

## Algorithmen und Datenstrukturen

Sommersemester 2013

### 4. Übungsblatt

#### Aufgabe 1: Java - Statische Komponenten

- a) Implementieren Sie eine Klasse `Bottle`, welche eine Flasche mit einer Bezeichnung, dem Alkoholgehalt sowie die Füllmenge repräsentiert. Zudem soll diese Klasse über eine Methode `getNumberOfBottles()` verfügen, um die Anzahl der bisher erzeugten `Bottle`-Objekte abfragen zu können. Erweitern Sie diese Klasse um eine entsprechende Funktionalität bereitstellen zu können. (2 Punkte)
- b) Implementieren Sie nun zwei weitere Klassen `BeerBottle` bzw. `WineBottle`, welche von der Klasse `Bottle` erben. `BeerBottle` soll zusätzlich über einen Brauernamen bzw. `WineBottle` über das Alter des Weines verfügen. Wie die Oberklasse `Bottle` sollen diese beiden Klassen über eine Methode `getNumberOfBottles()` verfügen, welche die Anzahl der erzeugten `BeerBottle`- bzw. `WineBottle`-Objekte zurückliefert.

Testen Sie Ihre Implementierung in einer `main`-Methode indem Sie einige `BeerBottle`- und `WineBottle`-Objekte erzeugen und anschließend mit Hilfe der zuvor erstellten Methoden die Anzahl der erzeugten `Bottle`-, `BeerBottle`- und `WineBottle`-Objekte ausgeben. (4 Punkte)

#### Aufgabe 2: $O$ - und $\Omega$ -Notation

Beweisen Sie folgende Behauptungen und Folgerungen:

- a)  $4n^5 - 3n^2 - 5n + 1$  ist  $O(n^5)$  (3 Punkte)
- b)  $4n^5 - 3n^2 - 5n + 1 = n^5 + 3 * (n^5 - n^2) - 5n + 1$  ist  $\Omega(n^5)$  (3 Punkte)
- c)  $n^4 - 100 * n^3$  ist **nicht**  $O(n^2)$  (3 Punkte)

### Aufgabe 3: Laufzeitabschätzungen

In dieser Aufgabe soll die Laufzeit in  $O$ -Notation abgeschätzt werden. Leiten Sie Ihre Abschätzung her.

a) methode3A

(4 Punkte)

```
public static void methode3A (int g) {
    int i = 0;
    int k = 0;
    while (i < g) {
        int temp = i;
        for (int j = 0; j < g; j++) k = k * temp;
        i = i + 1;
    }
    return;
}
```

b) methode3B - Stellen Sie zur Abschätzung die Rekursionsgleichung der Methode auf.  
(6 Punkte)

```
// erster Aufruf mit methode3B(array, array.length-1)
public static boolean methode3B(int[] array, int n){
    if (n == 0) return true;

    for (int i = 0; i < n; i++){
        if (array[i] == array[n]) return false;
    }
    return methode3B(array, n-1);
}
```

**Beachten** Sie die Hinweise auf dem ersten Übungsblatt.

---

**Abgabetermin:** Donnerstag, den 2. Mai bis 10 Uhr im Institut für Informationssysteme, 2. OG im Informatik-Neubau (Raum 2.071, Küche neben Sekretariat)