlen L[0], L[1], L[2], ..., L[n]
2 i = 0
3 s = 0
4 s = s + L[i]
5 i = i+1
6 Springe zu Zeile 4 falls i <= n ist; anderenfalls drucke(s/(n+1)).
```

Listing 4: Programm\_4

```
1 Input:  Natürliche Zahl x
2 t = x
3 t = t*t
4 t = x*t
5 t = t*t
6 t = t*t
7 t = x*t
8 t = t*t
9 t = x*t
10 t = t*t
11 t = t*t
12 t = x*t
13 t = t*t
14 drucke(t)
```