

Programmieren in C

Grundlagen

thoto

/dev/tal e.V.

16. März 2013 (Version 10. April 2013)

Agenda für Heute

- 1 zu den letzten Stunden ...
 - Nachträge
 - Wiederholung: Verzweigung
- 2 Funktionen
 - Was ist das?
 - Funktionsweise
 - sonstiges dazu

Variablennamen

- Groß- und Kleinschreibung muss beachtet werden
- Beginnt mit _ oder Buchstaben
- Besteht aus Zahlen, Buchstaben und _
- kein Schlüsselwort oder Funktion
- möglichst an Bedeutung gebunden.

printf()/scanf()

unsigned o/u/x/X

signed d/i

float e/E f/F g/G

short/long hh/h/l/l

man-Pages: Dokumentation

```
man
```

```
man 3 printf
```

```
man 2 open
```

Noch einmal: Verzweigungen

Verzweigung

```
if (foo==3)
    printf("foo ist 3");
else {
    printf("foo ist nicht drei 3");
    if (foo>=4)
        printf("foo ist groesser oder gleich 4");
    else if (foo==2)
        printf("foo ist 2");
    else
        printf("koennte foo 1 sein?");
}
```

Neu: Switch-Case!

Switch-Case

```
switch (foo) {  
    case 1:  
        printf("foo ist 1");  
        break;  
    case 2:  
        printf("foo ist 2");  
        break;  
    //...  
    default:  
        printf("foo ... unbekannt");  
}
```

Funktionen

Taschenrechner (I)

... switch-case und scanf() ...

Funktion vs. Methode

Methode: `void foo(parameter){ ... }`

Funktion: `typ foo(parameter){ ... }`

→ Das heißt: Funktion hat einen Typ!

⇒ Rückgabewert

Implementierung: Stack

... Bildchen! ...
siehe *stack.pdf*

Funktionsweise Funktionen & Methoden

... Bildchen! ...
siehe *stack.pdf*

Taschenrechner II

Was wir wollen:

- Funktionen!
- Switch-case!
- Floats!

Statische Variablen

- `static int foo;`
- persistent in der Funktion
- aber Aufpassen
- z.B. dass reservierter Speicher freigegeben wird