

# Praktische Informatik I

## Programmieren in Java

### Übung 6

#### Klassen, Vererbung

#### Aufgabe 1

Die Potenz  $(a + b)^n$  soll für ganzzahliges  $n \geq 0$  rekursiv berechnet werden. Schreiben Sie dazu eine Klassenmethode

```
static double reku(double a, double b, int n)
```

#### Aufgabe 2

Eine Firma beschäftigt Mitarbeiter, die entweder Angestellte oder Arbeiter sind.

Die konkrete Klasse `Mitarbeiter` verwaltet Informationen, die unabhängig vom Status der Mitarbeiter (Arbeiter oder Angestellter) für alle Mitarbeiter gespeichert werden müssen. Dies sind der Name und die Rentenversicherungsnummer, die jeweils als Zeichenkette gespeichert werden. Zur Ausgabe dieser Informationen dient die (konkrete) Methode

```
void mitarbeiterInfo()
```

Informationen, die darüber hinaus für die beiden Arten von Mitarbeitern gespeichert werden müssen, werden von den abgeleiteten Klassen `Angestellter` und `Arbeiter` bereitgestellt. Für Angestellte muss das Monatsgehalt gespeichert werden, für Arbeiter der Stundenlohn und die Anzahl der im aktuellen Monat gearbeiteten Stunden.

Beide Klassen verfügen jeweils über eine Methode `mitarbeiterInfo()` mit der gleichen Signatur wie in der Basisklasse, die alle gespeicherten Informationen ausgibt. Für einen Angestellten z. B. sind dies Name, Rentenversicherungsnummer, Status und Monatsgehalt. Die Methode `mitarbeiterInfo()` der abgeleiteten Klassen ruft jeweils die Methode `mitarbeiterInfo()` der Basisklasse auf, um Name und Rentenversicherungsnummer auszugeben.

- Geben Sie Beschreibungen der Klassen an. Überlegen Sie sich, mit welchen Zugriffsrechten Attribute und Methoden versehen sein sollten.
- Geben Sie die UML-Spezifikationen der Klassen an.
- Implementieren Sie die Klassen. Testen Sie Ihre Klassen z. B. mit dem folgenden Programm:

```

class MitarbeiterTest
{
    public static void main(String [] args)
    {
        // zwei Angestellte und einen Arbeiter deklarieren ...
        Angestellter boss, dilbert;
        Arbeiter dogbert;

        // ... und erzeugen
        boss = new Angestellter(
            "The Pointy Haired Boss", "221-45-7632", 15000);
        dilbert = new Angestellter("Dilbert", "234-67-8901", 7500);
        dogbert = new Arbeiter("Dogbert", "896-54-3217", 250.00, 160);

        // alle Informationen ueber Dilbert ausgeben
        dilbert.mitarbeiterInfo();

        // das Gleiche fuer den Chef
        boss.mitarbeiterInfo();

        // Dogbert erhaelt einen hoeheren Stundenlohn
        dogbert.setStundenlohn(300.00);

        // alle Informationen ueber Dogbert ausgeben
        dogbert.mitarbeiterInfo();
    }
}

```

/\*

Ausgabe des Programms:

```

Name:    Dilbert
RV-Nr.: 234-67-8901
Status: Angestellter
Gehalt: 7500.0

```

```

Name:    The Pointy Haired Boss
RV-Nr.: 221-45-7632
Status: Angestellter
Gehalt: 15000.0

```

```

Name:    Dogbert
RV-Nr.: 896-54-3217
Status: Arbeiter
Stundenlohn:    300.0
Anzahl Stunden: 160.0
Arbeitslohn:    48000.0

```

\*/

Das Programm ist in ueb5.zip enthalten

### Aufgabe 3

Ergänzen Sie die Klassen aus Aufgabe 2 jeweils um eine Methode

```

void tarifErhoehung(double prozent)

```

die Gehalt bzw. Stundenlohn des Mitarbeiters entsprechend anpasst. Warum sollte diese Methode in der Klasse `Mitarbeiter` abstrakt sein? Testen Sie Ihre Klassen mit einem geeigneten Hauptprogramm.