

Programmiergrundlagen

Hans Christian Feßl, BA

Überblick

Programmiersprachen

- Einsatz
- Arten
- Pseudocode
- Algorithmus / Semantik / Syntax

Programmiergrundlagen - JavaScript

- JavaScript definieren
- Ausgaben
- Variablen
- Datentypen (Werte)
- Operatoren
- Kontrollstrukturen – Bedingte Anweisungen
- Kontrollstrukturen – Schleifen
- Funktionen
- Kommentare

Was ist Programmieren

Programmieren ist das beschreiben eines logischen
Ablaufes

in einer für Systeme Verarbeitbarkeit Sprache.

Ähnlich

wie die Bauanleitung für ein Möbel,

oder ein Rezept für ein Gericht.

The background features a series of concentric circles in light gray, some solid and some dashed, creating a ripple effect. A large, solid red speech bubble is centered on the page, pointing downwards.

Einsatz - Programmiersprachen

Einsatz - Programmierspr achen

- Webanwendungen
- Spiele
- Büroanwendungen
- Wissenschaftliche Berechnungen
- Maschinensteuerung
- Etc.

Einsatz - Programmierspr achen

- Datenverarbeitung
- Eingabe vom Benutzer
- Verarbeiten der Eingabe
- Ausgabe der Ergebnisse
- Etc.

Arten- Programmierspr achen

3 Arten von Sprachen

- Maschinensprachen
- Assemblersprachen
- Hochsprachen
 - PHP, Javascript, Java, C++

Arten- Maschinenspra- chen

BA 0B 01 B4 09 CD 21 B4!

4C CD 21 48 61 6C 6C 6F!

2C 20 57 65 6C 74 21 24 !

01011101000001011000000011011010000001

00111001101001000011011010010011001100

11010010000101001000011000010110110001

10110001101111101100001000000101011101

10010101101100011101000010000100100100

Arten- Assemblersprache

```
.section .data!  
s: .ascii "Hallo Welt!"!  
  
.section .text!  
.globl _start!  
_start:!  
  
movl $4,%eax!! # Ausgabe Einleiten!  
movl $1,%ebx!! # Direkt ausgeben!  
movl $s,%ecx!! # Erstes Zeichen!  
movl $12,%edx!! # Länge der Zeichenkette !  
int $0x80!!! # Ausgabe durchführen!  
movl $1,%eax!! # Anweisung beenden!  
movl $0,%ebx!! # Rückgabewert 0 (=alles ok)!  
int $0x80!!! # Programm beenden!
```

Arten - Hochsprachen

- PHP

```
echo 'Hallo Welt! ';
```

- Javascript

```
document.write('Hallo Welt');
```

- Java

```
Public class HelloWorld {  
    public static void main(String[] args)  
        System.out.println("Hello World!")  
}
```

Pseudocode

Aussprechen was zu tun ist.

Pseudocode - Anmeldung

1. Prüfe Passwort.
2. Wenn Passwort korrekt:
 1. begrüße Benutzer
3. anderenfalls:
 1. Melde Fehler

Pseudocode - Anmeldung

1. Unterprogramm ausführen
2. Vergleich/Entscheidung
 1. Ausgabe
3. Vergleich/Entscheidung
 1. Ausgabe

Pseudocode - Anmeldung

1. Prüfe Passwort.
2. Wenn Passwort korrekt:
 1. begrüße Benutzer
3. anderenfalls:
 1. Melde Fehler

1. Unterprogramm ausführen
2. Vergleich/Entscheidung
 1. Ausgabe
3. Vergleich/Entscheidung
 1. Ausgabe

Pseudocode - Anmeldung

1. Prüfe Passwort.
2. Wenn Passwort korrekt:
 1. begrüße Benutzer
3. anderenfalls:
 1. Melde Fehler
1. Check password
2. If password is correct:
3. Welcome User
4. Else:
5. Display Error

1. Unterprogramm ausführen
2. Vergleich/Entscheidung
 1. Ausgabe
3. Vergleich/Entscheidung
 1. Ausgabe

Pseudocode - Anmeldung

1. Prüfe Passwort.
2. Wenn Passwort korrekt:
 1. begrüße Benutzer
3. anderenfalls:
 1. Melde Fehler

1. Check password
2. If password is correct:
 1. Welcome User
3. Else:
 1. Display Error

1. Unterprogramm ausführen
2. Vergleich/Entscheidung
 1. Ausgabe
3. Vergleich/Entscheidung
 1. Ausgabe

```
password = check(entered_password)
if password == true:
    alert("welcome")
else:
    alert("error")
```


Der Algorithmus

„Eine Reihe an Schritten die durchgeführt werden müssen um ein Problem zu lösen!“

„Folge von mechanisch ausführbaren Anweisungen“

„Einen zu beschreitenden (vorher definierten) Weg, welcher zur Erreichung eines Zieles zu gehen ist.“

Die Semantik

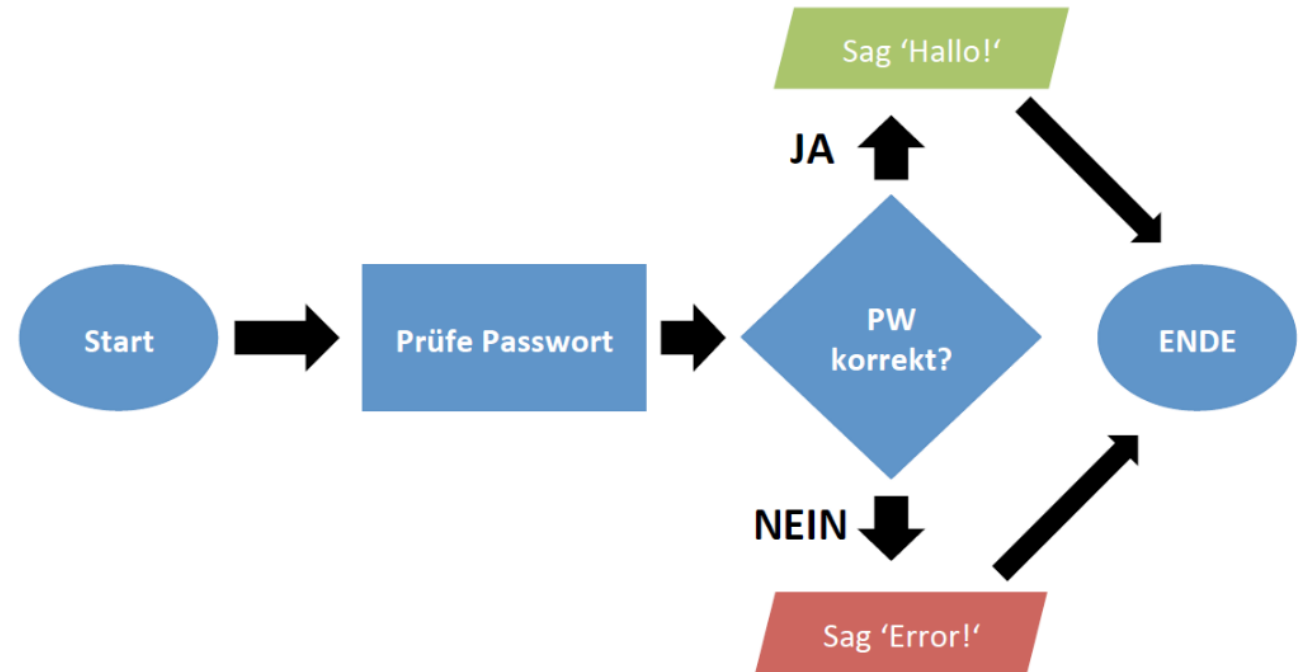
Bedeutungslehre

Ist die Bedeutung unseres Algorithmus

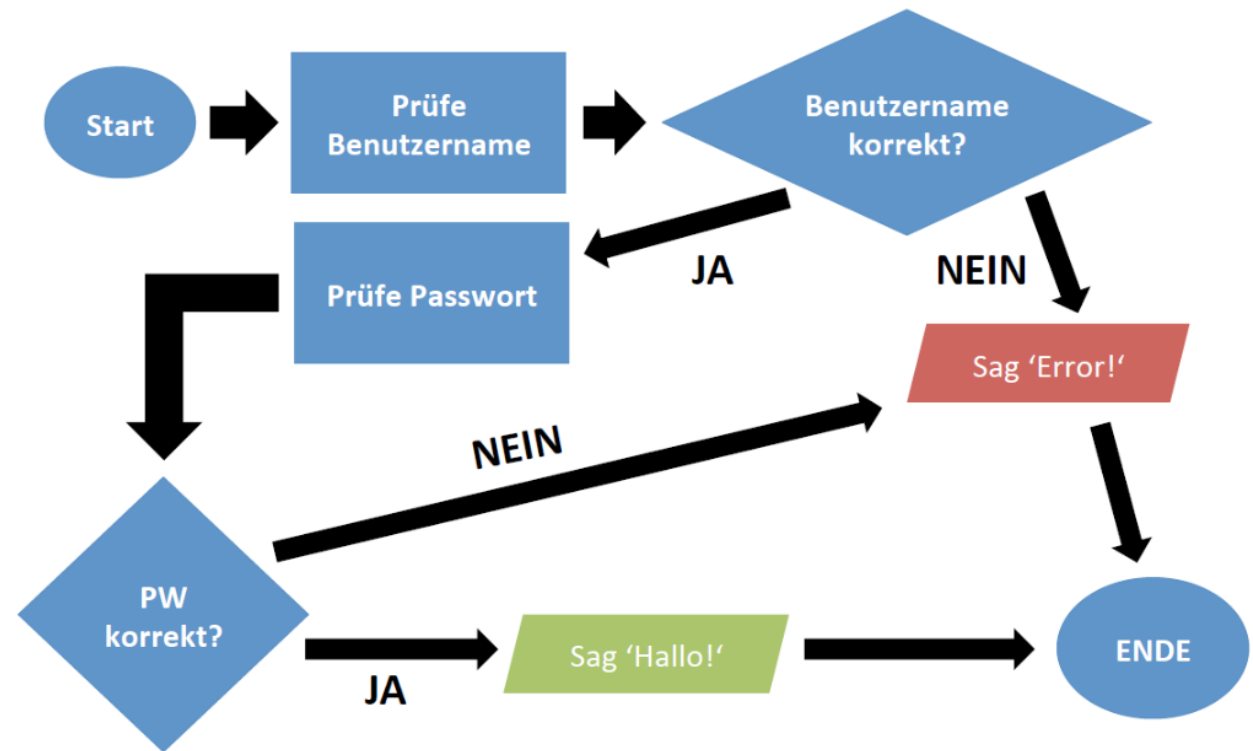
Sprich: Was der Algorithmus macht

Die Beschreibung des Verhaltens, das mit einer Interpretation eines syntaktisch korrekten Ausdrucks verbunden ist.

Die Semantik - Programmablauf



Die Semantik - Programmablauf



Die Syntax

Ist die Grammatik und Rechtschreibung einer Sprache.

Bezieht sich auf die Form und Struktur der Zeichen, ohne auf die Bedeutung einzugehen.

Legt den Aufbau und die Schreibweise der Sprache fest
Vorgaben der

The background features a series of concentric circles in light gray, some solid and some dashed, creating a ripple effect. A large red speech bubble is centered on the page, with a small triangular tail pointing downwards.

5 Tipps

5 Tipps

„Make everything as simple
as possible, but not simpler.“
-Albert Einstein

5 Tipps

„Ask ‚What would the user do?’ (You Are not the user)“
-Giles Colborne

5 Tipps

„Don't be afraid to break things“
-Mike Lewis

5 Tipps

„Reinvent the wheel often “
-Jason P Sage

5 Tipps

„Test precisely and concretely“
-Kevlin Henney

Praxis

Anhand von JavaScript (Basics)

Ausgaben

Direkte Ausgabe

```
document.write('Hallo Welt!');
```

Dialogfenster (Popup Boxes)!

```
alert('Hallo Welt! ');!
```

```
confirm('Press a button');!
```

```
prompt('Name eingeben', 'Mustermann');
```

Variablen

Bekommen von uns einen Namen.

Merken von Werten.

```
var MeineVariable;
```

```
MeineVariable = 42;
```

```
alert(MeineVariable);
```

Variablen

Werden solange im Arbeitsspeicher Gehalten bis das Programm beendet ist.

Flüchtiger Speicher.

In JS gibt es keine Möglichkeit Daten permanent zu speichern. Man benötigt noch weitere Hilfsmittel (zB.: AJAX, Cookies).

Variablennamen Regeln

- Variablennamen werden vom Programmierer vergeben.
Hierfür gibt es Regeln:
 - Keine reservierten Wörter verwenden
 - (zB.: var, function, alert)
 - Erste Stelle muss ein Buchstabe sein
 - Invalid
 - var 3x; var 4all;
 - Valid
 - var go4it; var free4all;
 - Dürfen mit \$ und _ beginnen
 - var _myVar;!
 - var \$slider;
 - Groß- und Kleinschreibung wird unterschieden
 - var world;!
 - var World;!
 - var WorlD;!
 - Sie sollen sinnvoll und gleich strukturiert sein!

Datentypen

Intenger / Floats Zahlen

2 Arten von numerischen Werten:

- Intenger (ganze Zahlen)
- Float (Fließkommazahlen)
- Number (Jede Zahl)

Beide können positiv oder negativ sein!

Intenger

Speichern Werte ohne Komma und Dezimalzahlen.

Float

Enthalten einen Dezimalpunkt (Kein Komma!)

Datentypen

Integer / Floats Zahlen

- **var** myNumber = 200; //Integer
- **var** _num = 105.20; //Float
- **var** a = 100;
- **var** b = 200.90;

Datentypen

Strings Zeichenketten

Müssen in Anführungszeichen gesetzt werden.
(Einfache oder Doppelte)

Anführungszeichen nicht mischen!!!

Kommt das gleiche Anführungszeichen im String vor kann man mit einem Backslash \ „escapen“.

In JS besteht zwischen ' und " kaum Unterschied.

Datentypen

Strings Zeichenketten

'Das ist ein String'

"Das ist auch ein String"

'So geht's aber nicht'

"Das auch nicht"

'So geht\'s aber wieder'

"So geht's auch"

Datentypen

Strings Zeichenketten

'Das ist ein String'

"Das ist auch ein String"

'So geht's aber nicht'

"Das auch nicht"

'So geht\'s aber wieder'

"So geht's auch"

Datentypen

Boolsche Werte
boolean bzw. bool

Ist ein logischer Wahrheitswert

Kann genau eine von zwei Informationen beinhalten!

wahr oder falsch!

1 oder 0!

true oder false!

yes oder no!

Datentypen

Boolsche Werte
boolean bzw. bool

Ist ein logischer Wahrheitswert

Kann genau eine von zwei Informationen beinhalten!

wahr oder falsch!

1 oder 0!

true oder false!

yes oder no!

Datentypen

null & undefined

null

Ist ein JS-Objekt zum Initialisieren und Löschen von Variablen. Variable enthält nicht länger einen Wert und der Speicher wird befreit.

undefined

Eine Variable, die nicht deklariert ist, aber Ausgegeben wird, hat keinen Wert. Daher undefined spricht nicht definiert.

Datentypen

Arrays
Listen

Sind dazu gedacht mehrere Werte in einer Variable zu speichern.

```
var ArrayName = new Array();
```

```
ArrayName[index] = Inhalt;
```

Datentypen

Arrays Listen

Sind dazu gedacht mehrere Werte in einer Variable zu speichern.

Indizierung beginnt bei Null

Datentypen können in Arrays gemischt sein

```
var ArrayName = new Array();
```

```
ArrayName[index] = Inhalt;
```

```
// Meine Farben
```

```
var farben = new Array();!
```

```
var farben[0] = 'rot';!
```

```
var farben[1] = 'gelb';!
```

```
var farben[2] = 'blau';!
```

```
var farben[3] = 'grün';!
```

```
var farben[4] = 'grün';!
```

Datentypen

Assoziative Arrays

Listen mit Namesverzeichnis

Man kann statt numerischen Werten auch Zeichenketten als Index verwenden.

```
var datum = new Array();
```

```
datum['tag'] = 30;
```

```
datum['monat'] = 10;
```

```
datum['jahr'] = 2012;
```

Datentypen

Mehrdimensionale Arrays

Man kann mehrere Dimensionen oder auch Ebenen erstellen.

```
var meinArray = new Array();  
meinArray[0] = new Array();  
meinArray[0][0] = 'Inhalt';  
meinArray[0][1] = 'Anderer Inhalt';
```

Zugriff:

```
alert(meinArray[0][1]);
```

Operatoren

(Zuweisen, Vergleichen, Rechnen,
(Logisch) Verbinden)

Operatoren

Zuweisungsoperator
(=)

Gleichheitszeichen

```
var SinnDesLebens = 42;
```

Operatoren

Vergleichsoperatoren

(==, ===,
!=, !==,
<, >,
<=, >=)

```
var SinnDesLebens = 42;
```

```
SinnDesLebens == 42      true
```

```
SinnDesLebens != 42      false
```

```
SinnDesLebens === "42"   false
```

```
SinnDesLebens !== "xyz"  true
```

```
SinnDesLebens > 42        false
```

```
SinnDesLebens < 42        false
```

```
SinnDesLebens >= 42       true
```

```
SinnDesLebens <= 42       true
```

Operatoren

Berechnungsoperatoren (+ - * / %)

```
var Zwei = 1+1;
```

```
var GarNix = 1-1;
```

```
var AuchNix = 81 / 3 - 27;
```

```
var WenigerAlsNix = 81 / (3 - 27);
```

```
var SinnDesLebens = 6 * 7;
```

```
var MachtAuchSinn = 84 / 2;
```

```
var x = 2012 % 4;
```

```
var y = 5 % 4
```


Operatoren

Berechnungsoperatoren (+ - * / %)

<code>var Zwei = 1+1;</code>	2
<code>var GarNix = 1-1;</code>	0
<code>var AuchNix = 81 / 3 - 27;</code>	0
<code>var WenigerAlsNix = 81 / (3 - 27);</code>	-3,857...
<code>var SinnDesLebens = 6 * 7;</code>	42
<code>var MachtAuchSinn = 84 / 2;</code>	42
<code>var x = 2012 % 4;</code>	0
<code>var y = 5 % 4</code>	1

Operatoren

Berechnungsoperatoren (+ - * / %)

var Zahl = 2; 2

Zahl += 3; 5

Zahl++; 6

Zahl -= 2; 4

Zahl--; 3

Zahl *= 4; 12

Zahl /= 3; 4

Zahl = Zahl + 3;

Zahl = Zahl + 1;

Zahl = Zahl - 2;

Zahl = Zahl - 1;

Zahl = Zahl * 4;

Zahl = Zahl / 3;