
5. Übungsblatt

12. Aufgabe

Gegeben sind folgende Klassen:

```
Class Point
{
    private:
        int x, y;
    public:
        Point() {x = y = 0;}
};

class Graphobj
{
};

class Rechteck : public Graphobj
{
    private:
        Point p1, p2;
    public:
        Rechteck () {}
};

class Kreis : public Graphobj
{
    private:
        Point mittelpkt;
        float radius;
    public:
        Kreis () {}
};
```

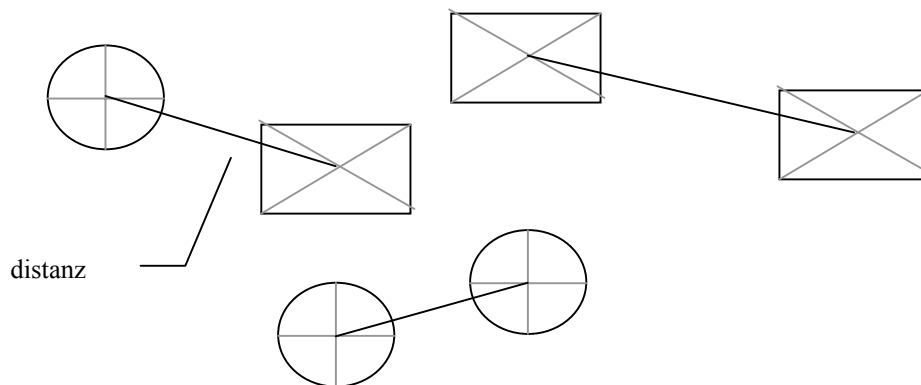
a) Stellen Sie die Klassenhierarchie zwischen diesen Klassen dar

In den folgenden Aufgabenteilen sollen Sie die Klassen ergänzen; Sie dürfen allerdings die angegebenen Klassenkomponenten nicht verändern

b) Es fällt auf, dass die Klasse Point ihre beiden Komponenten vor unbefugtem Zugriff aus einer anderen Klasse schützt.

Erweitern Sie Point dahingehend, dass die beiden Komponenten separat gelesen(abgefragt) und gesetzt werden können.

- c) Jedes Objekt von der Klasse Graphobj bzw. von Rechteck und Kreis soll einen Referenzpunkt besitzen (beim Kreis ist es der Mittelpunkt, beim Rechteck dessen Schwerpunkt). Geben Sie alle notwendigen Erweiterungen an (bei dieser Aufgabe soll Vererbung ausgenutzt werden!)
- d) Der Abstand zweier graphischer Objekte (Graphobj) soll der Abstand ihrer Referenzpunkte sein (siehe Skizze). Geben Sie die Anweisungen (Klassenerweiterungen) an, damit diese Methode zur Verfügung steht (z.B. `d = g1->distanz(g2)`)



13 Aufgabe (Praktikum)

Eine Mietfahrzeug-Agentur benötigt ein Verwaltungsprogramm für Ihren Fuhrpark, in dem folgende Fahrzeuge existieren:

- PKW
 - Limousine
 - Kombi
 - Cabrio
 - Bus
- LKW
- Anhänger
 - LKW-Anhänger
 - PKW-Anhänger
 - Wohnwagen

Diese Fahrzeuge sollen auf eine Klassenhierarchie abgebildet werden, in der folgende Eigenschaften und Methoden existieren:

A) Alle Fahrzeuge

- a. Eigenschaften:
 - i. Bezeichnung
 - ii. Verleihkosten je Tag
 - iii. Baujahr
 - iv. Verfügbar (ja/nein)
- b. Methoden
 - i. Einlesen (Eigenschaften beim Benutzer abfragen)
 - ii. Ausgabe (Eigenschaften am Bildschirm ausgeben)
 - iii. FahrzeugVerleihen (nicht mehr verfügbar!)
 - iv. FahrzeugZurücknehmen (wieder verfügbar!)

B) PKW (zusätzlich zu Fahrzeuge)

- a. Eigenschaften
 - i. KW
 - ii. Höchstgeschwindigkeit
 - iii. Sitzplätze

C) LKW (zusätzlich zu Fahrzeuge)

- a. Eigenschaften
 - i. KW
 - ii. Achsen
 - iii. Max. Zuladung

D) Anhänger (zusätzlich zu Fahrzeuge)

- a. Eigenschaften
 - i. Max. Zuladung
 - ii. Zugfahrzeugtyp (PKW/LKW)

Analog zu diesen Angaben sollen die aufgeführten fehlenden Fahrzeugtypen ergänzt werden (mögliche Angaben: Schlafplätze, Anzahl Türen/Schiebetüren, Kofferraumvolumen ⇔ Sitzplätze, Verdeckart,...)

Mit diesen Klassen soll dann eine (einfache) Fahrzeugverwaltung geschrieben werden. Hierbei sollen folgende Abläufe abgebildet werden:

- a) Eingabe von Fahrzeugen
- b) Ändern von Fahrzeugen
- c) Löschen von Fahrzeugen
- d) Anzeige aller Fahrzeuge
- e) Anzeige aller verfügbaren Fahrzeuge
- f) Fahrzeug als verliehen/verfügbar setzen

Die Verweise auf die Fahrzeuge sollen hierbei in einem Array gehalten werden. Sie können davon ausgehen, dass maximal 100 Fahrzeuge existieren.