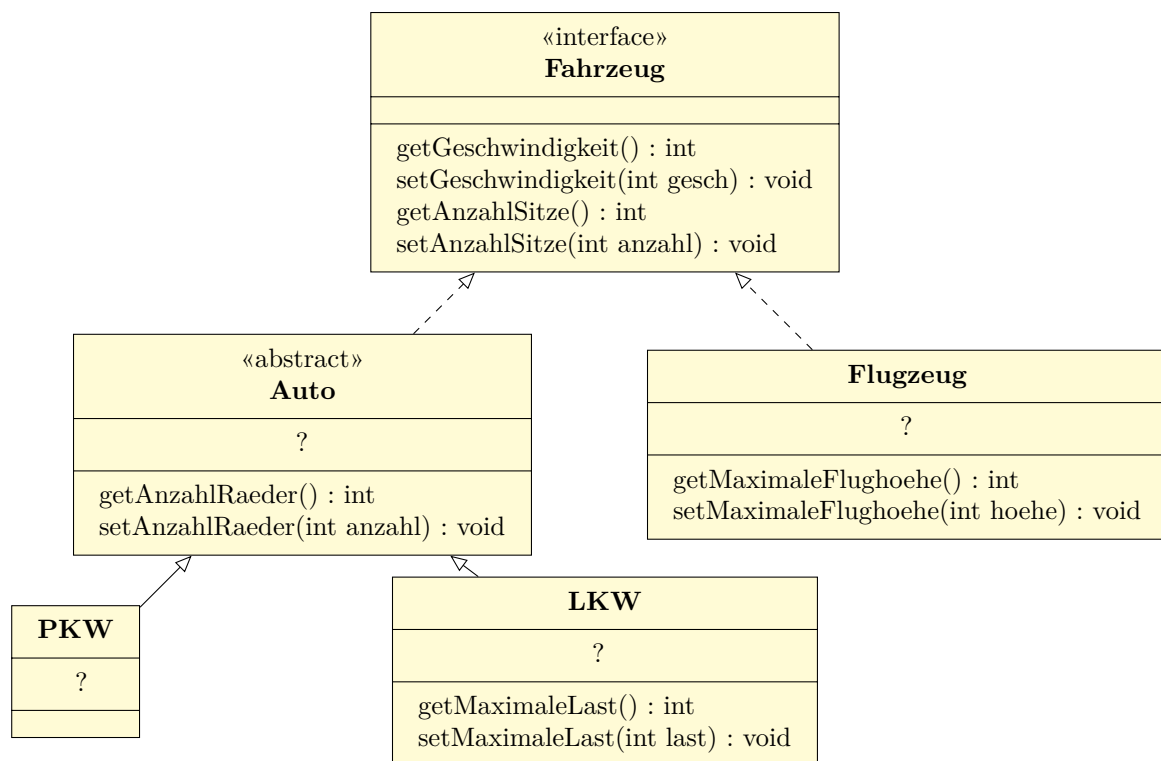


7. Übung zu Informatik I im WS 16/17
Abgabe: 5. Dezember 2016, 12⁰⁰ Uhr.

Aufgabe 17 - Gruppenabgabe

(10 Punkte)

Implementieren Sie drei Klassen **Flugzeug**, **PKW** und **LKW** sowie das Interface **Fahrzeug** und die abstrakte Klasse **Auto**. Setzen Sie dabei Vererbung anhand des folgenden UML Klassendiagramms ein:



Implementieren Sie alle Getter und Setter entsprechend dem UML Klassendiagramm! Achten Sie darauf, dass Methoden von Oberklassen nur implementiert werden sollen, wenn es notwendig ist. Fügen Sie die entsprechenden Attribute an geeigneter Stelle zu den Klassen hinzu! Wählen Sie die Sichtbarkeit der Attribute so, dass Sie in den Unterklassen verwendet werden können. Erstellen Sie Konstruktoren für alle Klassen. Jeder

Konstruktor soll Parameter für alle Attribute der Klasse und deren Oberklasse definieren. Zur Initialisierung der Attribute sollen der Konstruktor der jeweiligen Oberklasse, falls möglich, mitverwendet werden.

Aufgabe 18 - Gruppenabgabe

(8+2=10 Punkte)

- a) Entwickeln Sie eine Klasse `Stapel`! Diese soll zur Speicherung eines beliebigen aber festen Elementtyps (kein Basistyp) verwendet werden können. Benutzen Sie dazu das Konzept der generischen Typen. Ein Stapel stellt drei Methoden zur Verfügung:

legeDrauf	Legt ein Element, welches als Parameter übergeben wird, auf den Stapel.
nehmeHerunter	Nimmt das letzte auf den Stapel gelegte Element, löscht es vom Stapel und gibt dieses zurück.
istLeer	Gibt einen boolschen Wert zurück, der angibt, ob es Elemente auf dem Stapel gibt.

Falls die Methode `nehmeHerunter` aufgerufen wird, obwohl kein Element auf dem Stapel liegt, soll eine `StapelLeerException` geworfen werden! Diese Exception muss zunächst von Ihnen implementiert werden! Die Klasse `StapelLeerException` soll von `Exception` erben!

- b) Geben Sie an, ob folgendes Programmstück in Java funktioniert:

```
Stapel<Integer> stapel = new Stapel<>();
stapel.legeDrauf(3);
stapel.legeDrauf(new Integer(5));
int element = stapel.nehmeHerunter();
boolean istLeer = stapel.istLeer();
```

Erläutern Sie, aus welchem Grund dies der Fall ist und welche Eigenschaften von Java dafür verantwortlich sind!

Hinweis: Gehen Sie dabei davon aus, dass vom Programmstück geworfene Exceptions an höherer Stelle aufgefangen werden und daher bei der Frage nach der Funktionsfähigkeit dieses Programmstücks nicht beachtet werden müssen.

Aufgabe 19 - Gruppenaufgabe

(5+5=10 Punkte)

- a) Implementieren Sie zur Repräsentation von Euro-Münzen einen Aufzählungstyp (Enum) `Muenze`! Der Aufzählungstyp enthält für jede Euro-Münze genau ein Element:

```
EinCent, ZweiCent, FuenfCent, ZehnCent
ZwanzigCent, FuenfzigCent, EinEuro, ZweiEuro
```

Implementieren Sie außerdem eine Methode `int getValue()`, die den Wert einer

Euro-Münze (in Cent) liefert!

b) Implementieren Sie folgende statische Methode:

```
public static LinkedList<Muenze> getMuenzen(int wertInCent)
```

Diese liefert eine Liste von Euro-Münzen, sodass der Wert aller Münzen zusammen dem übergebenen nicht negativen Betrag (in Cent) entspricht! Beachten Sie dabei, dass die Liste nur eine minimale Anzahl an Münzen enthalten darf! Für negative Werte soll eine `IllegalArgumentException` geworfen werden!

Hinweis: `LinkedList` und `IllegalArgumentException` sind von Java zur Verfügung gestellte Klassen. Um diese in Ihrer Klasse zu verwenden, müssen Sie diese noch vor der Definition Ihrer Klasse importieren. Dies geschieht z.B. über folgende Import-Anweisungen:

```
import java.util.LinkedList;
import java.lang.IllegalArgumentException;
```

Hinweis: Das Hinzufügen von Elementen in eine `LinkedList<Muenze>` geschieht mit der Methode `add(Muenze m)`.

Anmerkung: *Versehen Sie ihre Abgaben bitte mit **Namen, Matrikelnummern, E-Mail-Adressen** und **Studiengängen** der beteiligten Bearbeiter/innen und laden Sie diese in der entsprechenden Aktivität im Learnweb hoch. Aufgaben, die mit dem Hinweis **Gruppenabgabe** versehen sind, dürfen mit maximal 3 Bearbeiter/innen gelöst werden. Aufgaben mit dem Hinweis **Einzelabgabe** müssen von jedem Studenten und jeder Studentin eigenständig gelöst und abgegeben werden. Viele der Aufgaben werden über das **EASy** (E-Assessment System) System im Learnweb eingereicht und automatisch vorausgewertet. Um Probleme bei der Abgabe wegen Überlastung des EASy-Servers zu vermeiden, würde ich Sie bitten, eine Abgabe auf den letzten Drücker nach Möglichkeit zu vermeiden. Bei Fragen zum Übungsbetrieb wenden Sie sich bitte an ihren/ihre Tutor/in oder an Tobias Reischmann.*