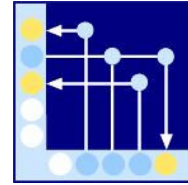




Hochschule Aalen

*Fakultät Elektronik und Informatik
Studiengang Informatik*



Programmieren 2

Vorlesung im Wintersemester 2014/2015

Prof. Dr. habil. Christian Heinlein

9. Übungsblatt (23. Dezember 2014)

Aufgabe 9: Schnittstellen

Teilaufgabe 9.a)

Definieren Sie `Account`, `ChargedAccount` und `LimitedAccount` mit den aus der Vorlesung bzw. früheren Aufgaben bekannten Eigenschaften als Schnittstellen! Fügen Sie dann `ChargedLimitedAccount` als weitere Schnittstelle zur Beschreibung von Konten, die sowohl gebührenpflichtig als auch limitiert sind, hinzu!

Teilaufgabe 9.b)

Implementieren Sie eine abstrakte Klasse `AbstractAccount` ohne irgendwelche Objektvariablen, die die Methoden `deposit`, `withdraw` und `transfer` der Schnittstelle `Account` wie folgt implementiert:

- `deposit` und `withdraw` übergeben den einzuzahlenden bzw. abzuhebenden Betrag entweder unverändert oder negiert an eine zusätzliche abstrakte Methode `booking`, die den jeweiligen Betrag zum aktuellen Kontostand addieren soll. (Da `AbstractAccount` aber keine Objektvariablen besitzt, kann diese Methode noch nicht implementiert werden.)
- `transfer` ruft in geeigneter Weise `deposit` und `withdraw` auf den beteiligten Konten auf.

Teilaufgabe 9.c)

Implementieren Sie jede der vier Schnittstellen aus Teilaufgabe a durch eine konkrete Klasse mit geeigneten Objektvariablen, Konstruktoren und Methodenimplementierungen!

Wählen Sie die Oberklasse jeder Klasse so, dass der Implementierungsaufwand möglichst gering ist!

Hinweis: Wenn die Hilfsmethode `check` in der Implementierungsklasse von `LimitedAccount` als öffentliche Klassenmethode mit geeigneten Parametern definiert wird, kann sie auch in Klassen verwendet werden, die nicht von dieser Klasse erben.

Teilaufgabe 9.d)

Zeichnen Sie einen Aufrufbaum für eine Überweisung zwischen zwei `ChargedLimitedAccount`-Objekten!