

## Übung 7

4. Dezember 2018

---

### Aufgabe 7.1: *Quellenverfahren*

In der Vorlesung wurde das Quellenverfahren besprochen, es dient der Konversion einer Dezimalzahl in eine Darstellung im Stellenwertsystem mit einer beliebigen Basis. Hierbei wird die Dezimalzahl wiederholt durch die neue Basis  $b$  geteilt und jeweils der ganzzahlige Divisionsanteil und der Rest ermittelt. Schreiben Sie ein Programm, welches das Quellenverfahren mit Hilfe einer while Schleife umsetzt. Konvertieren Sie mit Ihrem Programm die Zahl  $3425_{10}$  in das Dualsystem.

### Aufgabe 7.2: *Berechnung der Fakultät*

In der Mathematik gilt für die Berechnung der Fakultät:

$$n! = 1 \cdot 2 \cdot \dots \cdot n \quad (7.2.1)$$

Belegen Sie in Ihrem Programm  $n$  mit einem Wert und berechnen Sie die Fakultät.

### Aufgabe 7.3: *Verwendung von Switch*

Geben Sie einen Monat als Zahl über die Konsole ein (1 = Januar, 12 = Dezember). Ermitteln Sie mit Hilfe einer Switch-Anweisung, zu welchem Quartal der Monat gehört. Benutzen Sie hierbei höchstens 4 Case-Anweisungen.

### Aufgabe 7.4: *Arrays*

Erzeugen Sie ein Array der Länge 50 und initialisieren Sie das Array mit den Werten  $a[i] = i^4$ . (Siehe auch API-Dokumentation zu `java.lang.Math.pow(...)` um die Potenz zu berechnen)

1. Geben Sie das Array aus.
2. Geben Sie das Array rückwärts aus.
3. Vertauschen Sie das letzte Element mit dem ersten Element und geben Sie das Array aus.

### Aufgabe 7.5: *Arrays*

Erzeugen Sie ein zweidimensionales Array mit 5 Zeilen und 4 Spalten und initialisieren Sie dieses Array mit den Werten 1 bis 20 und geben es aus.

### Aufgabe 7.6: *Maximum sum*

By starting at the top of the triangle below and moving to adjacent numbers on the row below, the maximum total from top to bottom is 23.

```
    3
   7 4
  2 4 6
 8 5 9 3
```

That is,  $3 + 7 + 4 + 9 = 23$ .

Find the maximum total from top to bottom of the triangle below:

```
      75
     95 64
    17 47 82
   18 35 87 10
  20 04 82 47 65
 19 01 23 75 03 34
 88 02 77 73 07 63 67
99 65 04 28 06 16 70 92
```

41 41 26 56 83 40 80 70 33  
41 48 72 33 47 32 37 16 94 29  
53 71 44 65 25 43 91 52 97 51 14  
70 11 33 28 77 73 17 78 39 68 17 57  
91 71 52 38 17 14 91 43 58 50 27 29 48  
63 66 04 68 89 53 67 30 73 16 69 87 40 31  
04 62 98 27 23 09 70 98 73 93 38 53 60 04 23

*hint:* As there are only 16384 routes, it is possible to solve this problem by trying every route. Taken from <https://projecteuler.net/problem=18>