

## Übungsblatt 5

Abgabe bis Montag, 30.5.2016, 23:59 Uhr

### Hinweis:

Aufgaben immer per E-Mail (eine E-Mail pro Blatt und Gruppe) an den zuständigen Tutor schicken (bei Programmieraufgaben Java Quellcode und eventuell benötigte Datendateien).

### Aufgabe 5.1

Betrachten Sie folgenden Programmcode (Source-Code auf der Web-Seite).

```
public class Numbers {  
    public static void main (String args[]) {  
        double f = 2 / 4 * 0.25;           // Zeile 3  
        double g = 7;  
        double h = 1 / g;  
        int i = 7;  
        System.out.println(f);           // Zeile 7  
        System.out.println(1 / g * g);  
        System.out.println(1 / i * i);  
        f = (g * h) * (g * h) * (g * h);  
        System.out.println(f - 1.0);  
        f = (g * g * g) * (h * h * h);  
        System.out.println(f - 1.0);  
    }  
}
```

1. Welche Ausgaben liefert das Programm? Lösen Sie die Aufgabe zunächst ohne den Compiler zu verwenden. Begründen Sie Ihre Antwort!
2. Modifizieren Sie den arithmetischen Ausdruck in Zeile 3 ohne Veränderung der Zahlwerte und der Operatoren so, dass in Zeile 7 der Wert 0.125 ausgegeben wird.

## Aufgabe 5.2

Wie lautet die Ausgabe des folgenden Java-Programms?

```
class PlusPlus {
    public static void main(String[] arg) {
        int x = 3, y = 10;
        System.out.println( x++ );
        System.out.println( x + y );
        System.out.println( ++y + x );
        int z = ++y / x;
        y = z;
        z *= y;
        System.out.println( z );
        System.out.println( y * ++z - x++ );
        System.out.println( x );
    }
}
```

## Aufgabe 5.3

In dieser Aufgabe soll eine Klasse `Point2D` erstellt werden, die zwei-dimensionale Punkte darstellt, wobei die  $x$ - und  $y$ -Koordinaten jeweils als `double` Variablen gespeichert werden. Nutzen Sie die Java API Dokumentation um sich ggf. mit den Methoden der Klasse `java.lang.Math` vertraut zu machen, welche Sie zum Lösen dieser Aufgabe benötigen.

1. Schreiben Sie die Klasse `Point2D` mit einem Konstruktor, der es erlaubt die Koordinaten zu setzen. Die Klasse soll zusätzlich über Methoden zum Auslesen und Setzen der Koordinaten verfügen.
2. Erweitern Sie die Klasse `Point2D` um eine Methode zum Berechnen des Euklidischen Abstands zu einem anderen Objekt der Klasse `Point2D` mit dem folgenden Prototyp:

```
double distance(Point2D pt);
```

3. Schreiben Sie ein Programm, das die erstellte Klasse verwendet. Dieses Programm soll den Benutzer zur Eingabe zweier Punkte auffordern. Anschließend sollen die Koordinaten der beiden Punkte sowie deren Euklidischer Abstand auf dem Bildschirm ausgegeben werden.

## Aufgabe 5.4

Betrachten Sie einen Auszug aus der Klasse `Circle`

```
class Circle {  
    public Circle(Point2D center, double radius) { ... }  
    public void setCenter() { ... }  
    public Point2D getCenter() { ... }  
    public void setRadius() { ... }  
    public double getRadius() { ... }  
    public double getArea() { ... }  
    public double getCircumference() { ... }  
    public double distance(Circle c) { ... }  
  
    private Point2D center;  
    private double radius;  
}
```

1. Vervollständigen Sie den Konstruktor sowie die set- und get- Methoden der Klasse `Circle`.
2. Vervollständigen Sie die Methoden `getArea` und `getCircumference`, die den Flächeninhalt bzw. den Umfang des Kreises zurückgeben.
3. Vervollständigen Sie die Methode `distance`, die den Abstand  $d$  zu einem anderen Kreis berechnet (siehe Abbildung). Sollten sich die Kreise überschneiden, soll eine negative Zahl zurückgegeben werden.

