

Themen:

Stoff der Vorlesungskapitel 2 (Objekt-Attribute und -Methoden)
und 3 (Konstruktoren und Referenzen)

Aufgabe 2.1

Betrachten Sie die Klassendefinition von `OOP1K` und die Testklasse `TestOOP1K`. (Sie finden den Quellcode dieser beiden Klassen im LEA-Ordner „Code“ unter „Dateien zu Aufgabe U02A1“.)

Erläutern Sie anhand eines schematischen Speicherabbilds die Situation nach dem Initialisieren der Variablen (in der `main`-Methode: Nach Ausführen der Code-Zeile 11).

Versuchen Sie **ohne das Programm laufen zu lassen (!)** vorherzusagen, wie die Ausgabe am Ende des Programms aussieht.

Auch wenn Sie für die folgende Aufgabe bereits eine Lösung (zB aus den Vorlesungsfolien von A. Priesnitz) kennen: Versuchen Sie es noch einmal selbst!

Aufgabe 2.2

a) Definieren Sie eine Klasse `Punkt2D` zur Darstellung von Punkten in der 2-dimensionalen Ebene mit den folgenden Eigenschaften:

- Jeder Punkt ist durch seine beiden (double-) Koordinaten gegeben.
- Es gibt drei Konstruktoren zur Erzeugung von Punkten:
 1. einen Konstruktor ohne Parameter: er soll den Ursprung (Nullpunkt) erzeugen
 2. einen Konstruktor, dem zwei beliebige Koordinaten des Punktes übergeben werden
 3. einen Konstruktor, der ein (bereits existierendes) Punkt-Objekt als Parameter erhält und eine Kopie dieses Objektes erzeugt (einen Klon, also einen weiteren Punkt mit den gleichen Koordinaten)
- Die Koordinaten eines Punktes können durch `get`-Methoden einzeln abgefragt werden.
- `verschiebeUm`: „verschiebt“ den gegebenen Punkt um (dx, dy) (entspricht Vektoraddition)
- `moveTo`: „bewegt“ den gegebenen Punkt an die übergebenen Koordinaten (x, y)
- `toString`: erzeugt einen String, der die Koordinaten eines Punktes durch Komma getrennt und in runde Klammern eingeschlossen, enthält.
- * `distanceTo`: berechnet den Abstand zwischen dem gegebenen Punkt und einem beliebigen anderen Punkt

Hinweis:

Den Abstand eines Punktes $P = (x_1, y_1)$ vom Punkt $Q = (x_2, y_2)$ berechnet man nach dem Satz des Pythagoras aus $d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$.

Berechnen Sie das Quadrat einer Zahl *nicht* durch Verwendung der Methode `Math.pow`, sondern durch explizite Multiplikation mit sich selbst! Die Quadratwurzel eines Wertes z erhalten Sie mit `Math.sqrt(z)`

b) Schreiben Sie eine Testklasse, in der Sie (mindestens drei) Punkte erzeugen, wobei jeder Konstruktor-Typ mindestens einmal verwendet wird. Geben Sie anschließend die erzeugten Punkte auf dem Monitor aus.

c) Schreiben Sie verschiedene Testmethoden, um alle Methoden (mit unterschiedlichen Punkten) zu testen.

Aufgabe 2.3

Ein Kreis (in der euklidischen Ebene) ist durch die Koordinaten seines Mittelpunkts und den Radius gekennzeichnet.

Entwickeln Sie eine Klasse zur Modellierung von Kreisen mit folgenden Tools:

- Erzeugung eines Kreises durch Angabe von Mittelpunkt und Radius
- Erzeugung eines Kreises mit den Default-Werten Mittelpunkt = Ursprung, Radius = 1.0
- Verschiebung des Kreises um δ_x, δ_y
- Skalierung um einen Faktor (Vergrößerung/Verkleinerung des Radius')
- Berechnung der Fläche
- Berechnung des Umfangs
- Test, ob ein gegebener Punkt innerhalb des Kreises liegt (ein Punkt liegt innerhalb eines Kreises, wenn sein Abstand vom Mittelpunkt höchstens gleich dem Radius ist)

Nutzen Sie die Klasse `Punkt2D`.

(Auch diese Aufgabe ist noch ohne `if`-statement, also allein mithilfe boolescher Ausdrücke lösbar.)