



Prof. Dr. Thomas Wiemann

Algorithmen und Datenstrukturen

Übungsblatt 2

Wintersemester 2025/26

Aufgabe 2.1 (Verifikation)

Betrachten Sie den folgenden Algorithmus:

```
int n, x, y, z;
do { n = IO.readInt(); } while (n < 0);
x = 2; y = n; z = 1;
// Schleifeninvariante:  $z * x^y = 2n$  und  $y \geq 0$ 
while (y > 0) {
    if (y % 2 == 1) {
        z = z * x;
        y = y - 1;
    } else {
        x = x * x;
        y = y / 2;
    }
}
IO.println (z);
```

Welches Problem löst er und welchen Vorteil hat dieser Ansatz gegenüber einer naiven Implementierung? Weisen Sie mittels der in der Vorlesung vorgestellten Zusicherungsmethode die partielle Korrektheit mit der vorgegebenen Schleifeninvariante nach. Argumentieren Sie, warum er immer anhält, um die totale Korrektheit nachzuweisen.

Aufgabe 2.2 (Halteproblem)

Reproduzieren Sie den in der Vorlesung vorgestellten Beweis zum Halteproblem vor, d.h., das es kein Programm geben kann, welches entscheidet, ob ein anderes für einen gegebenen Input terminiert. Warum wurde das Bild der Fee gewählt, um die Argumentation gegen die Existenz eines Programms zur Lösung des Halteproblems zu argumentieren. Ist die Argumentation allgemeingültig?

Aufgabe 2.3 (Insertion Sort)

Implementieren und benchmarken Sie `INSERTION_SORT`. Konstruieren Sie dabei jeweils Best- und Worst-Case-Szenarien. Entsprechen die Ergebnisse Ihren Erwartungen?

Aufgabe 2.4 (Merge Sort Manuell)

Führen Sie `MERGE_SORT` auf dem Array $A = [31, 41, 59, 26, 41, 58, 1, 12, 43]$ auf Papier aus.