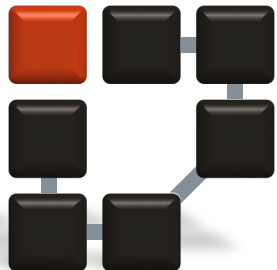


# Informatik 1 für Nebenfachstudierende Grundmodul

## Grundbegriffe

Kai-Steffen Hielscher  
Folienversion: 11. Oktober 2019



Informatik 7  
Rechnernetze und  
Kommunikationssysteme



FRIEDRICH-ALEXANDER  
UNIVERSITÄT  
ERLANGEN-NÜRNBERG

TECHNISCHE FAKULTÄT

# Inhaltsübersicht

- Kapitel 1 - Einführung und Übersicht
  - Was ist Informatik?
  - **Grundbegriffe**
  - Datendarstellung
  - Hardware von Computersystemen
  - Grundsoftware üblicher Computersysteme

# Wie arbeiten Computer?

- Computer berechnen Wettervorhersagen, steuern Raumfähren, spielen Schach, machen Musik, erzeugen unglaubliche Effekte in Kino-Filmen ...
- **Zentrale Frage** der Vorlesung: Wie arbeiten Computer?
  - Wie wird beispielsweise das Schachprogramm des Computers bedient?
  - Wie ist das Programm, das Schach spielt, aufgebaut?
  - Wie wird im Hauptspeicher des Rechners die Information über den Spielstand gespeichert und verändert?
  - Wie sind die Nullen und Einsen in den einzelnen Speicherzellen organisiert sind und wie werden sie verändert?
  - Wie beeinflussen elektrischen Signale die Transistoren und Widerstände, aus welchen sowohl die Speicherzellen als auch der Prozessor aufgebaut sind?
- **Zwei Aspekte** der Informatik:
  - Wissenschaft von der systematischen **Informationsverarbeitung**
  - **Anwendung:** Elektronische Datenverarbeitung (EDV)

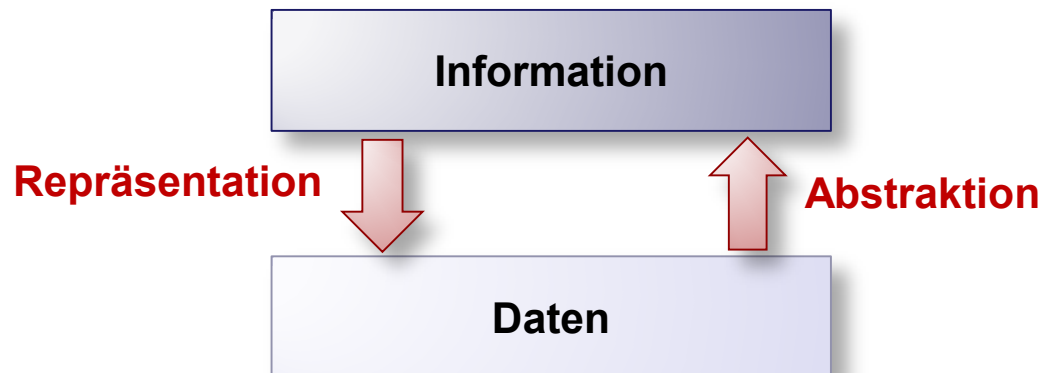
# Daten und Informationen

## ■ Daten:

- „Zeichen [...], die zum Zweck der Verarbeitung Informationen auf Grund bekannter oder unterstellter Abmachungen darstellen“ (DIN 44300)
- EVA-Prinzip der Datenverarbeitung: Eingabe - Verarbeitung - Ausgabe

## ■ Information:

- im weiteren Sinne „Reduktion von Ungewissheit“
- im engeren Sinne „Reduktion von Ungewissheit aufgrund von Kommunikationsvorgängen“ (Ergebnis eines Kognitionsprozesses)
- **Information = Daten + Bedeutung**



# Bits

- Ein **Bit** ist die kleinstmögliche Einheit der Information.
- Ein Bit ist die Informationsmenge in einer Antwort auf eine Frage, die zwei Möglichkeiten zulässt:
  - ja oder nein,
  - wahr oder falsch,
  - schwarz oder weiß,
  - ...
- Zu einer solchen Frage lässt sich immer eine **Codierung** der Antwort festlegen. Da es zwei mögliche Antworten gibt, reicht ein Code mit zwei Zeichen. Man benutzt dazu meist die Zeichen
  - 0 (nein, falsch, weiß, ...) und
  - 1 (ja, wahr, schwarz, ...).
- **Technische Realisierungen** sind z.B.
  - elektrische Ladungen: 0 = ungeladen, 1 = geladen
  - elektrische Spannungen: 0 = 0 Volt, 1 = 5 Volt
  - Magnetisierungen: 0 = unmagnetisiert, 1 = magnetisiert

# Bitfolgen

- Lässt eine Frage mehrere Antworten zu, so enthält die Beantwortung der Frage mehr als ein Bit an Information.
- **Beispiel:**
  - Die Frage, aus welcher Himmelsrichtung, Nord, Süd, Ost oder West, der Wind weht, lässt vier mögliche Antworten zu.
  - Der Informationsgehalt in der Beantwortung der Frage ist aber nur zwei Bit:
    - 1. Weht der Wind aus einer der Richtungen Nord oder Ost (ja/nein)?
    - 2. Weht der Wind aus einer der Richtungen Ost oder West (ja/nein)?
  - Antwort ja auf die erste Frage und nein auf die zweite Frage lässt sich durch die Bitfolge 10 repräsentieren, d.h.  
00 ↔ Süd      01 ↔ West      10 ↔ Nord      11 ↔ Ost
  - Offensichtlich gibt es genau vier mögliche Folgen von zwei Bit.
  - Mit zwei Bit können wir also Fragen beantworten, die vier mögliche Antworten zulassen.
- **Es gibt genau  $2^N$  mögliche Bitfolgen der Länge N.**

# Nibble und Hexadezimalzahlen

- Rechner können viel besser als Menschen mit Bitfolgen umgehen.
- Besser ist es lange Bitfolgen in 4-Bit-Gruppen, **Nibble** genannt, zusammenzufassen.
- Die Bitfolge
  - 01001000011000010110110001101100
- liefert dann die **Nibblefolge**
  - 0100 1000 0110 0001 0110 1100 0110 1100 .
- 4-Bit-Gruppen lassen sich auch als **Hexadezimalziffer** darstellen:

0000 = 0	0100 = 4	1000 = 8	1100 = C
0001 = 1	0101 = 5	1001 = 9	1101 = D
0010 = 2	0110 = 6	1010 = A	1110 = E
0011 = 3	0111 = 7	1011 = B	1111 = F

- Damit lässt sich die obige Bitfolge kompakter als **Hexadezimalzahl** 48616C6C schreiben.

# Bytes

- Wenn ein Rechner Daten nach dem EVA-Prinzip verarbeitet operiert er immer mit Gruppen von 8, 16, 32 oder 64 Bits.
- Deswegen spricht man von 8-Bit-/16-Bit-/32-Bit- oder 64-Bit-Rechnern.
- Meist sind es jedoch Mischformen, z.B. Rechner, die intern 32-Bitfolgen verarbeiten, aber Folgen zu 64 Bits lesen oder schreiben.
- Immer jedoch ist die Länge einer Bitfolge ein Vielfaches von 8.
- Eine Gruppe von 8 Bit heißt Byte, abgekürzt 1B, und lässt sich durch zwei Nibble oder zwei Hexadezimalziffern darstellen.
- Abkürzungen
  - In der Informatik werden die Zahlen 2, 8, 16, 1024 und andere Vielfache der Zahl 2 besonders häufig verwendet, z.B.
    - $8 = 2 \cdot 2 \cdot 2 = 2^3$
    - $1024 = 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2 = 2^{10}$



# Abkürzungen

- Als Abkürzungen für Speichergrößen, Verarbeitungsleistung, Kommunikationsbandbreite verwendet man
  - **1 k** =  $1024 = 2^{10}$  (k = **Kilo**)
  - **1 M** =  $1024 \cdot 1024 = 2^{20}$  (M = **Mega**)
  - **1 G** =  $1024 \cdot 1024 \cdot 1024 = 2^{30}$  (G = **Giga**)
  - **1 T** =  $1024 \cdot 1024 \cdot 1024 \cdot 1024 = 2^{40}$  (T = **Tera**)
  - **1 P** =  $1024 \cdot 1024 \cdot 1024 \cdot 1024 \cdot 1024 = 2^{50}$  (P = **Peta**)
  - **1 E** =  $1024 \cdot 1024 \cdot 1024 \cdot 1024 \cdot 1024 \cdot 1024 = 2^{60}$  (E = **Exa**)
- Für Längen- und Zeitmaße werden die üblichen Einheiten zur Basis 10 benutzt:
  - **1 m** =  $1/1000 = 10^{-3}$  (m = **Milli**)
    - ◆ z.B. *1 mm = Millimeter, 1 ms = 1 Millisekunde*
  - **1 μ** =  $1/1000000 = 10^{-6}$  (μ = **Mikro**)
  - **1 n** =  $1/1000000000 = 10^{-9}$  (n = **Nano**)
  - **1 p** = ... =  $10^{-12}$  (p = **Pico**)
  - **1 f** = ... =  $10^{-15}$  (f = **Femto**)
- Für Längenangabe findet man häufig  
**1" = 1 in = 1 inch = 1 Zoll = 2,54 cm**