

Name: _____ Tutor: _____
 Matrikelnummer: _____ Punkte: _____
 Gruppe: _____ Abzugeben bis: 6.12.2000, 12:00
 Übungsleiter: _____ Bearbeitungsdauer: _____

Aufgabe 1 (5 + 2 + 3 Punkte): Modellierung

Suchen Sie sich aus der realen Welt konkrete, miteinander in Beziehung stehende Dinge (Objekte), die Sie als Klassen modellieren.

Beispiel:

```
public class Fahrrad {
    Rad raeder[]; // Beziehung mit Raedern
    double rahmenGewicht;
    int produktionsJahr;

    double gesamtGewicht() {
        return rahmenGewicht + raeder[0].gewicht() +
            raeder[1].gewicht();
    }
    ...
}

public class Rad {
    ...
}
```

- a) Implementieren Sie die Klassen in Java. Gefordert sind mind. 3 Klassen, mit mind. je 3 unterschiedlichen Attributen und 3 (nichttrivialen) Methoden. Die Klassen müssen sich referenzieren (dem entsprechen die oben genannten Beziehungen zwischen den Objekten).
- b) Zeichnen Sie für die Klassen unter Punkt a) eine konkrete Objektstruktur mit den Beziehungen zueinander (so wie in der Vorlesung vorgestellt). Jede Klasse muss durch mindestens ein Objekt vertreten sein.
- c) Schreiben Sie ein Testprogramm, das die unter Punkt b) festgelegte Objektstruktur anlegt (inkl. Referenzen zwischen den Objekten) und alle Variablen initialisiert. Anschliessend sollen die Methoden getestet werden.

Aufgabe 2 (2 + 1 Punkte): Klasse String und JDK-Dokumentation

Suchen Sie in der JDK-Dokumentation nach der Klasse String (Sie benötigen dazu einen HTML-fähigen Browser). Tip: Man findet die Klasse String im Package "java.lang".

- a) Sehen Sie sich die Beschreibung der Klasse String an, suchen Sie sich 4 Methoden aus und Testen Sie sie mit je mindestens 4 *interessanten* Testfällen.
- b) Es gibt auch noch die Klasse StringBuffer, die man ähnlich wie die Klasse String verwenden kann. Es gibt jedoch einen Grund warum man in gewissen Fällen besser StringBuffer verwenden sollte. Finden Sie ihn heraus (Steht alles in der Doku).

Aufgabe 3 (1 + 2 + 3 + 5 Punkte): Lineare Liste

Es sollen Spenden verwaltet werden. Eine Spende enthält Daten über Betrag (max. 100000000.-), eine Kennzeichnung für die Währung (ATS oder EURO), Name des Spenders, Spendennummer (zur eindeutigen Identifikation der Spende) und ob der Betrag schon eingezahlt wurde oder erst zugesagt.

a) Schreiben Sie eine Klasse namens `Contribution` (=Spende), deren Instanzen je einer solchen Spende entsprechen. Überlegen Sie sich die Datentypen der Instanzvariablen.

b) Schreiben Sie Methoden namens "get..." (Lese Wert ...) und "set..." (Setze Wert ...) für alle verwendeten Instanzvariablen. Überprüfen Sie bei den "set..."-Methoden auf mögliche ungültige Werte (z.B. negative Spendenbeträge, etc.) und geben Sie in solchen Fällen einen entsprechenden Fehler aus (`writeln (...), ...`).

c) Machen Sie aus den Spenden eine lineare Liste. Schreiben Sie dazu eine Klasse `ContributionList`, die eine Variable namens *head* (Anfang der Liste) enthält. Fügen Sie der Klasse `ContributionList` ausserdem eine Variable namens *next* (Verweis auf das nächste Element in der Liste) hinzu (Welchen Typ müssen die Variablen *head* und *list* haben?). Implementieren Sie nun eine Methode "*registerContribution (Contribution item)*" in der Klasse `ContributionList`, die eine vorher erzeugte und initialisierte Spende der linearen Liste (vorne) anfügt.

d) Bei den folgenden beiden Methoden (implementiert in `Contribution`) soll der Betrag umgerechnet und die Kennzeichnung für die Währung korrigiert werden. Ist die Währung schon umgerechnet, so soll nichts gemacht werden.

`void convertFromATSToEURO () // Rechnet eine Spende von ATS in EURO um`

`void convertFromEUROToATS () // Rechnet eine Spende von EURO in ATS um`

Implementieren Sie folgende Methoden in der Klasse `ContributionList`:

`void listAll () // Schreibt alle vorhandenen Spenden auf den Bildschirm`

`double sum () // Summiert alle vorhandenen Spenden`

`double max () // Berechnet den maximalen Betrag aller Spenden`

`void listGreater (double value) // Schreibt alle Spenden mit einem Betrag größer als value auf den Bildschirm`

`void convertAllFromATSToEURO () // Aendere die Betraege aller Spenden auf EURO (die Methode convertFromATSToEURO () muss dazu verwendet werden)`

`void convertAllFromEUROToATS () // Aendere die Betraege aller Spenden auf ATS (die Methode convertFromEUROToATS () muss dazu verwendet werden)`

Verwenden Sie zum Testen eine eigene Klasse namens `TestContribution`.