

# Algorithmen und Programmierung

## 1. Übung – Lösungsvorschläge

1. Aufgabe:

a)

```
import Prog1Tools.IOTools;

public class KHochK {

    public static void main(String [] args) {
        int k, zaehler, potenz;

        k = IOTools.readInteger("Geben_Sie_k_ein:_");
        zaehler = 0;
        potenz = 1;
        while (zaehler < k) {
            potenz = potenz * k;
            zaehler++;
        }
        System.out.println("k_hoch_k=_ " + potenz);
        return;
    }
}
```

b) Sei  $zaehler_l$  der Wert der Variable *zaehler* (Zähl-Variablen der Schleife) nach dem  $l$ -ten Schleifendurchlauf. Es gilt folgende Invariante für  $l \geq 1$ :

$$zaehler_l = l.$$

### Induktionsanfang

Vor dem ersten Durchlauf der Schleife ist der Wert von *zaehler* gleich Null, also

$$zaehler_0 = 0.$$

Nach dem ersten Durchlauf ist *zaehler* um Eins erhöht, also

$$zaehler_1 = 1.$$

### Induktionsschritt

Für  $1 < l \leq k$  ist laut Voraussetzung (Invariante)

$$zaehler_{l-1} = l - 1.$$

Damit ist aufgrund der Inkrementierung im Schleifenrumpf

$$zaehler_l = (l - 1) + 1 = l.$$

Somit ist für  $k = l$

$$zaehler_l = k.$$

Nach dem  $k$ -ten Schleifendurchlauf ist also die Bedingung der While-Schleife  $zaehler < k$  nicht mehr erfüllt und die Schleife wird verlassen. Da  $k$  laut Definition endlich ist (nur ein endlicher Wert kann eingelesen werden), ist folglich auch die Anzahl der Schleifendurchläufe endlich.

2. Aufgabe:

```
import Prog1Tools.IOTools;

public class Dreieck {

    public static void main(String [] args) {

        int anz = IOTools.readInteger("Anzahl_der_Zeilen:_");

        int i=1;

        while ( i<=anz) {
            int j=1;
            while ( j<=i) {
                System.out.print("*");
                j++;
            }
            System.out.println();
            i++;
        }
    }
}
```