



Prof. Dr. Bernhard Seeger
Dipl.-Inf. Marc Seidemann
Johannes Dröner, M.Sc.

Übungen zur Vorlesung
Praktische Informatik I

Abgabe: Montag, 04.11.2013,
bis **spätestens** 10:00 Uhr
über die ILIAS-Plattform

Blatt 2

Hinweis: Ab diesem Übungsblatt soll nur eine Abgabe pro Gruppe eingereicht werden!

Auf diesem Übungsblatt sollen Sie sich unter anderem mit Kalenderberechnungen beschäftigen. Laden Sie dazu zunächst die Datei **Kalender.java** aus ILIAS herunter.

Aufgabe 2.1: Wochentagsberechnung

(3+1=4P)

- a) In dieser Aufgabe sollen Sie eine Methode `static int ermittleWochentag(int tag, int monat, int jahr)` implementieren, die zu einem gegebenen Datum den Wochentag ermittelt. Der Wochentag kann mit Hilfe der folgenden Formel ermittelt werden, wobei t für Tag, m für Monat und j für Jahr steht:

$$\left(t + 2 * m + \frac{3 * m + 3}{5} + j + \frac{j}{4} - \left(\frac{j}{100} - \frac{j}{400} - 2 \right) \right) \% 7$$

(Der Operator „%“ bestimmt den Rest, der bei einer Ganzzahldivision übrig bleibt, z. B. $22 \% 5 = 2$)

Das Ergebnis ist eine Zahl zwischen 0 und 6, wobei 0 Samstag entspricht, 1 Sonntag, 2 Montag, usw.

- b) Testen Sie Ihre Methode mit folgenden Daten, indem Sie `ermittleWochentag` aus der `main`-Methode heraus aufrufen:
- 24.12. 2013
 - 01. Mai 2014

Dokumentieren Sie Ihren Programmcode (siehe Skript Folie 109).

Aufgabe 2.2: Anzahl der Tage im Jahr

(2P)

Implementieren Sie eine Methode `static int tageImJahr(int jahr)`, in der Sie die Anzahl der Tage im Jahr unter Berücksichtigung von Schaltjahren ermitteln (siehe Skript Folie 122). Definieren Sie dazu in der Methode eine Variable `int tage`.

Dokumentieren Sie Ihren Programmcode (siehe Skript Folie 109).

Aufgabe 2.3: Monatstage

(5P)

In dieser Aufgabe sollen Sie die Anzahl der Tage in einem bestimmten Monat ermitteln. Implementieren Sie dazu die Methode `static int tageImMonat(int monat, int jahr)`, die für den angegebenen Monat (1-12) die Anzahl der Wochentage zurückgibt. Um festzustellen, ob es sich um ein Schaltjahr handelt, können Sie die Methode aus Aufgabe 2.2 zur Ermittlung der Anzahl der Tage im Jahr verwenden.

Dokumentieren Sie Ihren Programmcode (siehe Skript Folie 109).

Aufgabe 2.4: Tag im aktuellen Jahr

(8P)

Sie sollen nun eine Methode `static int tagDesJahres(int tag, int monat, int jahr)` implementieren, die zu einem Datum berechnet, welchen Tag es im jeweiligen Jahr darstellt (z.B. ist der 02.02. der 33 Tag im Jahr). Verwenden Sie hierfür die Methode aus Aufgabe 2.3.

Für diese Aufgabe sollen Sie die in der Vorlesung vorgestellte WHILE-Schleife (siehe Skript Folie 123) benutzen, um über die Monate des Jahres zu iterieren.

Dokumentieren Sie Ihren Programmcode (siehe Skript Folie 109).

Aufgabe 2.5: Tage bis Weihnachten

(12P)

Implementieren Sie eine Methode `static int tageBisWeihnachten(int tag, int monat, int jahr)`, die zu einem gegebenen Datum die Anzahl der Tage bis zum nächsten Weihnachtsfest (24.12.) ermittelt. Sie können dafür die Methode aus Aufgabe 2.4 verwenden.

Beachten Sie dabei auch den speziellen Fall, dass das gegebene Datum zwischen dem 24.12. und Neujahr liegt. In diesem Fall muss die Zeit bis zum Weihnachtsfest im nächsten Jahr ermittelt werden!

Dokumentieren Sie Ihren Programmcode (siehe Skript Folie 109).

Aufgabe 2.6: Pi

(6P)

In dieser Aufgabe sollen Sie Pi mit Hilfe einer WHILE-Schleife annähern. Dazu verwenden wir folgende Berechnungsvorschrift:

$$\frac{\pi^2}{6} = \frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \frac{1}{5^2} + \dots$$

Erstellen Sie eine neue Klasse *Funktionen* und implementieren Sie eine Methode `static double pi()`. Brechen Sie die Berechnung nach 1000 Schritten ab. Die Wurzel einer Zahl `zahl` können Sie mit folgendem Befehl ziehen: `Math.sqrt(zahl)`.

Hinweis: Um Rundungsfehler zu vermeiden, verwenden Sie bitte in dieser Methode für alle benötigten Variablen den Typ `double`.

Dokumentieren Sie Ihren Programmcode (siehe Skript Folie 109).