

— Übungsblatt 1 (T) —

Aufgabe 1 (T)

(Analyse von Algorithmen)

Gegeben seien die folgenden Java-Methoden:

```
public static void meth(int n) {  
    step1();  
    for (int i=1; i <= n; i++) {  
        step1();  
        step2(n);  
        for (int j=1; j <= n; j++) {  
            step1();  
        }  
    }  
}  
  
public static void step2(int k) {  
    for (int i=1; i <= k; i++) {  
        step1();  
        step1();  
        for (int j=1; j <= 2*k; j++) {  
            step1();  
        }  
    }  
}
```

Bestimmen Sie die Ausführungszeit (den Aufwand) der Methode `meth` in Groß-O-Notation in Abhängigkeit von ihrem Parameter n unter der Annahme, dass `step1` die dominante Grundoperation der beiden Methoden ist.

Ermitteln Sie zunächst die Anzahl der durchgeführten Grundoperationen und begründen Sie dabei Ihre Rechenschritte jeweils kurz. Geben Sie dann am Ende die entsprechende Groß-O-Notation an.

Aufgabe 2 (T)

(Beschreibung von Algorithmen)

Formulieren Sie einen Algorithmus, der eine Menge von reellen Zahlen normiert, indem er zunächst die Summe aller Zahlen in der Menge berechnet und danach jede Zahl der Menge durch diese Summe dividiert und durch das dabei berechnete Ergebnis ersetzt. Realisieren Sie dies dadurch, dass Sie zunächst einen Algorithmus für die Berechnung der Summe aller Zahlen in der Menge formulieren und diesen dann im eigentlichen Algorithmus verwenden.

Geben Sie die Algorithmen in mathematischer Notation, in Pseudo-Code-Notation und in Java an.

Aufgabe 3 (T)

(Analyse von Algorithmen)

Gegeben seien die folgenden Java-Methoden:

```
public static void work(int n) {
    domStep();
    domStep();
    for (int i=1; i <= n; i++) {
        domStep();
        domStep();
        domStep();
        action(n*n);
        for (int j=1; j <= n; j++) {
            action(n);
        }
    }
}

public static void action(int k) {
    for (int i=1; i <= k; i++) {
        domStep();
        domStep();
        for (int j=1; j <= 7*k; j++) {
            domStep();
        }
    }
}
```

Bestimmen Sie die Ausführungszeit (den Aufwand) der Methode `work` in Abhängigkeit von ihrem Parameter n unter der Annahme, dass `domStep` die dominante Grundoperation der beiden Methoden ist.

Ermitteln Sie zunächst die Anzahl dieser durchgeführten Grundoperationen und begründen Sie dabei Ihre Rechenschritte jeweils kurz. Geben Sie dann am Ende die entsprechende Groß-O-Notation an.