Aufgabe 1

a) Implementierung

```
class StaplerStrg
                                           void unten()
   double höhe, winkel;
                                               h\ddot{o}he = 0;
   void heben()
                                           void neigenVor()
      h\ddot{o}he = h\ddot{o}he + 10;
                                               winkel = winkel + 5;
   void senken()
                                           void neigenZurück()
      h\ddot{o}he = h\ddot{o}he - 10;
                                               winkel = winkel - 5;
   void oben()
                                           void neigenMitte()
      h\ddot{o}he = 200;
   }
                                              winkel = 0;
                                        }
```

d) Objektdiagramm

st1: StaperStrg

höhe = 30 winkel = 10 st2: StaperStrg

höhe = 190 winkel = -5

Aufgabe 2

a) Klassendiagramm

```
Konto
stand: double
zinssatz: double
einzahlen()
auszahlen()
zinsErhöhen()
zinsSenken()
abrechnung()
```

b) Implementierung

```
class Konto
{
    double stand;
    double zinssatz;

    void einzahlen()
    {
        stand = stand + 100;
    }

    void auszahlen()
    {
        stand = stand - 100;
    }

    void zinsErhöhen()
    {
        zinssatz = zinssatz + 0.01;
    }

    void zinsSenken()
    {
        zinssatz = zinssatz - 0.01;
    }

    void abrechnung()
    {
        stand = stand + (stand * zinssatz / 12.0);
    }
}
```

Aufgabe 3

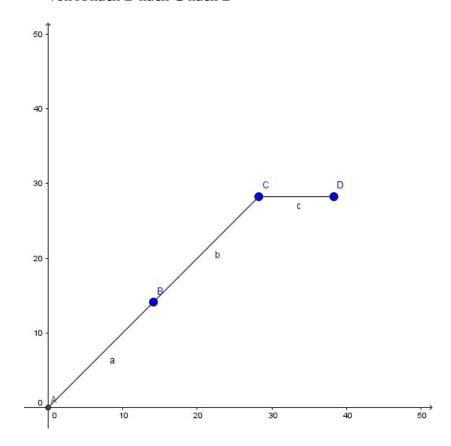
a) Klassendiagramm

Sprite	
x, y: double speed: double richtung: double	
zumUrsprung() stop() schneller() langsamer() dreheLinks() dreheRechts() bewege()	

c) Objektdiagramm

sp1: Sprite
x = 38.28
x = 38.28 y = 28.28
speed = 10.0
richtung = 0.0

b) Weg des Sprite im Koordinatensystem: von A nach B nach C nach D



d) Bedeutung von Attributen und Methoden

x, y	die aktuelle Position des Objekts im Koordinatensystem
speed	distanz, mit der sich das Objekt bei jeder Bewegung weiterbewegt
richtung	winkel im Koordinatensystem, in dem sich das Objekt bewegt
zumUrsprung()	setzt das Objekt an den Punkt (0 / 0)
stop()	setzt die Bewegungsgeschwindigkeit auf 0
schneller()	erhöht die Geschwindigkeit
langsamer()	verringert die Geschwindigkeit
dreheLinks()	dreht die Richtung um 15° nach links
dreheRechts()	dreht die Richtung um 15° nach rechts
bewege	ändert die Position des Objekts entsprechend von speed und richtung

Autor: Christian Pothmann – <u>cpothmann.de</u> Freigegeben unter <u>CC BY-NC-SA 4.0</u>, März 2021

