

Repetitorium: Strukturierte Programmierung

08.04.15

Felix Epp



h_da



ia



h_da

 **Fraunhofer**
SIT



Termine und Ausblick

Heute: Repetitorium und Fotos

Morgen: Kick Off

MIT:

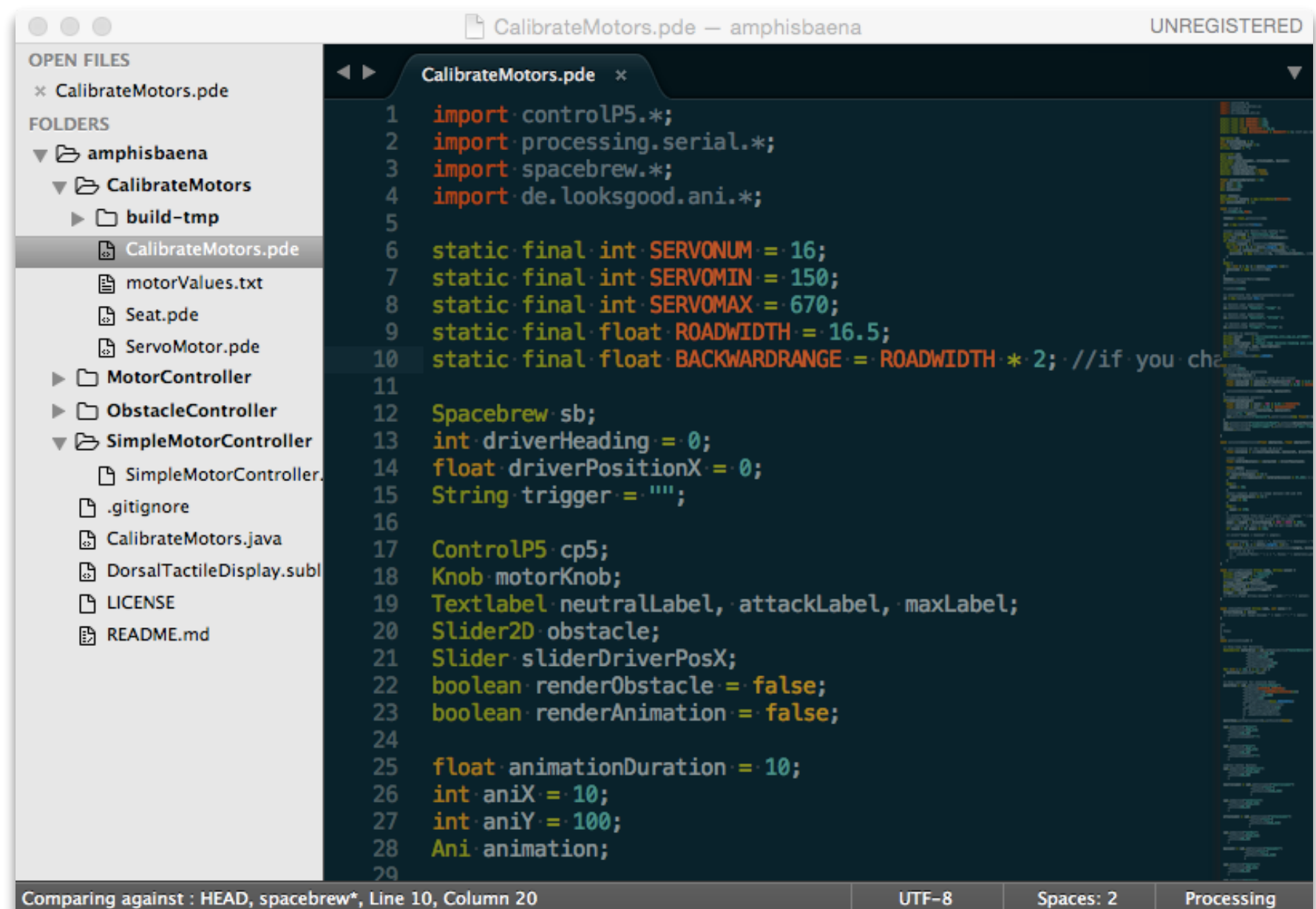
Objekt-orientierte Programmierung
Anwendungsentwicklung



Favorite Tool



Sublime Text 3
Package Control
Processing

A screenshot of the Sublime Text 3 editor interface. The title bar shows 'CalibrateMotors.pde — amphisbaena' and 'UNREGISTERED'. The left sidebar shows the 'OPEN FILES' and 'FOLDERS' panels. The 'FOLDERS' panel shows a tree structure with 'amphisbaena' as the root, containing 'CalibrateMotors' and 'build-tmp'. The 'OPEN FILES' panel lists several files, including 'CalibrateMotors.pde', 'motorValues.txt', 'Seat.pde', 'ServoMotor.pde', 'MotorController', 'ObstacleController', 'SimpleMotorController', 'SimpleMotorController.', '.gitignore', 'CalibrateMotors.java', 'DorsalTactileDisplay.subl', 'LICENSE', and 'README.md'. The main editor area shows the code for 'CalibrateMotors.pde'. The code is in a dark theme with syntax highlighting. It includes imports for 'controlP5.*', 'processing.serial.*', 'spacebrew.*', and 'de.looksgood.ani.*'. It defines several static final variables: 'SERVONUM' (16), 'SERVOMIN' (150), 'SERVOMAX' (670), 'ROADWIDTH' (16.5), and 'BACKWARDRANGE' (ROADWIDTH * 2). It also declares objects for 'Spacebrew', 'ControlP5', 'Knob', 'Textlabel', 'Slider2D', and 'Slider'. The code is currently at line 10, column 20. The status bar at the bottom shows 'Comparing against : HEAD, spacebrew*, Line 10, Column 20', 'UTF-8', 'Spaces: 2', and 'Processing'.

Funktionen

Bestehen aus...

- Name
- Parameter
- Anweisungen
- (Rückgabewert)

Nützt...

- Wiederverwendbarkeit von Codeabschnitten
- Strukturierung in „funktionelle“ Einheiten

```
void drawTarget(float xloc, float yloc, int size, int num) {  
    float grayvalues = 255/num;  
    ...  
}
```

Arrays

Besitzen...

- Variablentyp
- Mehrere Elemente
 - mit Indizes
 - und Wert
- Länge (length)

```
float[] nus = new float[255];  
nus[0] = 0.1;
```

Dient...

- größeren Datenmengen
- Iteration in Schleifen

Built in...

```
length; append(); arrayCopy();  
concat(); expand(); reverse();  
shorten(); sort(); splice();  
subset();
```

Arrays: 2D, Lists, Assoziativ

2 Dimensional Arrays

- Tabellarische Daten
- z.B. Objekte x y verteilt

```
int[][] px1 = new int[40][40];  
px1[0][0] = 255;
```

Lists

- haben undefinierte Länge
- komplexere Handhabung

```
ArrayList<Ball> balls;  
balls = new ArrayList<Ball>();
```

Assoziative Arrays

- Strings als Indizes (z.B. PHP)

```
$array = [  
    "red" => "f00",  
    "blue" => "00f"  
];
```

Dokumentation

Kommentiere wichtige
Anweisungsblöcke und
Funktionen

```
/**  
 * Functions xy does something!  
 */  
void someFunction() {  
    int a = 3 * 2; //math!!!  
}
```

Selbsterklärende Namen für
Funktionen/Variablen

```
int calcSomething(int xWidth) {  
    return xWidth * 5;  
}
```

Spezielle Notationen

```
camelCaseIsEasierToRead();  
Boolean bUseTypePrefix = true;
```


Dokumentation

Unsauberer Code macht
Einhörner traurig!



```
/*daskannjagarkeinerlesen*/  
void malOhneLeerzeichen(){  
    //nix
```

```
}  
void malMit() {  
    //nix
```

```
}  
void malNochMehr()  
{  
    //nix
```

```
    t no = dies+das*(mehr(von)  
    ())/nochwas);
```

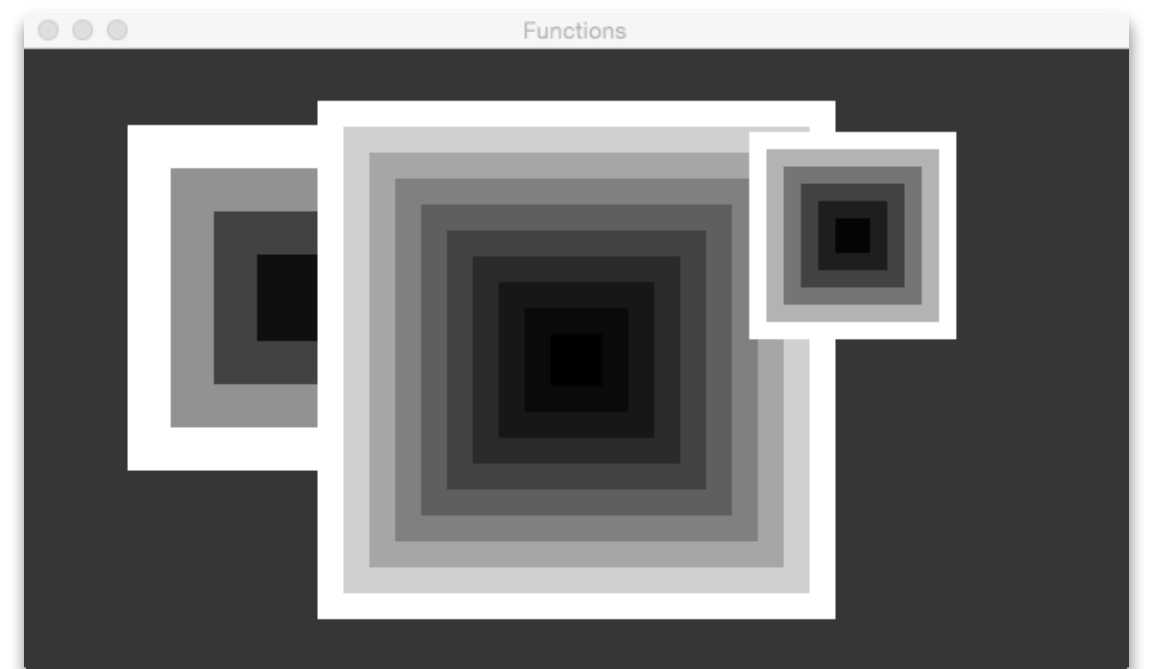


Say cheez...



Funktionen: Praxis

1. Öffne Processing > Examples > Structure > Functions
2. Spiele mit den Parametern für's Verständnis
3. Ändere die Form der Objekte (z.B. `rect()`)
4. Erstelle und nutze eine Funktion die den Grauwert zurück gibt
5. Optional: Farbe? Farbwert nicht linear sondern anhand der Fläche?



Arrays: Praxis

1. Speichere den Code aus der vorherigen Aufgabe unter neuem Namen
2. Erstelle Arrays und fülle sie mit den drawTarget() Parametern
3. Baue eine Schleife, welche die Werte ausließt und drawTarget() aufruft
4. Nutze array.length
5. Füge weitere Objekte hinzu, indem du die Arrays füllst

Praxis: Rekursion

1. Öffne Beispiel > Basic > Structure > Recursion
2. Spiele mit den Parametern der Funktion `drawCircle()`
3. Öffne einen neuen Sketch
4. Schreibe eine rekursive Funktion die eine Zahl, so lange durch 2 teilt solange die eingegebene Zahl gerade ist. Die ungerade Zahl sollte am Ende ausgegeben werden. Nutze den Modulus Operator „%“
5. Wer fertig ist, geht herum und hilft

Mehr?

Generative Gestaltung 1/4 Pecha Kutcha

- Wähle eine Methode
 - Rauschen (Gaussian Distribution, Perlin Noise, etc.)
 - Rekursion (Baumdiagramme, Fractals, etc.)
 - Kurven/Schwingungen (Sinus & Co, Lissajous-Figur, etc.)
 - Physik (Attractor, ParticleSystems, etc.)
- Bereite eine Live Demo vor die maximal 1:40 Minuten dauert

Material

- Processing Examples: Math, Fractals and L-Systems, Simulate, Geometry > NoiseSphere
- <http://www.generative-gestaltung.de/code>
- <https://processing.org/tutorials/>

Mehr?

Design Patterns 1/4 Pecha Kutcha

Suche einen Design Pattern aus

Erkläre:

- Struktur (Beispiel in Processing)
- Einsatzgebiete
- Zusätzliche Infos / Abwandlungen

Format:

- 5 Folien
- 20 Sekunden pro Folie (automatisch)
- Dateiformat frei (Keynote, Powerpoint, PDF, ...)

Lunch Break?

First known LolCat

Origin of LolSpeak

Origin of familiar Meme styling
with fat caption on picture

<http://knowyourmeme.com/memes/happy-cat>

