

# Algorithmen und Programmierung

## 9. Übung

### 1. Aufgabe:

Die Invariante für den Algorithmus von GILL (siehe 8. Übung) lautet:

$$\text{ggT}(a, b) = \text{ggT}(x_l, y_l) \quad \text{und} \quad x_l \cdot v_l + y_l \cdot u_l = 2 \cdot a \cdot b$$

Weisen Sie nach, dass die Invariante gilt und dass der Algorithmus  $\text{ggT}(a, b)$  und  $\text{kgV}(a, b)$  zurückliefert! Hinweis: Es gilt  $\text{kgV}(a, b) = \frac{a \cdot b}{\text{ggT}(a, b)}$ .

### 2. Aufgabe:

- Geben Sie ein Feld der Länge 5 und ein Pivotelement an, so dass der Wert von  $i$  oder  $j$  des Algorithmus „Partition“ der Vorlesung am Ende die Feldgrenze überschreitet.
- Gegeben sei folgende leicht abgewandelte Variante des Partitionierungs-Algorithmus aus Aufgabe 2 der 8. Übung:

```
x = feld[feld.length - 1];  
i = -1;  
j = feld.length - 1;  
while ( i < j ) {  
  do {  
    j = j - 1;  
  } while ( feld[j] > x );  
  do {  
    i = i + 1;  
  } while ( feld[i] < x );  
  if ( i < j )  
    >> vertausche feld[i] und feld[j] <<  
  else  
    >> gib j aus <<  
}
```

Finden Sie heraus, wann dieser Algorithmus **nicht** funktioniert und warum!

### 3. Aufgabe:

Gegeben seien zwei Matrizen  $A$  und  $B$ . Entwickeln Sie einen Algorithmus zur Multiplikation dieser beiden Matrizen und setzen Sie diesen in Java um. Welche Voraussetzung müssen die Matrizen  $A$  und  $B$  erfüllen, um miteinander multipliziert werden zu können?

4. Aufgabe:

Schreiben Sie eine Methode `swappedCopy`, die als Ergebnis den „gespiegelten“ Inhalt eines eindimensionalen Arrays `a` vom Typ `int [ ]` liefert. Das heißt, das erste Element von `a` ist das letzte Element von `swappedCopy(a)` und so weiter. Hierbei dürfen keine Seiteneffekte auftreten, die Feldkomponenten von `a` sollen also unverändert bleiben.

Schreiben Sie eine weitere Methode `swap`, die ebenfalls über diese Funktion verfügt, aber den Rückgabebetyp `void` besitzt. Hierzu sollen bewusst Seiteneffekte eingesetzt werden; das Ergebnis soll somit am Ende der Methode in `a` selbst stehen.

5. Aufgabe:

Gegeben sei folgendes Programmstück:

```
static double max(int a, int b, double c){
    // Maximum bestimmen
}
static double max(int a, double b, double c){
    // Maximum bestimmen
}
static double max(int a, double b){
    // Maximum bestimmen
}
static double max(double a, int b){
    // Maximum bestimmen
}

public static void main(String [ ] args){
    int x=4,y=5,z=6;
    System.out.println("Maximum:_" + max(x,y,z));
    System.out.println("Maximum:_" + max(x,y));
}
```

Welche Methode `max` wird jeweils verwendet?

6. Aufgabe:

Was geschieht beim Übersetzen bzw. Ausführen der folgenden Programmstücke?

a)

```
int k;
System.out.println("Variable:_" + k);
```

b)

```
int [] f = new int [5];  
for (int i=0; i<5; i++){  
    System.out.println("Feld:_" + f[i]);  
}
```

c)

```
public static class Test{  
    public int c;  
}  
  
public static void main(String [] args){  
    Test t = new Test();  
    System.out.println("Klasse:_" + t.c);  
}
```