

# Programmieren - was ist das genau?

- Programmieren heisst Computerprogramme herstellen (von griechisch *programma* für Vorschrift).
- Ein Computerprogramm ist Teil der **Software** eines Computers. Als Software bezeichnet man die elektronischen, nicht physischen Informationen, die das Gerät zum Laufen bringen.
- Als **Hardware** bezeichnet man die sichtbaren, physischen Bestandteile eines Computers.

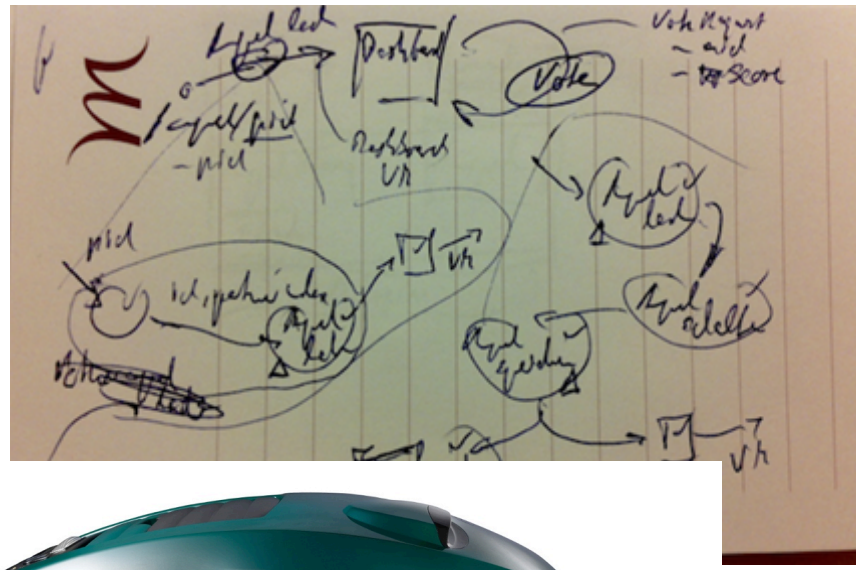


# Softwaredesign > Softwareentwurf



- Um komplizierte Programme zu erstellen, benötigt ein Programmierer einen entsprechenden Softwareentwurf:
- ◆ Zusammen mit dem Auftraggeber wird entschieden, welche Bedingungen das Programm erfüllen muss.

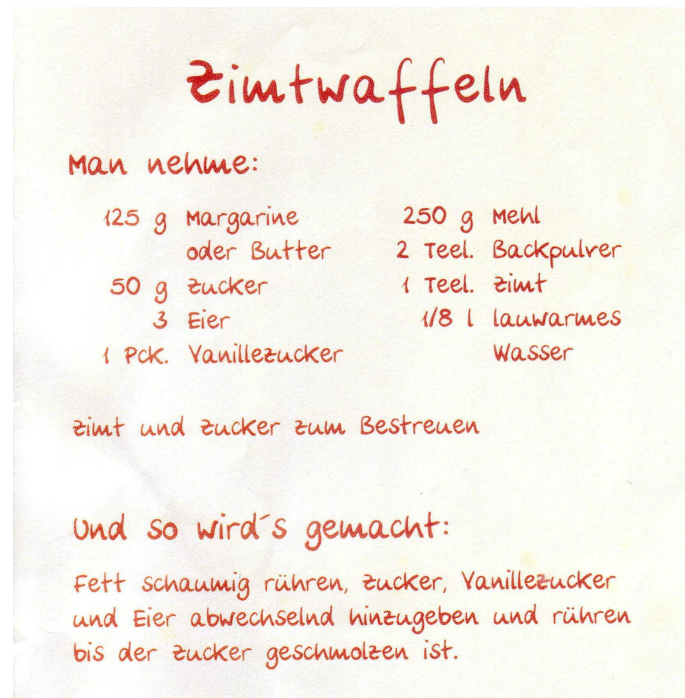
Beispiel Rasenroboter...



# Softwaredesign > Algorithmus

- ◆ Ein Programm benötigt eine Anleitung, wie es zum erwünschten Ziel kommen soll. Diese **Algorithmen** sind Bestandteile jedes Programmes.

Beispiel: Kuchenrezept



Während ein Mensch aufgrund seiner Vernunft und Intelligenz Algorithmen schnell erfassen kann, braucht es **für einen Computer ganz einfache und klare Anweisungen**, um zum Ziel zu kommen.

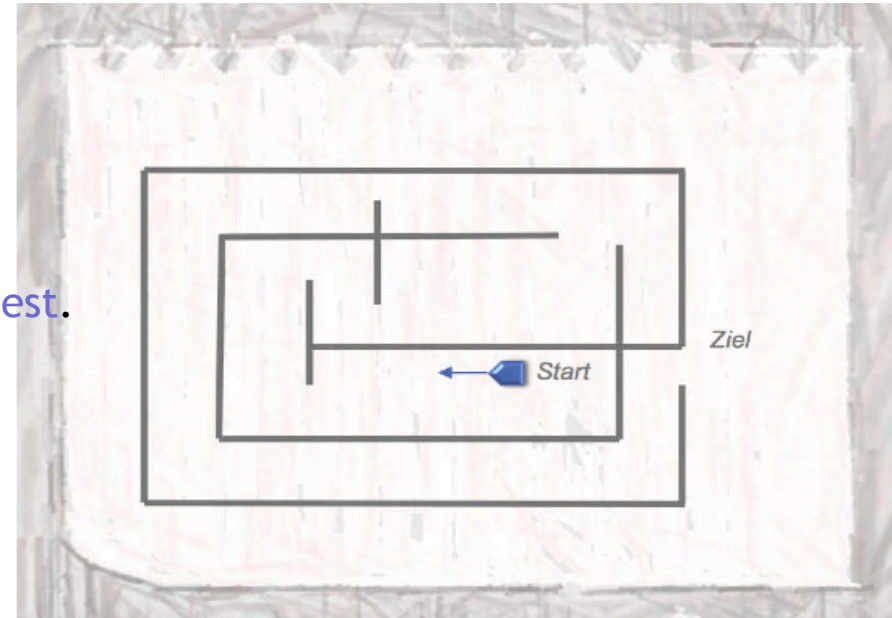


# Softwaredesign > Algorithmus

Ein Computer oder Roboter braucht klare Anweisungen.

Erkläre einem Blinden den Weg aus dem Labyrinth.

- Gehe geradeaus, bis du zu einer Wand kommst.
- Danach drehst du dich nach rechts.
- Folge der Wand so lange, bis du den Ausgang findest.





# Quellcode

- Mithilfe des Softwareentwurfs schreibt der Programmierer nun den **Quellcode**, einen für den Menschen lesbaren Text.
- Je nach Programmiersprache besteht der Quellcode aus unterschiedlichen Wörtern und Zeichenfolgen.
- Diese Quellcodes werden in einem speziellen Textprogramm geschrieben, der sogenannten **Entwicklungsumgebung**.



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
  <meta name="TITLE" content="Mein Weblog" />
  <meta name="KEYWORDS" content="Weblog, Programmieren" />
  <meta name="DESCRIPTION" content="Mein Weblog über Programmieren" />
  <link rel="stylesheet" type="text/css" href="style.css" />
  <script language="javascript" type="text/javascript" src="script.js" />
</head>
<body bgcolor="#ffffff" width="100%">
```

# Compiler

- Als **Compiler** wird ein Übersetzungsprogramm bezeichnet, das den Quellcode umschreibt und somit für den Computer lesbar macht („maschinenlesbar“).
- Die maschinenlesbaren Programme bestehen aus Folgen von Nullen und Einsen (Binärzahlen), für den Menschen nur schwer lesbar. Im Speicher des Computers abgelegt, können diese aber einzelne Befehle auslösen und sind so zum Beispiel verantwortlich für das Erscheinen eines Buchstabens auf dem Bildschirm.



# Binärcode



- Als **Binärcode** bezeichnet man eine Folge aus Nullen und Einsen, die für den Computer eine Information beinhaltet. Eine bestimmte Ziffernfolge löst einen eindeutigen Befehl aus, zum Beispiel die Wiedergabe des Buchstabens A.
- Anstelle von *0/1* könnte auch *wahr/falsch* oder eine andere 2er-Kombination verwendet werden.
- Eine Ziffer des Codes wird **Bit** genannt, eine Achterfolge nennt man **Byte**. Oft werden die Bit in 7er- oder 8er-Folgen notiert.
- Um die Systeme zu vereinheitlichen, haben die entsprechenden Folgen eine klar definierte Bedeutung.

Zum Beispiel:

A: 1000001

a: 1100001



# Überprüfen des Programmes

- Nach der Programmierung wird die Software auf allfällige Fehler überprüft.

## Debugging

Dies geschieht oft mit passenden Überprüfungsprogrammen. Die häufigsten Mängel sind fehlerhafte Eingaben in der Programmiersprache.

- Das Programm wird auf Robustheit geprüft, es soll absturzsicher sein.
- Updates müssen einfach hinzugefügt werden können.
- Es soll effizient arbeiten und nicht zu viel Speicherplatz einnehmen.

# Verschiedene Programmiersprachen



- Heute wird meist mit den sogenannten „höheren Programmiersprachen“ gearbeitet. Diese sind für den Menschen deutlich einfacher zu verstehen.
- Die Programmiersprachen werden häufig weiter entwickelt, und neue Versionen erscheinen.
- Heute oft verwendete Programmiersprachen sind *Java*, *Javascript*, *COBOL* (*Banken*), *PHP* (*Webseiten*), *Ruby* (*Webseiten*), *Swift* (*Mobile*)
- *Schweizer haben diverse Programmiersprachen erfunden. Darunter SCALA (damit ist bspw. «Twitter» programmiert)*

# C (und C++)

- alte Programmiersprache (1972 entstanden), immer noch weit verbreitet
- Stammprogramm von neueren Sprachen wie C++, Java und Objective-C
- Basis des Linux-Betriebssystems
- C++ als neuere, schnellere, aber auch schwer zu lernende Variante von C

```
#include <stdio.h>

int main(void)
{
    puts("Hallo Welt!");
}
```

# Java

- relativ einfache und sichere Programmiersprache
- vielseitig einsetzbar, läuft auf allen Betriebssystemen
- wurde zum Standard bei Internetseiten
- Java ist relativ langsam, der Wortschatz der Programmiersprache ist relativ beschränkt

```
class Hallo {  
    public static void main( String[] args ) {  
        System.out.println("Hallo Welt!");  
    }  
}
```



# Basic

- Programmiersprache für Microsoft Office
- viele verschiedene „Basic-Dialekte“
- für Anfänger leicht zu erlernen, klare Fehlermeldungen
- relativ geringe Verbreitung, da relativ langsam und auf Betriebssystem Windows beschränkt

```
PRINT "Hallo Welt!"
```

# Objektiv-C

- Programmiersprache für Anwendungen und Apps der Apple-Familie
- basiert auf der Programmiersprache C

```
#import <stdio.h>
int main()
{
    puts("Hallo Welt!");
    return 0;
}
```

# Kurze Geschichte des Programmierens

- 1842

Die britische Mathematikerin Ada Lovelace erstellt eine mathematische Vorschrift (Algorithmus) zur Ausrechnung von Bernoulli-Zahlen. Gedacht ist dieses „Programm“ für eine Rechenmaschine, die aber nie gebaut wurde.

- 1930

Das Lambda-Kalkül als formale Sprache wird eingeführt und heute noch in der theoretischen Informatik und in der Logik verwendet.

- 1959

Die amerikanische Mathematikerin Grace Hopper entwickelt den ersten Compiler. Die ersten höheren Programmiersprachen entstehen.

- 1964 / 1972

Die Programmiersprachen BASIC und C entstehen, um neuen Bedürfnissen der sich entwickelnden Computertechnik gerecht zu werden.

# Kurze Geschichte des Programmierens



## Schriftliche Lernkontrolle