

# Simulation einer mündlichen Abiturprüfung – Informatik

Thema: OOP, Algorithmen, Datenbanken, Formale Sprachen

March 9, 2025

## Aufgabe 1: Analyse eines Sortieralgorithmus

Ein unbekannter Algorithmus zur Sortierung eines Arrays wird bereitgestellt. Der Name der Klasse lautet `Sortierung`.

### Gegebener Java-Code

```
public class Sortierung {
    public static void sortiere(int[] arr) {
        for (int i = 1; i < arr.length; i++) {
            int key = arr[i];
            int j = i - 1;
            while (j >= 0 && arr[j] > key) {
                arr[j + 1] = arr[j];
                j = j - 1;
            }
            arr[j + 1] = key;
        }
    }

    public static void main(String[] args) {
        int[] werte = {5, 2, 9, 1, 5, 6};
        sortiere(werte);
        for (int zahl : werte) {
            System.out.print(zahl + " ");
        }
    }
}
```

### Teilfragen zur Analyse des Codes

1. Beschreiben Sie die Funktionsweise des Algorithmus. Welche Art von Sortierung wird hier durchgeführt?

2. Welche Art der Implementierung wurde hier verwendet und warum könnte sie vorteilhaft sein?
3. Verdeutlichen Sie die Funktionsweise des Algorithmus anhand eines Beispielaufrufs mit dem Array  $\{5, 2, 9, 1, 5, 6\}$ .
4. Wie viele Vergleiche und Verschiebungen sind erforderlich, um das Array vollständig zu sortieren?
5. Bestimmen Sie die Zeitkomplexität des Algorithmus.
6. Wie verhält sich der Algorithmus im Vergleich zu anderen bekannten Sortieralgorithmen?

## Aufgabe 2: Normalisierung einer Bestelldatenbank

Ein Online-Shop speichert Bestellungen in einer relationalen Datenbank. Die ursprüngliche Tabellenstruktur ist wie folgt:

Table 1: Ursprüngliche nicht normalisierte Tabelle

BestellID	Kunde	Produkt	Kategorie	Preis	Menge
1	Max Meier	Laptop	Elektronik	1200	1
2	Lisa Becker	Kopfhörer	Elektronik	200	1
3	Max Meier	Maus	Zubehör	50	2
4	Julia Schmitt	Schreibtisch	Möbel	300	1
5	Thomas Müller	Monitor	Elektronik	250	1

### Teilaufgaben zur Normalisierung

1. Identifizieren Sie Redundanzen in der Tabelle.
2. Zerlegen Sie die Tabelle in mehrere normalisierte Tabellen bis zur 3. Normalform.
3. Welche Vorteile bringt die Normalisierung in diesem Fall?

## Aufgabe 3: Analyse einer formalen Grammatik

Eine kontextfreie Grammatik  $G$  erzeugt gültige Telefonnummern im deutschen Format:

- Terminale:  $\{0 - 9, -, \}$
- Nichtterminale:  $\{S, V, T\}$
- Startsymbol:  $S$
- Produktionsregeln:

$$S \rightarrow V - T$$

$$V \rightarrow 030 \mid 040 \mid 089 \mid 0221 \mid 069$$

$$T \rightarrow 1T \mid 2T \mid 3T \mid \dots \mid 9T \mid 1234567$$

## **Teilfragen zur Grammatik**

1. Leiten Sie die Telefonnummer 030-1234567 mit der Grammatik ab.
2. Welche Einschränkungen hat diese Grammatik? Sind alle gültigen deutschen Telefonnummern darstellbar?
3. Wie könnte die Grammatik erweitert werden, um internationale Vorwahlen zu berücksichtigen?