

EINI

LogWing/WiMa/MP

**Einführung in die Informatik für
Naturwissenschaftler und Ingenieure**

Vorlesung 2 SWS WS 18/19


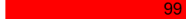
















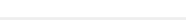
Dr. Lars Hildebrand
Fakultät für Informatik – Technische Universität Dortmund
lars.hildebrand@tu-dortmund.de
<http://ls14-www.cs.tu-dortmund.de>


























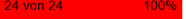



Praktikum & Übung

EINI LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

Übungszeitslot	# Übungen	Platzangebot (Summe)	Beliebtheit (# Prio 1)
Mo : 08:30 - 11:00	2	48	 17
Mo : 11:00 - 13:30	1	24	 99
Mo : 13:30 - 16:00	1	24	 38
Mo : 16:00 - 18:30	2	48	 33
Di : 11:00 - 13:30	1	24	 53
Di : 13:30 - 16:00	1	24	 43
Di : 16:00 - 18:30	2	48	 15
Di : 08:30 - 11:00	2	48	 17
Mi : 11:00 - 13:30	2	48	 25
Mi : 13:30 - 16:00	2	48	 36
Mi : 16:00 - 18:30	2	48	 30
Mi : 08:30 - 11:00	2	48	 67
Do : 13:30 - 16:00	1	24	 23
Do : 16:00 - 18:30	1	24	 26
Do : 11:00 - 13:30	1	24	 148
Fr : 16:00 - 18:30	2	48	 18
Fr : 11:00 - 13:30	2	48	 32
Fr : 13:30 - 16:00	2	48	 24
Interessenten gesamt:			 537 / 696 = 77%

Montag 08:30 - 11:00	Gruppe 1	Hildebrand	 13 von 24	54,2%
Montag 08:30 - 11:00	Gruppe 2	Hildebrand	 2 von 24	8,3%
Montag 11:00 - 13:30	Gruppe 3	Hildebrand	 24 von 24	100%
Montag 13:30 - 16:00	Gruppe 4	Hildebrand	 24 von 24	100%
Montag 16:00 - 18:30	Gruppe 5	Hildebrand	 24 von 24	100%
Montag 16:00 - 18:30	Gruppe 6	Hildebrand	 24 von 24	100%
Dienstag 08:30 - 11:00	Gruppe 7	Hildebrand	 23 von 24	95,8%
Dienstag 08:30 - 11:00	Gruppe 8	Hildebrand	 1 von 24	4,2%
Dienstag 11:00 - 13:30	Gruppe 9	Hildebrand	 24 von 24	100%
Dienstag 13:30 - 16:00	Gruppe 10	Hildebrand	 24 von 24	100%
Dienstag 16:00 - 18:30	Gruppe 11	Hildebrand	 1 von 24	4,2%
Dienstag 16:00 - 18:30	Gruppe 12	Hildebrand	 16 von 24	66,7%
Mittwoch 08:30 - 11:00	Gruppe 13	Hildebrand	 24 von 24	100%
Mittwoch 08:30 - 11:00	Gruppe 14	Hildebrand	 24 von 24	100%
Mittwoch 11:00 - 13:30	Gruppe 15	Hildebrand	 2 von 24	8,3%
Mittwoch 11:00 - 13:30	Gruppe 16	Hildebrand	 23 von 24	95,8%
Mittwoch 13:30 - 16:00	Gruppe 17	Hildebrand	 23 von 24	95,8%
Mittwoch 13:30 - 16:00	Gruppe 18	Hildebrand	 23 von 24	95,8%
Mittwoch 16:00 - 18:30	Gruppe 19	Hildebrand	 24 von 24	100%
Mittwoch 16:00 - 18:30	Gruppe 20	Hildebrand	 24 von 24	100%
Donnerstag 11:00 - 13:30	Gruppe 21	Hildebrand	 24 von 24	100%
Donnerstag 13:30 - 16:00	Gruppe 22	Hildebrand	 24 von 24	100%
Donnerstag 16:00 - 18:30	Gruppe 23	Hildebrand	 24 von 24	100%
Freitag 11:00 - 13:30	Gruppe 24	Hildebrand	 21 von 24	87,5%
Freitag 11:00 - 13:30	Gruppe 25	Hildebrand	 12 von 24	50%
Freitag 13:30 - 16:00	Gruppe 26	Hildebrand	 24 von 24	100%
Freitag 13:30 - 16:00	Gruppe 27	Hildebrand	 24 von 24	100%
Freitag 16:00 - 18:30	Gruppe 28	Hildebrand	 5 von 24	20,8%
Freitag 16:00 - 18:30	Gruppe 29	Hildebrand	 12 von 24	50%

In diesem Kapitel:

- Prolog
- Rechensysteme
- Datendarstellung

▶ Rechensysteme

- ▶ Was macht ein Rechensystem aus?
 - Hardware
 - Verwendung

▶ Datendarstellung

- ▶ Grundbegriffe
- ▶ Texte, Programme, Grafiken
- ▶ Logische/boolesche Werte
- ▶ Natürliche Zahlen, ganze Zahlen, Gleitpunktzahlen
- ▶ Daten vs. Informationen

- Prolog
- Rechensysteme
- Datendarstellung

Was macht ein Rechensystem aus?

- ▶ **Physikalische Komponenten (Hardware):**
 - ▶ erbringen spezielle Leistungen
 - ▶ machen Funktionen verfügbar

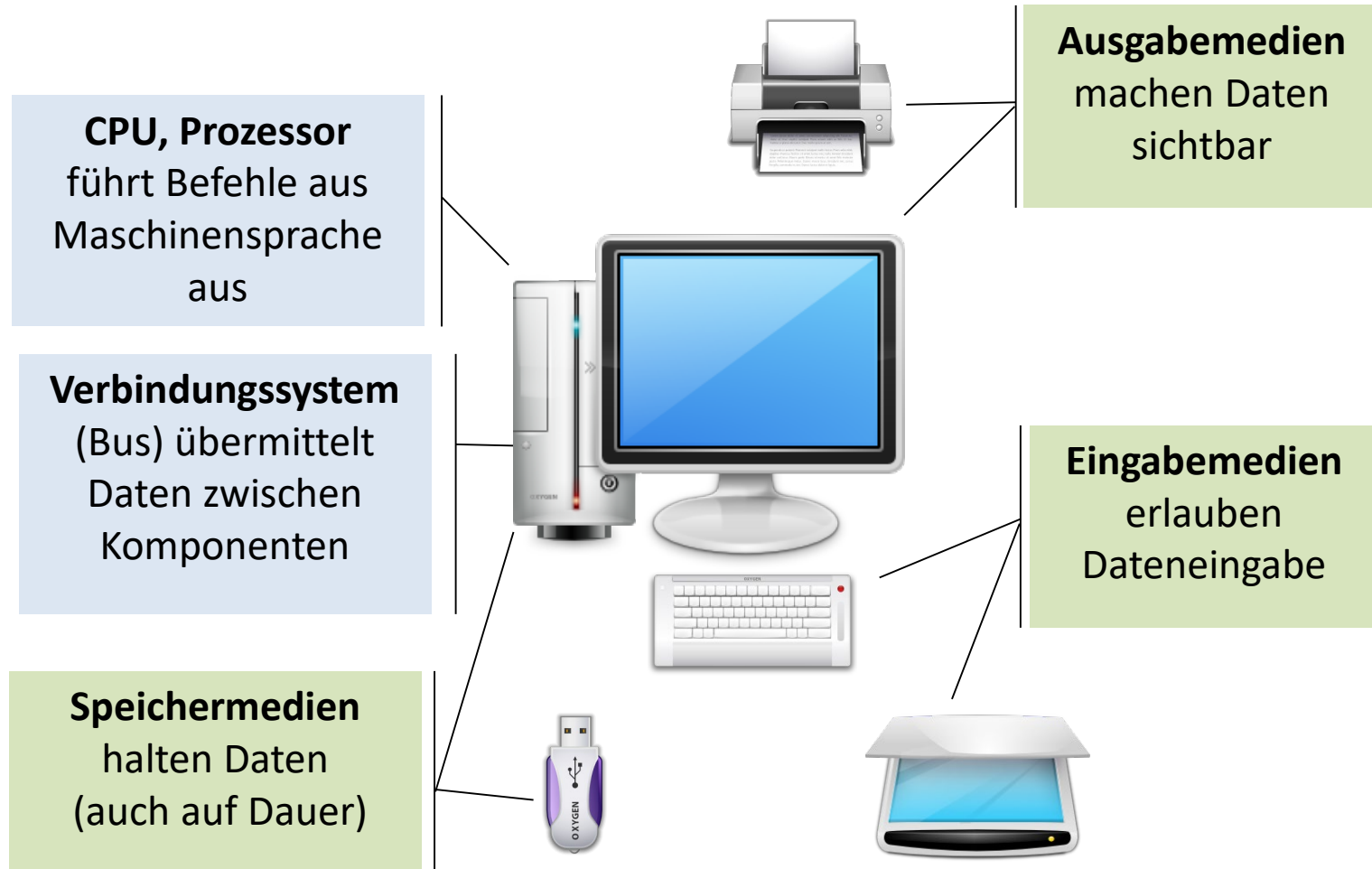
EINI LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstel-
lung

In diesem Kapitel:

- Prolog
- **Rechensysteme**
- Datendarstellung



Was macht ein Rechensystem aus?

Verwendung

- ▶ Privatanwender:
 - ▶ Textverarbeitung
 - ▶ Tabellenkalkulation
 - ▶ E-Mail
 - ▶ im Internet surfen, Informationsbeschaffung, ...

- ▶ Firmen:
 - ▶ wie Privatanwender, außerdem:
 - ▶ Verwaltung von Firmendaten und Arbeitsvorgängen, Produktionsplanung und -steuerung, Buchhaltung, ...
 - Datenbankapplikationen
 - Enterprise Ressource Systeme wie SAP R/3
 - ▶ Steuerung automatisierter Fertigungsanlagen

- Warum geht das?

EINI LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstel-
lung

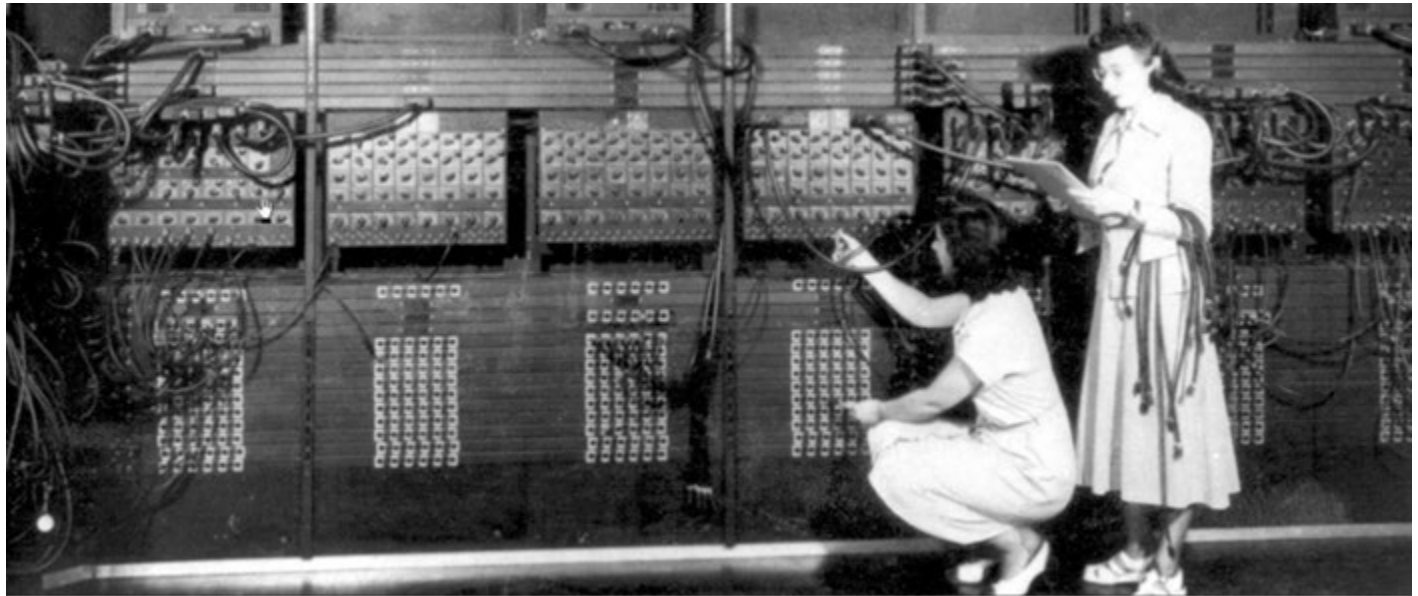
In diesem Kapitel:

- Prolog
- **Rechensysteme**
- Datendarstellung

Was macht ein Rechensystem aus?

Frei programmierbare Rechner!

- ▶ Rechensysteme sind flexibel einsetzbar.
- ▶ Ihre Fähigkeiten lassen sich an die jeweiligen Anforderungen anpassen durch
 - ▶ Programmierung
 - ▶ Softwareentwicklung



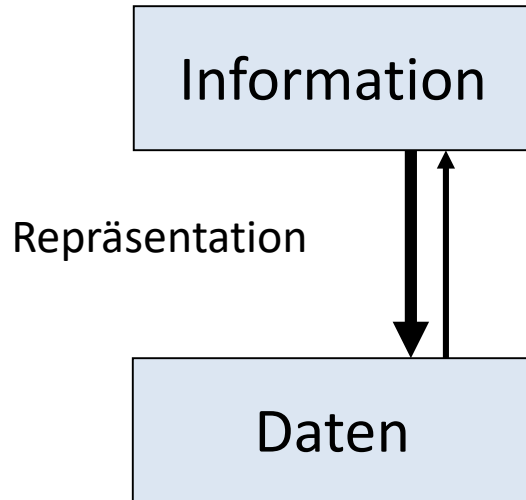
Quelle: TomsHardware.com

- Prolog
- **Rechensysteme**
- Datendarstellung

Wozu wird ein Rechensystem genutzt?

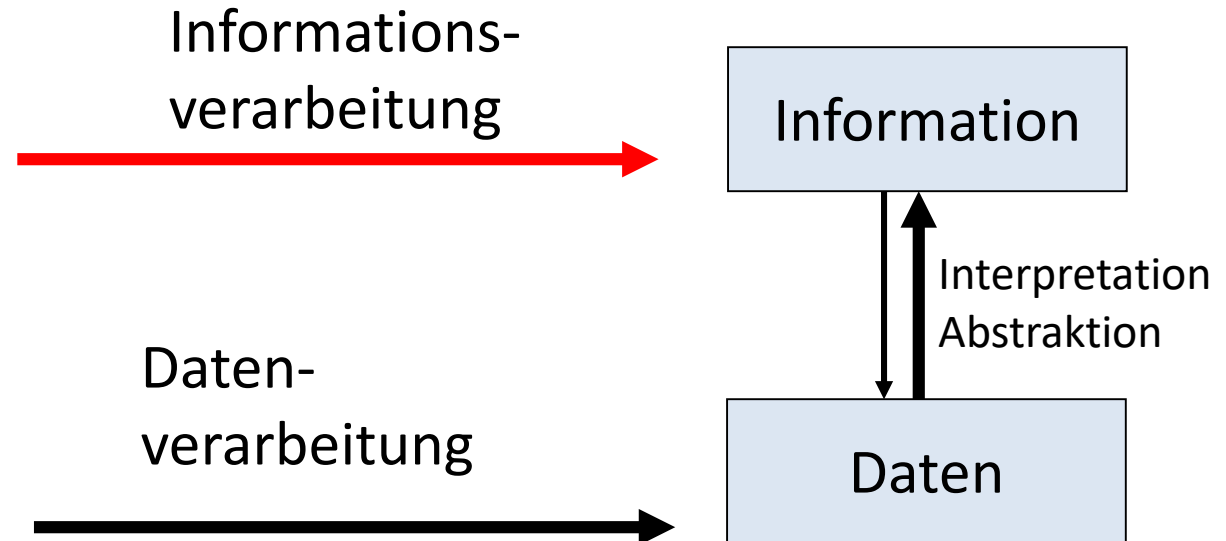
Wunsch:

Informationsverarbeitung



Wirklichkeit:

Datenverarbeitung



Anmerkung 1:

Die grundsätzliche Crux der Informatik besteht darin, dass ein System ohne eigenes Verstehen und ohne eigene Erkenntnis geschaffen wird, das dennoch ein sinnvolles Verhalten zeigen soll.

Anmerkung 2:

Repräsentation von Informationen durch Daten kann auf unterschiedlichen Ebenen erfolgen. Wir werden uns direkt im Anschluss mit der elementarsten Ebene befassen: die Werte 0 und 1.



Artikel im EINI-Wiki:

→ **Programmierung**

EINI LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstel-
lung

In diesem Kapitel:

- Prolog
- **Rechensysteme**
- Datendarstellung

Zwischenstand

- ✓ Was ist Informatik ?
- ✓ Was macht ein Rechensystem aus?
- Frage: Wie werden Daten in einem Rechner dargestellt?
 - ▶ Buchstaben, Zeichenketten, Texte, ...
 - ▶ Grafiken
 - ▶ Algebren
 - Boolesche Algebra: Operationen AND, OR, NOT
 - Natürliche Zahlen, ganze Zahlen, reellwertige Zahlen: Operationen Addition, Division, Modulo, ...
 - ❖ Achtung: Genauigkeit der Darstellung und damit auch von Berechnungen ist begrenzt!
Wertebereiche für Zahlen sind beschränkt!
- ▶ Ziel: Mit dem **Entwurf von Algorithmen und Programmen**, der Programmierung von Rechensystemen und zugehörigen Programmiersprachen befassen.

EINI LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Darstellung elementarer Daten

Gliederung

- ▶ Grundbegriffe der Datendarstellung
- ▶ Datendarstellung im Überblick
- ▶ Texte
- ▶ Programme, Grafiken
- ▶ Logische/Boolesche Werte
- ▶ Natürliche Zahlen
 - ▶ Umrechnung: Dezimal in Binär
- ▶ Ganze Zahlen
 - ▶ Zweierkomplement
 - ▶ Überprüfung der Zulässigkeit von Resultaten
- ▶ Gleitpunktzahlen
- ▶ Daten vs. Informationen

EINI LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Grundbegriffe der Datendarstellung

Bit

- ▶ kleinstmögliche Einheit der Information(sdarstellung)
- ▶ Einheitenzeichen: bit
- ▶ erlaubt Antwort auf eine Frage mit nur zwei Antwortmöglichkeiten
 - ▶ z.B. {ja,nein}, {wahr,falsch}, {schwarz,weiß}, {links,rechts},
 - ▶ meist durch {0,1} codiert
- ▶ technische Umsetzung durch
 - ▶ Ladungen: 0 = ungeladen, 1 = geladen
 - ▶ Spannungen: 0 = 0 Volt, 1 = 5 Volt
 - ▶ Magnetisierung: 0 = unmagnetisiert, 1 = magnetisiert

Wir gehen im folgenden von {0,1} als verfügbar aus.

Grundbegriffe der Datendarstellung

Bitfolgen

- ▶ basieren auf Sequenzen $\{0,1\}^n$, $n \in \mathbb{N}$
- ▶ erlauben Codierung von Mengen, z.B.:

000 = Süd	001 = West
010 = Nord	011 = Ost
100 = Südost	101 = Nordwest
110 = Nordost	111 = Südwest

Es gibt genau 2^n unterschiedliche Bitfolgen der Länge n .

Hexadezimalzahlen

- ▶ Bitfolgen werden schnell unübersichtlich, daher: Blöcke aus 4 Bits als „Ziffer“:

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

- entspricht Zahlendarstellung zur Basis 16.

Grundbegriffe für Datendarstellung

Byte

- ▶ Rechner behandeln keine einzelnen Bits: kleinste betrachtete Bitfolge ist das Byte = 8 Bits
- ▶ gröbere Granularität kommt heutzutage nur als Vielfaches von 8 Bit vor, z.B. 16 Bit-, 32 Bit-, 64-Bit-Rechner
- ▶ 1 Byte erlaubt $2^8=256$ Werte zu unterscheiden, z.B. zur Codierung von Buchstaben
- ▶ übliche Abkürzungen: B = Byte, b = Bit

$$1K = 1024 = 2^{10} \text{ (K = kilo)}$$

$$1M = 1024 \cdot 1024 = 2^{20} \text{ (M = mega)}$$

$$1G = 1024 \cdot 1024 \cdot 1024 = 2^{30} \text{ (G = giga)}$$

$$1T = 1024 \cdot 1024 \cdot 1024 \cdot 1024 = 2^{40} \text{ (T = tera)}$$

$$1P = 1024 \cdot 1024 \cdot 1024 \cdot 1024 \cdot 1024 = 2^{50} \text{ (P = Peta)}$$

$$1E = 1024 \cdot 1024 \cdot 1024 \cdot 1024 \cdot 1024 \cdot 1024 = 2^{60} \text{ (E = exa)}$$

Für Längen/Zeiten:

$$1m = 10^{-3} \text{ (m = milli)}$$

$$1\mu = 10^{-6} \text{ (\mu = mikro)}$$

$$1n = 10^{-9} \text{ (n = nano)}$$

$$1p = 10^{-12} \text{ (p = pico)}$$

$$1f = 10^{-15} \text{ (f = femto)}$$

Datendarstellung: Komplexe Datentypen

- ▶ Texte
- ▶ Programme
- ▶ Grafiken (Bilder)
- ▶ Zahlen, Algebren
 - ▶ Boolesche Algebra, Wahrheitswerte
 - ▶ Natürliche Zahlen
 - ▶ Ganze Zahlen
 - ▶ Reellwertige Zahlen
- ▶ Anmerkung:
 - ▶ Bei der Darstellung von Zahlen werden wir erkennen, dass nicht alle aus der Mathematik vertrauten Eigenschaften von Zahlen auf einem Rechner erhalten bleiben.

EINI LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Datendarstellung: Texte

- ▶ **Texte** = Zeichenfolgen aus Buchstaben und Satzzeichen
 - ▶ Codierung jedes Buchstabens/Zeichens durch Bitfolge
- ▶ **ASCII** = *American Standard Code for Information Interchange*
 - ▶ 7 Bit (= max. 128 Zeichen), Tabelle mit Nummerierung aller Zeichen
 - ▶ z.B. „a“: Nummer 97, „A“: Nummer 65, „?“: Nummer 63
 - ▶ Klein- und Großbuchstaben nach Alphabet durchnummeriert
 - ▶ übliche Erweiterung auf PCs: 8 Bit (weitere Sonderzeichen, z.B. Umlaute)
 - ▶ Erweiterung in Europa: Latin-1 (nach Norm ISO 8859-1)
- ▶ **Unicode** (z.B. von Java verwendet)
 - ▶ 16 Bit (= max. 65536 Zeichen)
 - ▶ siehe <http://www.unicode.org>
(Quick Links → Code Charts: nach Sprachen sortiert)
 - ▶ als Obermenge weltweit geläufiger Zeichensätze

Datendarstellung: Texte

► Unicode-Tabelle: Latin 1

0020	0	@	P	`	p		°	À	Ð	à	ð	
0021	!	1	A	Q	a	q	±	Á	Ñ	á	ñ	
0022	"	2	B	R	b	r	²	Â	Ò	â	ò	
0023	#	3	C	S	c	s	³	Ã	Ó	ã	ó	
0024	\$	4	D	T	d	t	¼	Ä	Ô	ä	ô	
0025	%	5	E	U	e	u	¥	Å	Õ	å	õ	
0026	&	6	F	V	f	v	¦	Æ	Ö	æ	ö	
0027	'	7	G	W	g	w	§	Ç	×	ç	÷	
0028	(8	H	X	h	x	¨	È	Ø	è	ø	
0029)	9	I	Y	i	y	©	É	Ù	é	ù	
002A	*	:	J	Z	j	z	ª	Ê	Ú	ê	ú	
002B	+	;	K	[k	{	«	Ë	Û	ë	û	
002C	,	<	L	\	l		¬	¼	Ì	Ü	ì	ü
002D	-	=	M]	m	}	-	½	Í	Ý	í	ý
002E	.	>	N	^	n	~	®	¾	Î	Þ	î	þ
002F	/	?	O	_	o		™	¿	Ï	ß	ï	ÿ

EINI LogWing /
WiMa

Kapitel 1

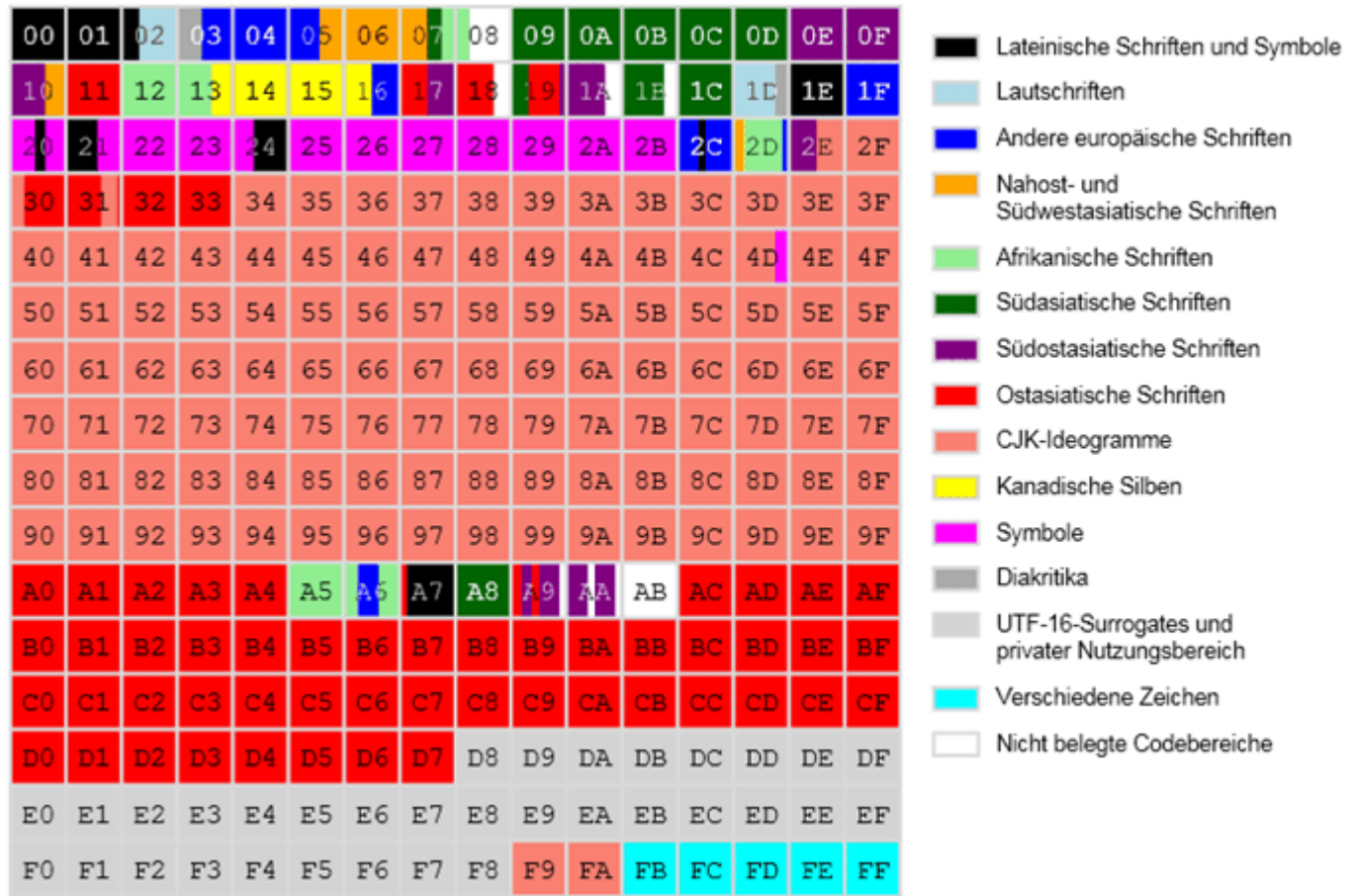
Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Datendarstellung: Texte

- ▶ Grundlegender mehrsprachiger Codebereich der Unicode-Tabelle



EINI LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

▶ Programme

- ▶ Ein Programm wird zunächst als Quelltext erzeugt und wie normaler Text repräsentiert.
- ▶ Übersetzungsprogramme (Compiler) erzeugen daraus Programmcode in Maschinsprache.
- ▶ Auf jeder Abstraktionsebene müssen alle Anteile eines Programms durch Bitfolgen codiert werden.

```
public class HelloWorld {  
    /**  
     * @param args  
     */  
    public static void main(String[] args) {  
        // TODO  
        System.out.println("Hello World");  
    }  
}
```

```
// Bytecode stream: 03 3b 84 00 01 1a 05 68 3b a7 ff f9  
// Disassembly:  
iconst_0      // 03  
istore_0      // 3b  
iinc 0, 1     // 84 00 01  
iload_0       // 1a  
iconst_2      // 05  
imul          // 68 3b a7 ff f9  
istore_0      // 03  
goto -7       // 01 1a 05 68 3b a7 ff f9  
:1000000075812F12019912025212025890004D125E  
:10001000027B90005B120285750200744D12022D66  
:100020001200691203271200E304F9D8FED9FC7408  
:100030000E12022DE59012037A20B304050280020D  
:1000400015028502A030B2D37502FF80CE4449501C  
:100050003820503128686578293A00414455776541
```

Datendarstellung: Grafiken

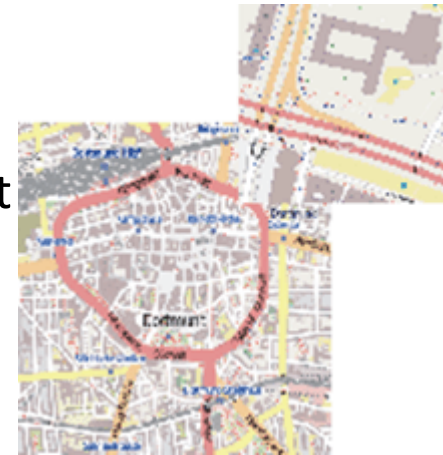
▶ Rastergrafik

- ▶ Grafik wird als eine Folge einzelner Rasterpunkte dargestellt.
- ▶ Einzelner Rasterpunkt durch 1 Bit oder 1+ Bytes (Farbe) codiert.



▶ Vektorgrafik

- ▶ Grafik wird aus Linien zusammengesetzt, für die Anfangs- / Endpunkte /etc. codiert werden müssen.



▶ Boolesche Algebra

- ▶ **Trägermenge** = {false,true} (oft auch als {0,1}) mit
- ▶ **Operationen** (z.B.):
 - Und-Verknüpfung: AND,
 - Oder-Verknüpfung: OR,
 - Negation: NOT,
 - Exklusives Oder: XOR

▶ Darstellung in Rechnern

- ▶ erfordert meist 1 Byte (mindestens) als kleinste behandelbare Dateneinheit
 - 1 Bit wäre im Prinzip ausreichend, jedoch ist im Rechner ein einzelnes Bit nur als Element innerhalb eines Bytes und über das zugehörige Byte adressierbar.
 - Bitfelder dagegen lassen sich mit Platzverbrauch 1 Bit je Boolescher Variable verwalten, wobei das Bitfeld insgesamt jedoch eine Größe in ganzen Bytes haben muss.

Darstellung von natürlichen Zahlen

Satz: Jede natürliche Zahl n besitzt zur Basis $p \geq 2$ ($p \in \mathbb{N}$) eine eindeutige m -stellige p -adische Darstellung der Form

$$n = \sum_{i=0}^{m-1} \alpha_i \cdot p^i \quad \text{mit } 0 \leq \alpha_i < p \text{ und } m \geq \log_p n$$

Bemerkungen:

- ▶ Positionales Zahlensystem
- ▶ Ziffern dürfen Basiswert p nicht erreichen!
- ▶ Für uns üblich: Dezimalzahlen $p=10$ und m nach Bedarf

$$2003_{10} = 2 \cdot 10^3 + 0 \cdot 10^2 + 0 \cdot 10^1 + 3 \cdot 10^0$$

Darstellung von natürlichen Zahlen: Beispiel

Bemerkungen:

- ▶ Ziffern dürfen Basiswert p nicht erreichen!
- ▶ Im Rechner üblich: Binärzahlen $p=2$, $m=16$, 32 oder 64

$$1110_2 = 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 8_{10} + 4_{10} + 2_{10} = 14_{10}$$

- ▶ Gelegentlich zur Dokumentation von Zahlenwerten/Adressen:
 - ▶ Hexadezimal: $p=16$
 - ▶ Oktal: $p=8$

- Prolog
- Rechensysteme
- **Datendarstellung**

Umrechnung: Dezimal in Binär

Umrechnung von Dezimalzahlen in Binärzahlen durch ganzzahlige Division und Modulo-Operation, d.h.:

$$n = \sum_{i=0}^{m-1} \alpha_i \cdot p^i = p \cdot \left(\sum_{i=1}^{m-1} \alpha_i \cdot p^{i-1} \right) + \alpha_0$$

$$\alpha_0 = n \bmod p \quad \sum_{i=1}^{m-1} \alpha_i \cdot p^{i-1} = n \div p$$

$$\alpha_i = (n \div p^i) \bmod p$$

also **fortgesetztes Dividieren**.

Der Rest r (mathematisch formal: modulo) liefert die Ziffernfolge:

z.B. $4711_{10} = 10010011001\mathbf{1}_2$

$4711 / 2 = 2355$ mit Rest 1 -> „**rechtste**“ 1 in der Binärdarstellung

$2355 / 2 = 1177$ mit Rest 1 -> „**vorletzte**“ 1 in der Binärdarstellung

Anmerkung: Geläufige Rechenoperationen sind für p -adische Zahlendarstellung unabhängig von p gültig.

- Prolog
- Rechensysteme
- **Datendarstellung**

Umrechnung: Dezimal in Binär - Beispiel

Beispiel $2011_{10} =$

1005 R 1

502 R 1

251 R 0

125 R 1

62 R 1

31 R 0

15 R 1

7 R 1

3 R 1

1 R 1

0 R 1

EINI LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstel-
lung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Datendarstellung: Ganze Zahlen

p -adische Systeme definieren nur positive Zahlen und die 0.

▶ Vorzeichenbetragsdarstellung (VB-Zahlen)

- ▶ Standardverfahren unserer Schulmathematik
- ▶ Vorzeichen „+“ oder „-“ (3. + 4. Zeichen in der Kodierung)
- ▶ Unhandlich bei automatisierter Arithmetik

Bei unterschiedlichen Vorzeichen muss eine Fallunterscheidung für die Addition getroffen werden:

Seien $|x|$ und $|y|$ die Beträge der Summanden und $s = x + y$.

- ▶ $(x \geq 0 \wedge |x| \geq |y|) \Rightarrow (|s| = |x| - |y|) \wedge (s \geq 0)$
- ▶ $(x \geq 0 \wedge |x| < |y|) \Rightarrow (|s| = |y| - |x|) \wedge (s < 0)$
- ▶ $(x < 0 \wedge |x| \geq |y|) \Rightarrow (|s| = |x| - |y|) \wedge (s < 0)$
- ▶ $(x < 0 \wedge |x| < |y|) \Rightarrow (|s| = |y| - |x|) \wedge (s \geq 0)$

Datendarstellung: Ganze Zahlen

- ▶ 2er-Komplement
 - ▶ Vermeidet Vorzeichen
 - ▶ Anzahl der Stellen muss nicht bekannt sein
 - ▶ Erzeugung aus binärer Zahl: alle Stellen invertieren und 1 addieren
 - ▶ Berechnung als Dezimalwert: höchstwertiges Bit hat negativen Wert
- ▶ Addition kann sehr einfach auf die Addition von Binärzahlen zurückgeführt werden.

EINI LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Datendarstellung: Darstellung von Zahlen

Standardformate

- ▶ Wie bereits angedeutet, realisieren Rechner Zahlendarstellungen nur für bestimmte Wertebereiche (festes m).
- ▶ Diese Wertebereiche dienen als **Datentyp** für Variablen. Analog zu $X \in \mathbb{Z}$ wird vereinbart: `int x`.
- ▶ In der Programmiersprache Java werden folgende Bereiche angeboten (unabhängig von 32- oder 64-Bit-Versionen):

Bereich	Größe	Datentyp
$-128, \dots, 127$	8 Bit	byte
$-32768, \dots, 32767$	16 Bit	short
$-2^{31}, \dots, 2^{31}-1$	32 Bit	int
$-2^{63}, \dots, 2^{63}-1$	64 Bit	long

Datendarstellung: Festpunktzahlen

- ▶ Bisher: natürliche und ganze Zahlen (binär, 2er-Komplement)
- ▶ Gebrochene Zahlen: **Schwierigkeiten bei der Genauigkeit** der Darstellung:

$$x = \sum_{i=0}^{m-1} \alpha_i \cdot 2^i + \sum_{i=1}^{n-1} \alpha_i \cdot 2^{-i}$$

- ▶ Hinweise auf Schwierigkeiten im Dezimalsystem:
 - π (irrational), keine endliche Darstellung im Dezimalsystem
 - periodische, gebrochene Dezimalzahl bei $1/3$
- ▶ Binärzahlen: Problem auch bei Zahlen mit endlicher Dezimaldarstellung:
 - dezimal 0.1 wird binär zu $0.00011001100110011\dots$
- ▶ leider stehen natürlich nur eine endliche und feste Anzahl Bits zur Verfügung ...

**EINI LogWing /
WiMa**

Kapitel 1

Rechensysteme
und Datendarstel-
lung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Überlegungen zur Jahreszahl 2018

EINI LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstel-
lung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Datendarstellung: Gleitpunktzahlen

Gleitpunktzahlen in Programmiersprachen

Bereich	Bytes	Stellen	Delphi	Java
+ - 2,9 E -39 ... 1,7 E 38	6	11-12	real	
+ - 1,5 E -45 ... 3,4 E 38	4	7-8	single	float
+ - 5,0 E -324 ... 1,7 E 308	8	15-16	double	double
+ - 3,4 E -4932 ... 1,1 E 4932	10	19-20	extended	

❖ Die Notation mit „E“ (=Exponent) bedeutet

$$3,1415E2 = 3,1415 * 10^2 = 314,15$$

und entstammt Norm IEEE 754. Sie ist in Programmiersprachen üblich.

EINI LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

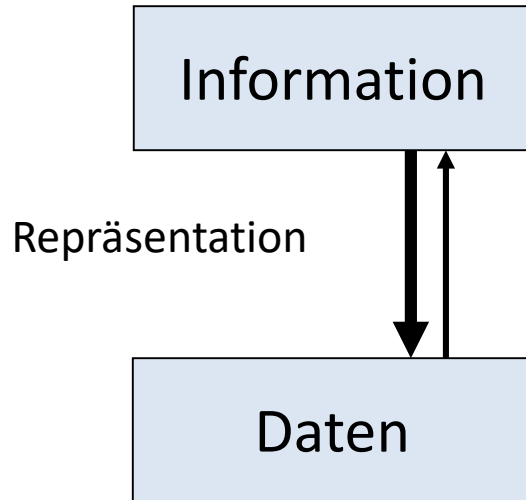
In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Wozu wird ein Rechensystem genutzt?

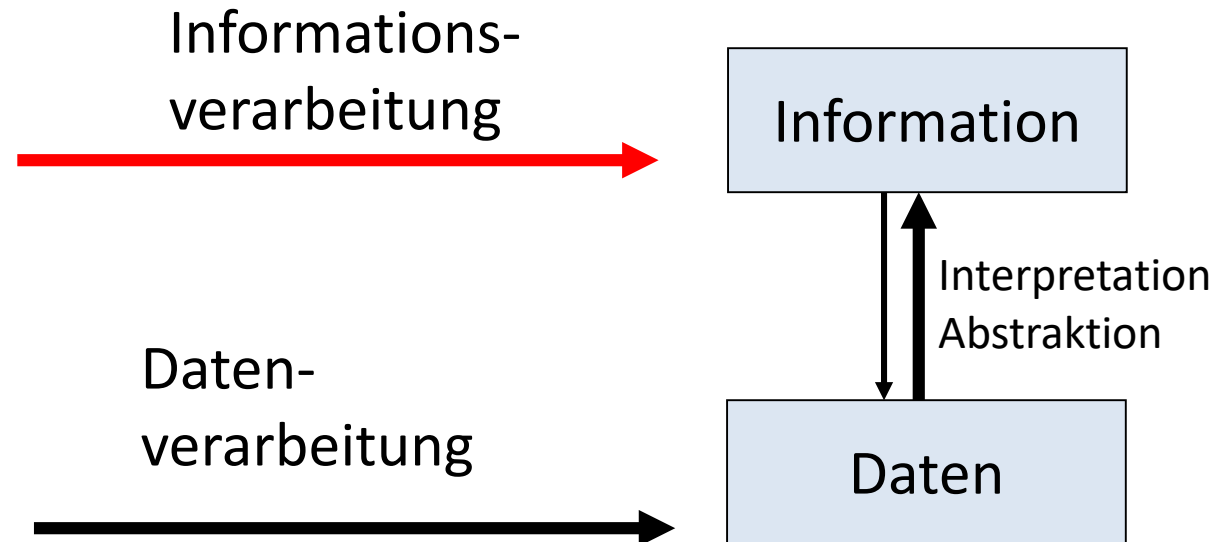
Wunsch:

Informationsverarbeitung



Wirklichkeit:

Datenverarbeitung



Bisher betrachtet:

Behandlung von einfachen mathematischen Objekten, nämlich Zahlen (natürliche, ganze Zahlen, reellwertige Zahlen)

Repräsentation und **Interpretation** sind wesentlich, um ein Rechensystem mit seinen Fähigkeiten zur Datenverarbeitung für die Informationsverarbeitung sinnvoll nutzen zu können.



Artikel im EINI-Wiki:

- **Bit**
- **Byte (Bitfolge)**
- **Bitfolgen**
- **Hexadezimalzahlen**
- **Maschinensprache**
- **Compiler**
- **Boolesche Algebra**
- **Dezimal- und Binärsystem**
- **Zweierkomplement**
- **Java**
- **Datentyp**
- **Programmiersprache**

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Zwischenstand

- ✓ Was ist Informatik ?
- ✓ Was macht ein Rechensystem aus?
- ✓ Frage: Wie werden Daten in einem Rechner dargestellt?
 - ✓ Buchstaben, Zeichenketten, Texte, ...
 - ✓ Grafiken
 - ✓ Algebren
 - ✓ Boolesche Algebra: Operationen AND, OR, NOT
 - ✓ Natürliche Zahlen, ganze Zahlen, reellwertige Zahlen: Operationen Addition, Division, Modulo, ...
 - ❖ Achtung: Genauigkeit der Darstellung und damit auch von Berechnungen ist begrenzt!
Wertebereiche für Zahlen sind beschränkt!
- Ziel: Mit dem **Entwurf von Algorithmen und Programmen**, der Programmierung von Rechensystemen und zugehörigen Programmiersprachen befassen.

EINI LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**



Vielen Dank für Ihre Aufmerksamkeit!

Nächste Termine

- | | |
|-------------------------------|-------------------|
| ▶ Nächste Vorlesung – WiMa | 25.10.2018, 08:15 |
| ▶ Nächste Vorlesung – LogWing | 26.10.2018, 08:15 |
| ▶ Beginn Praktikum | 22.10.2018 |