

# Programmieren in Java

## Einige Übungen zum Kapitel Klassen und Objekte

### Aufgabe 1 (40 Punkte)

Die Objekte der öffentlichen Klasse `Time24` dienen zur Darstellung von Uhrzeiten im 24-Stunden-Format.

Die Klasse besitzt die privaten Attribute

- `hour` und
- `minute`,

die beide vom Typ `int` sind.

Die Klasse hat drei öffentliche Konstruktoren

- den allgemeinen Konstruktor, der zwei Argumente vom Typ `int` hat, mit deren Werten die Attribute initialisiert werden,
- den Standardkonstruktor, der die beiden Attribute mit dem Wert 0 initialisiert,
- und einen so genannten Kopierkonstruktor, der als Parameter ein Objekt vom Typ `Time24` hat. Die Attribute des neu erzeugten Objekts werden mit den Attributwerten des Parameter-Objekts initialisiert.

Die Klasse hat die folgenden öffentlichen Methoden:

- Die Methode `readTime()` liest die Attributwerte eines `Time24`-Objekts ein. Sie verwendet dazu die Klassenmethode `readInt()` der Klasse `TastaturEingabe`.
- Die Methode `writeTime()` gibt ein `Time24`-Objekt in der Form `<Stunde> : <Minute>` aus, also z.B. `12:15`.
- Die Methode `addTime()` hat ein Argument vom Typ `int`, welches immer  $\geq 0$  sei und ein Zeitintervall, angegeben in Minuten, darstellt. Dieses Zeitintervall wird zur aktuellen Uhrzeit hinzu addiert.
- Zum lesenden bzw. schreibenden Zugriff auf die Attributwerte dienen die Methoden `getHour()` und `setHour()` sowie `getMinute()` und `setMinute()`.
- Die parameterlose Methode `toString()` erzeugt aus dem `Time24`-Objekt eine Zeichenkette der Form `"<Stunde>:<Minute>"`, also z.B. `"12:15"`, und liefert diese Zeichenkette als Rückgabewert.

Die Konstruktoren und alle Methoden, welche die Attributwerte verändern, also insbesondere auch `readTime()` und `addTime()`, müssen sicherstellen, dass die Werte der Attribute `hour` und `minute` in den Bereichen 0 bis 23 bzw. 0 bis 59 liegen. Dazu ist es eventuell nötig, die Attributwerte entsprechend anzupassen; dieser Prozess wird als Normalisierung bezeichnet und hier von

- der privaten parameterlosen Methode `normalizeTime()` übernommen.

Die folgende Tabelle enthält Beispiele für Normalisierungen:

Original-Uhrzeit	normalisierte Uhrzeit
14:80	15:20
27:15	3:15 (am nächsten Tag)
22:150	0:30 (150 Minuten = 2 Stunden und 30 Minuten)

Bei der Implementierung der Methode `normalizeTime()` sind die Operatoren `/` (ganzzahlige Division) und `%` (Rest bei der ganzzahligen Division) nützlich.

- a) Implementieren (d.h., schreiben) Sie die Klasse `Time24` in Java. Gehen Sie der Einfachheit halber davon aus, dass die Attribute `hour` und `minute` keine negativen Werte annehmen.

- b) Schreiben Sie eine Methode `main()` für die Klasse `Time24`.

Legen Sie zunächst drei Referenzen `sanfrancisco`, `frankfurt` und `melbourne` auf `Time24`-Objekte an.

Erzeugen Sie ein `Time24`-Objekt für San Francisco und lesen Sie die aktuelle Uhrzeit in San Francisco ein.

Erzeugen Sie `Time24`-Objekte für die Uhrzeit in Frankfurt (+ 9 Stunden) und Melbourne (+ 17 Stunden).

Geben Sie die drei Uhrzeiten aus.