

Aufgabe 1

Das **Klassendiagramm** für den Gabelstapler aus den Folien wurde etwas erweitert:

- Die Methode `heben()` erhöht das Attribut `höhe` um 10, `senken()` verringert es entsprechend um 10.
- `oben()` setzt das Attribut `höhe` auf den festen Wert 200, `unten()` entsprechend auf 0.
- `neigenVor()` erhöht das Attribut `winkel` um 5, `neigenZurück()` verringert es entsprechend.
- `neigenMitte()` setzt `winkel` auf 0.

StaplerStrg
höhe, winkel: double
heben() senken() oben() unten() neigenVor() neigenZurück() neigenMitte()

- a) **Implementiere** die Klasse mit BlueJ.
- b) **Übersetze** die Klassen und **erzeuge** mit BlueJ zwei Objekte der Klasse `StaplerStrg`. Die Objekte sollen `st1` und `st2` heißen.
- c) **Führe** folgende Methoden nacheinander **aus**:
Für `st1`: `unten` – `heben` – `heben` – `heben` – `neigenVor` – `neigenVor`
Für `st2`: `oben` – `senken` – `senken` – `neigenZurück` – `mitte` – `neigenZurück` – `heben`
- d) Prüfe die Werte für beide Objekte mit dem **Objektinspektor**.
- e) Zeichne mit UMLet ein **Objektdiagramm** der beiden Objekte nach der Ausführung der Methoden in Aufgabe c).

Aufgabe 2

In dieser Aufgabe sollst du ein Klassenmodell erstellen und es dann implementieren.

Eine Bank möchte ihre Konten mithilfe einer Software verwalten.

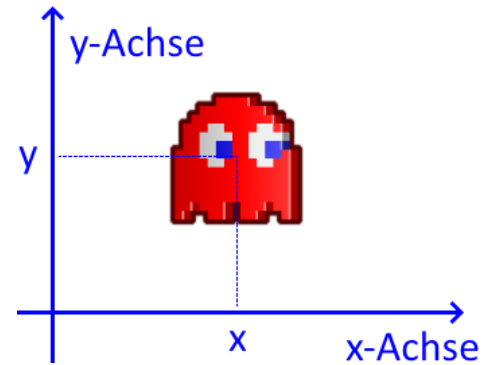
Es sollen zunächst nur die Konten modelliert werden:

- Jedes Konto hat einen **Kontostand** (in Euro)
 - Außerdem hat jedes Konto einen **Zinssatz** (Beispiel: ein Wert von 0,05 entspricht 5% Zinsen pro Jahr auf den Kontostand). Der Zinssatz kann sich im Laufe der Zeit ändern.
 - Es soll eine Methode geben, um Geld ins Konto **einzuzahlen**. Da wir noch keine Parameter kennen, soll der Kontostand sich mit jeder Einzahlung pauschal um 100 € erhöhen. Entsprechend gibt es eine Methode zur **Auszahlung**, die 100 € vom Kontostand abzieht.
 - Es soll eine Methode geben, um den Zinssatz zu **erhöhen**. Eine Ausführung der Methode erhöht den Zinssatz um 1% (d.h. um 0,01). Entsprechend eine Methode zum **senken** des Zinssatzes.
 - Eine Methode soll die monatlichen Zinsen zum aktuellen Kontostand **hinzuaddieren**.
 - Achtung: Der Zinssatz stellt die Jahreszinsen dar. Im Monat wird also nur ein Zwölftel dieser Zinsen ausgezahlt.
 - Hinweis: Die Methode implementiert den Vorgang *einer* Abrechnung. Über die Frage, wie diese dann monatlich ausgeführt wird, brauchst du dir an dieser Stelle keine Gedanken zu machen.
 - Hilfestellung: Die Grundrechenarten werden in Java mit +, −, * und / dargestellt.
- a) Stelle das Kontomodell als **Klassendiagramm** dar.
- b) **Implementiere** das Modell entsprechend deines Klassendiagramms.
- c) **Übersetze** deine Klasse.
Erzeuge ein Objekt und prüfe, ob deine Methoden korrekt funktionieren.

Aufgabe 3

In zweidimensionalen Spielen gibt es Objekte, die sich bewegen – z.B. Super Mario, seine Gegner, Powerups etc. Solche beweglichen, graphischen Objekte nennt man „**Sprite**“. Sie bestehen aus einem oder mehreren Bildern und bewegen sich innerhalb eines Koordinatensystems.

Der folgende Quellcode implementiert eine einfache Version eines Sprites (noch ohne Bilder):



```
class Sprite
{
    double x, y;
    double richtung;
    double speed;

    void zumUrsprung()
    {
        x = 0;
        y = 0;
    }

    void stop()
    {
        speed = 0;
    }

    void schneller()
    {
        speed = speed + 10;
    }

    void langsamer()
    {
        speed = speed - 10;
    }

    void dreheLinks()
    {
        richtung = richtung + 15;
    }

    void dreheRechts()
    {
        richtung = richtung - 15;
    }

    void bewege()
    {
        x = x + Math.cos(richtung)*speed;
        y = y + Math.sin(richtung)*speed;
    }
}
```

- Zeichne ein **Klassendiagramm** zu diesem Quellcode.
- Ein Sprite-Objekt namens sp1 wird erzeugt und die folgenden Methoden werden nacheinander für dieses Objekt ausgeführt:
zumUrsprung – schneller – schneller – dreheLinks – dreheLinks – dreheLinks – bewege – bewege – dreheRechts – dreheRechts – dreheRechts – langsamer – bewege

Skizziere den Weg des Sprites in ein **Koordinatensystem** (x- und y-Achse jeweils bis 50).

Berechne dazu, wie sich x und y durch die Methoden verändern.

Achte darauf, dass der Taschenrechner bei sin/cos im Gradmaß (nicht im Bogenmaß) rechnet.

- Zeichne ein **Objektdiagramm** für den Sprite sp1 nach den Methodenaufrufen in b).
- Beschreibe** mit eigenen Worten, welche Bedeutung deiner Meinung nach die Attribute und welche Funktion die einzelnen Methoden haben (zu jedem Attribut / jeder Methode ein Satz).