

## Aufgabe 9.6: Iteration und Rekursion

### 9.6.1 Analyse eines rekursiven Algorithmus

Analysieren Sie das folgende Programm. Was wird hier berechnet? Ist Ihnen ein alternativer (nicht rekursiver) Lösungsweg bekannt?

```
// Datei: Rekursion.java
public class Rekursion
{
    public int rekursAufruf (int n)
    {
        if (n > 1)
            return n + rekursAufruf (n - 1);
        return 1;
    }

    public static void main (String[] args)
    {
        Rekursion rek = new Rekursion();
        System.out.println (rek.rekursAufruf (50));
    }
}
```

### 9.6.2 Analyse eines iterativen Algorithmus

Welche mathematische Formel berechnet das Programm? Wie lautet das Ergebnis?

```
// Datei: Iteration.java

public class Iteration
{
    public int iterativAufruf (int n)
    {
        int wert = 1;

        for (int i = 2; i <= n; i++)
        {
            wert *= i;
        }
        return wert;
    }

    public static void main (String[] args)
    {
        Iteration it = new Iteration();
        System.out.println (it.iterativAufruf (4));
    }
}
```

### 9.6.3 Berechnung der Potenz $a^n$

- a) Berechnen Sie mit Hilfe einer Rekursion die Potenz  $a^n$ . Die Variablen  $a$  und  $n$  sollen sich auf Werte größer 0 beschränken und von der Tastatur eingelesen werden. Vervollständigen Sie hierzu im folgenden Programm die mit . . . . gekennzeichneten Stellen:

```
// Datei: PotenzRekursiv.java

public class PotenzRekursiv
{
    public static . . . . . berechnePotenz (. . . . .)
    {
        . . . . .
    }

    public static void main (String[] args)
    {
        java.util.Scanner scanner =
            new java.util.Scanner (System.in);
        try
        {
            System.out.println
                ("Gib einen Wert >0 für a ein: ");
            int a = scanner.nextInt();

            System.out.println
                ("Gib einen Wert >0 für n ein: ");
            int n = scanner.nextInt();

            int ergebnis = berechnePotenz (a, n);
            System.out.println ("Das Ergebnis ist: " +
                               ergebnis);
        }
        catch (Exception ex)
        {
            System.out.println (ex.toString());
        }
    }
}
```

- b) Erweitern Sie die Methode `berechnePotenz()`, sodass sie auch mit den Werten  $a = 0$  und  $n = 0$  zurechtkommt. Negative Werte werden weiterhin nicht betrachtet. Beachten Sie, dass eine Potenz für den Exponenten 0 immer 1 liefert. Ausnahme: Falls  $a$  den Wert 0 hat, ist das Ergebnis immer 0.
- c) Schreiben Sie eine neue Klasse `PotenzIterativ`. Diese Klasse implementiert die Lösung aus Aufgabe 9.6.3 b) mit Hilfe einer Iteration. Verwenden Sie dazu eine `for`-Schleife.