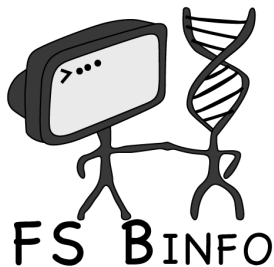


Objektorientierte Programmierung



Sebastian Pölsterl

Fachschaft Bioinformatik

24. November 2009

Zeitplan

- 27.10 Einführung in Linux und die Shell
 - 03.11 Arbeiten mit Dateien auf der Shell
 - 10.11 Einführung in Java
 - 17.11 Umgang mit Arrays, Schleifen und Dateien
 - 24.11 **Einführung in objekt-orientierte Programmierung**
 - 01.12 Rekursion und Iteration
 - 08.12 GUI Entwicklung
 - 15.12 Debuggen mit Eclipse
-
- 12.01 Taschenrechner in Java I
 - 19.01 Taschenrechner in Java II
 - 26.01 Wiederholung

Webseite: <http://www.bioinformatik-muenchen.com/bioinfocom/informatik-tutorium>

Was ist Objektorientierte Programmierung (OOP)?

- Der Mensch nimmt die Welt als eine Sammlung von Objekten wahr
- Logische Einheiten in Objekte verpacken um die Realität besser abbilden zu können
- Die vier Grundsäulen:
 - Abstraktion
 - Kapselung
 - Vererbung
 - Polymorphie



Was ist eine Klasse/Objekt?

- Werden mit `class` gekennzeichnet
- Klassenname wird meist groß geschrieben
- Können Attribute und Methoden haben (Zugriff mit `.`)
- Instanzen von Objekten werden mit `new` erzeugt

```
class Counter {  
    int counter = 0; /* Attribut */  
    int increment() { /* Methode */  
        counter++;  
        return counter;  
    }  
  
    public static void main (String[] args) {  
        Counter c = new Counter ();  
        System.out.println(c.increment ());  
    }  
}
```

Konstruktoren

- Werden bei der Konstruktion eines Objekts mit `new` aufgerufen
- Besondere Signatur: kein Rückgabewert, selber Name wie Klasse
- Mit `this` greift man auf Attribute und Methoden der eigenen Klasse zu
- Hilft wenn lokale Variablen den selben Namen wie Objektvariablen haben

```
class Counter {  
    int counter;  
    public Counter(int initial_value) {  
        this.counter = initial_value;  
    }  
}
```

Übung 1

1. Erstelle eine Klasse Kreis mit den Attributen x , y und durchmesser
2. Der Konstruktor soll alle 3 Parameter erwarten
3. Implementiere die Methoden berechneOberflaeche und berechneVolumen die den Umfang bzw. das Volumen zurückliefern sollen



Statische Methoden und Attribute

- Methoden und Attribute können als `static` deklariert werden
- Im Gegensatz zu Objektvariablen gelten statische Variablen für alle Instanzen der Klasse gleichermaßen
- Statische Methoden können auch aufgerufen werden ohne vorher eine Instanz des Objekts zu erzeugen (siehe `main`-Methode)

```
class Counter {  
    static int counter;  
    static int increment() {  
        counter++;  
        return counter;  
    }  
    public static void main (String [] args) {  
        System.out.println(Counter.increment ());  
    }  
}
```

Übung 2

1. Erstelle ein statisches Attribut "anzahl" in der Klasse Kreis, die angibt wie viele Instanzen der Klasse Kreis erzeugt wurden

Übungen für Daheim

1. Erstelle die Klassen Trapez, Rechteck und Quadrat und implementiere die Methoden berechneOberflaeche und berechneVolumen wie in Übung 1
2. In welcher Beziehung stehen die Objekte Trapez, Rechteck und Quadrat?