

Übungen zu Einführung in die Informatik: Programmierung und Software-Entwicklung

Aufgabe 5-1 Wiederholungsanweisungen in Java (while-Schleife) *Präsenz*

Ein Autokonzern investiert für die Entwicklung eines neuen Fahrzeug-Modells einen bestimmten Investitionsbetrag. Der Konzern möchte kalkulieren, nach wievielen Jahren sich der Investitionsbetrag amortisiert hat, d.h. nach wievielen Jahren der Gewinn aus den Einnahmen durch den Autoverkauf diesen Investitionsbetrag deckt. Dabei wird davon ausgegangen, dass ein Auto 22500 Euro in der Herstellung kostet und für 25000 Euro verkauft wird. Der Konzern schätzt dazu, wieviele Autos im ersten Jahr verkauft werden. Er geht davon aus, dass jedes Jahr 5% mehr Autos als im Vorjahr verkauft werden.

Schreiben Sie in einer Klasse `Investitionsrechner` ein Java-Programm, welches pro Jahr ausgibt, das wievielte Jahr gerade berechnet wird, wieviele Autos in diesem Jahr schätzungsweise verkauft werden, wie hoch die Herstellungskosten für alle verkauften Autos in diesem Jahr sind, mit welchem Umsatz in diesem Jahr gerechnet werden kann, wie hoch der Reingewinn (also Umsatz - Herstellungskosten) in diesem Jahr ist und wie hoch der aktuelle Gesamtgewinn über alle Jahre gerechnet ist. Das Java-Programm soll diese Berechnung stoppen, sobald der Gesamtgewinn über alle Jahre den Investitionsbetrag erreicht oder überschreitet.

Bestimmen Sie zunächst, welche lokalen Variablen Sie benötigen (mit Typ). Testen Sie Ihr Programm für 1) einen Investitionsbetrag von einer Million Euro und einer geschätzten Anzahl von 50 im ersten Jahr verkauften Autos und 2) für einen Investitionsbetrag von 25 Millionen Euro und einer geschätzten Anzahl von 1000 im ersten Jahr verkauften Autos.

Aufgabe 5-2 Wiederholungsanweisungen in Java (while-Schleife) *Hausaufgabe*

Sie möchten ein Immobilien-Darlehen über eine unbegrenzte Laufzeit aufnehmen und berechnen, nach wievielen Jahren Sie das Darlehen komplett zurückgezahlt haben. Mit der Bank haben Sie einen jährlichen Zinssatz sowie einen Tilgungssatz vereinbart. Pro Jahr wird eine gleichbleibende Jahresrate an die Bank gezahlt in Höhe von $\text{Darlehensbetrag} * (\text{Zinssatz} + \text{Tilgungssatz})/100.0$.

Am Ende jedes Jahres sind Jahreszinsen entsprechend des vereinbarten Zinssatzes angewandt auf die Restschuld am Ende des vorherigen Jahres zu zahlen. Die Jahrestilgung am Ende jedes Jahres ergibt sich dann durch die Differenz aus Jahresrate und Jahreszinsen. Die Restschuld am Ende jedes Jahres verringert sich entsprechend der Jahrestilgung.

Schreiben Sie in einer Klasse `Tilgungsrechner` ein Java-Programm, das für einen gegebenen Darlehensbetrag, Zinssatz und Tilgungssatz eine Ausgabeliste in der unten angegebenen Form ausgibt, in der, bis das Darlehen vollständig zurückgezahlt ist, pro Jahr die Jahreszinsen, die Jahrestilgung und die Restschuld am Ende des Jahres ausgewiesen sind.

Ausgabeformat:

```
Jahr 1:  
  Jahreszinsen: 200.0  
  Jahrestilgung: 1000.0  
  Restschuld: 9000.0  
Jahr 2:  
  Jahreszinsen: 180.0  
  Jahrestilgung: 1020.0
```

```

    Restschuld: 7980.0
Jahr 3:
    Jahreszinsen: 159.6
    Jahrestilgung: 1040.4
    Restschuld: 6939.6
Jahr 4:
    Jahreszinsen: 138.792
    Jahrestilgung: 1061.208
    Restschuld: 5878.392
...

```

Testen Sie Ihr Programm für 1) einen Darlehensbetrag von 10000 Euro, einen Zinssatz von 2.0% sowie einen Tilgungssatz von 10.0% und 2) einen Darlehensbetrag von 300000 Euro, einen Zinssatz von 3.5% sowie einen Tilgungssatz von 2.0%. Im Fall 1) soll die oben angegebene Ausgabe erfolgen.

Aufgabe 5-3 **Wiederholungsanweisungen in Java (for-Schleife)** *Präsenz*

Sie möchten einen Bausparvertrag aufnehmen und berechnen, wieviel Bausparvermögen Sie nach einer vorgegebenen Zeit angespart haben. Die Verzinsung ist fix auf jährlich 4.5% festgelegt und wird jeweils am Ende des Jahres aufgeschlagen. Verträge sind nur mit einer jährlichen Laufzeit möglich. Die Einzahlung in den Bausparvertrag erfolgt monatlich mit einem festen Betrag.

Schreiben Sie in einer Klasse `Bausparkonto` ein Java-Programm, das für eine Laufzeit von 6 Jahren (72 Monaten) das am Ende des Zeitraums angesparte Vermögen berechnet, wenn monatlich 200 Euro eingezahlt werden. Verwenden Sie dazu als Iterationsanweisung eine `for`-Schleife. Bestimmen Sie zunächst, welche lokalen Variablen Sie benötigen (mit Typ).

Aufgabe 5-4 **Wiederholungsanweisungen in Java (for-Schleife)** *Hausaufgabe*

Bei Ihrem Supermarkt um die Ecke gibt es ein neues Bonuspunkte-System. Für jeden Tag, an dem Sie einkaufen gehen, erhalten Sie einen Bonuspunkt (unabhängig vom Betrag des Einkaufs). Gehen Sie aber an mehreren Tagen in Folge einkaufen, bekommen Sie pro Tag mehr Bonuspunkte, wie folgt: Am ersten Tag erhalten Sie einen Bonuspunkt. Am 2. und 3. Tag erhalten Sie je zwei Bonuspunkte; am 4. bis 6. Tag erhalten Sie je drei Bonuspunkt; am 7. bis 10. Tag erhalten Sie je vier Bonuspunkte usw. Das heißt, Sie erhalten einen Tag lang einen Bonuspunkt, zwei Tage lang zwei Bonuspunkte, drei Tage lang drei Bonuspunkte usw.

Schreiben Sie in einer Klasse `Bonuspunkte` ein Java-Programm, das berechnet, wie viele Tage hintereinander Sie einkaufen gehen müssen, um 100 Punkte **pro Tag** zu erhalten. Das Programm soll für jede Anzahl an Bonuspunkten ≤ 100 ausgeben, ab welchem Tag Sie diese Anzahl erhalten. Folgende Ausgabe sollte also produziert werden:

```

Ab Tag 1 erhält man pro Einkauf: 1 Bonuspunkte
Ab Tag 2 erhält man pro Einkauf: 2 Bonuspunkte
Ab Tag 4 erhält man pro Einkauf: 3 Bonuspunkte
Ab Tag 7 erhält man pro Einkauf: 4 Bonuspunkte
Ab Tag 11 erhält man pro Einkauf: 5 Bonuspunkte
Ab Tag 16 erhält man pro Einkauf: 6 Bonuspunkte
Ab Tag 22 erhält man pro Einkauf: 7 Bonuspunkte
...
Ab Tag 4852 erhält man pro Einkauf: 99 Bonuspunkte
Ab Tag 4951 erhält man pro Einkauf: 100 Bonuspunkte

```

Verwenden Sie dazu als Iterationsanweisung eine `for`-Schleife. Bestimmen Sie zunächst, welche lokalen Variablen Sie benötigen (mit Typ).

Besprechung der Präsenzaufgaben in den Übungen ab 06.11.2014. Abgabe der Hausaufgaben bis Mittwoch, 19.11.2014, 14:00 Uhr über UniworX (siehe Folien der ersten Zentralübung).

- *Erstellen Sie zu jeder Aufgabe eine Klasse, die den Namen trägt, der in der Aufgabe gefordert ist.*
- *Geben Sie nur die entsprechenden .java-Dateien ab. Wir benötigen **nicht** Ihre .class-Dateien.*