

EINI

WiMa/MP

Einführung in die Informatik

Vorlesung 2 SWS WS 26/27

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

Kapitel 1

Rechensysteme und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- Datendarstellung

WISE 26/27 | EINFÜHRUNG IN DIE INFORMATIK (EINI) PRAKTIKUM – STATISTIKEN + SOLVER

Übungszeitslot	Übung	Übungsgruppenleiter	Platzangebot	Beliebtheit (# Prio 1)
Mo : 10:00 - 12:30	Gruppe 1	Hildebrand	15	<div><div></div></div> 26
Mo : 12:30 - 15:00	Gruppe 2	Hildebrand	15	<div><div></div></div> 30
Di : 12:30 - 15:00	Gruppe 3	Hildebrand	15	<div><div></div></div> 7
Di : 16:00 - 18:30	Gruppe 4	Hildebrand	15	<div><div></div></div> 8
Mi : 10:00 - 12:30	Gruppe 5	Hildebrand	15	<div><div></div></div> 18
Do : 14:00 - 16:30	Gruppe 6	Hildebrand	15	<div><div></div></div> 9
Interessenten gesamt:				<div><div></div></div> 74 / 90 = 82%

EINI LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

Zuteilung:

Montag 10:00 - 12:30	Gruppe 1	Hildebrand	15 von 15	100%
Montag 12:30 - 15:00	Gruppe 2	Hildebrand	15 von 15	100%
Dienstag 12:30 - 15:00	Gruppe 3	Hildebrand	15 von 15	100%
Dienstag 16:00 - 18:30	Gruppe 4	Hildebrand	7 von 15	46.7%
Mittwoch 10:00 - 12:30	Gruppe 5	Hildebrand	15 von 15	100%
Donnerstag 14:00 - 16:30	Gruppe 6	Hildebrand	7 von 15	46.7%
keine Übung			0 von 74	0%

In diesem Kapitel:

- Prolog
- Rechensysteme
- Datendarstellung

Zuteilung:

Montag 10:00 - 12:30	Gruppe 1	Hildebrand	15 von 15	100%
Montag 12:30 - 15:00	Gruppe 2	Hildebrand	15 von 15	100%
			15 von 15	100%
Erfolg: Es sind 71 E-Mails erfolgreich versendet worden. 3 Adressaten haben keine E-Mailadresse hinterlegt.				
Erfolg: Die Lösung wurde erfolgreich in die Datenbank übertragen.				
Mittwoch 10:00 - 12:30	Gruppe 5	Hildebrand	15 von 15	100%
Donnerstag 14:00 - 16:30	Gruppe 6	Hildebrand	7 von 15	46.7%
keine Übung			0 von 74	0%

In diesem Kapitel:

- Prolog
- Rechensysteme
- Datendarstellung

Themen

► Rechensysteme

- Was macht ein Rechensystem aus?
 - Hardware
 - Verwendung

► Datendarstellung

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

EINI LogWing /
WiMa

Kapitel 1

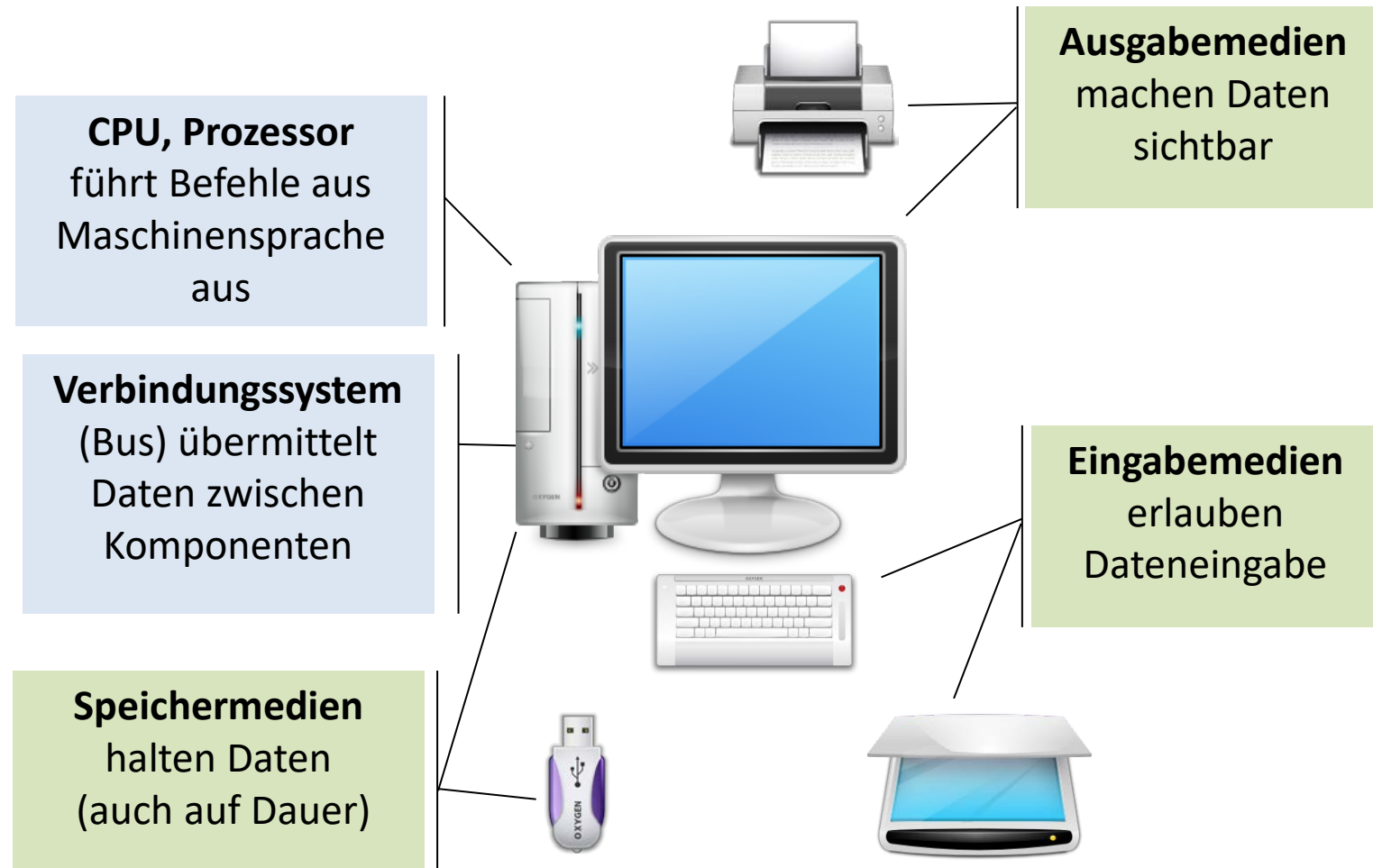
Rechensysteme
und Datendarstel-
lung

In diesem Kapitel:

- 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

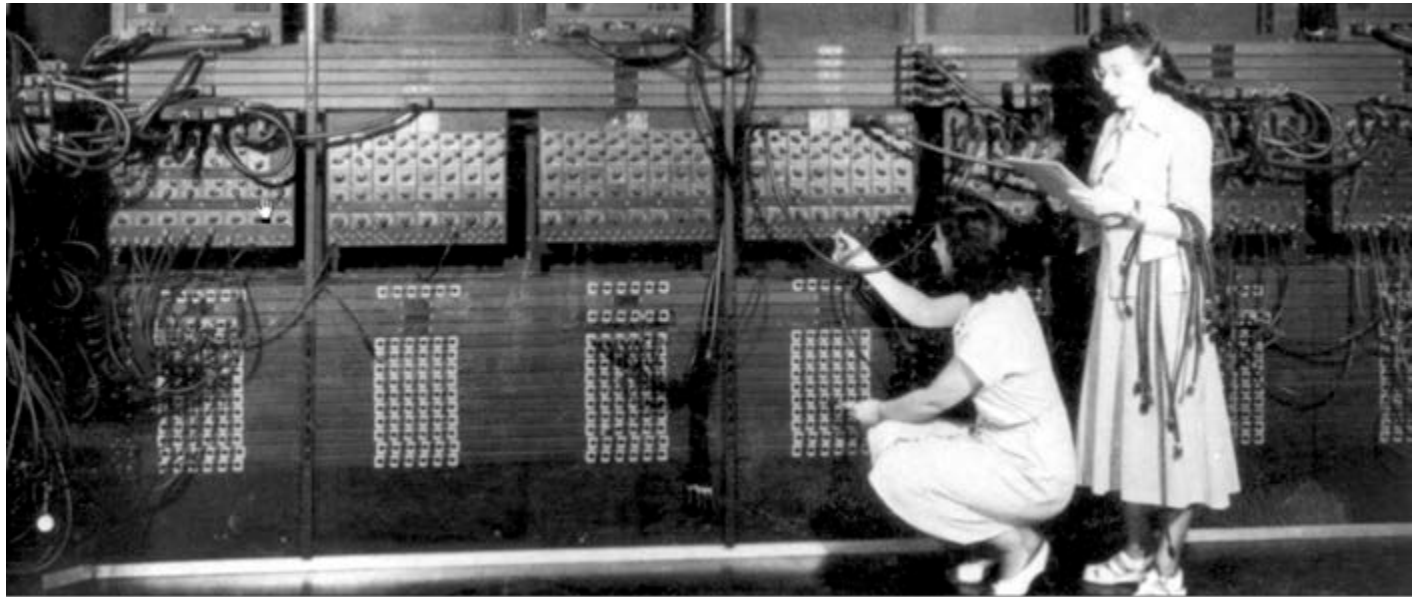
In diesem Kapitel:

- 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

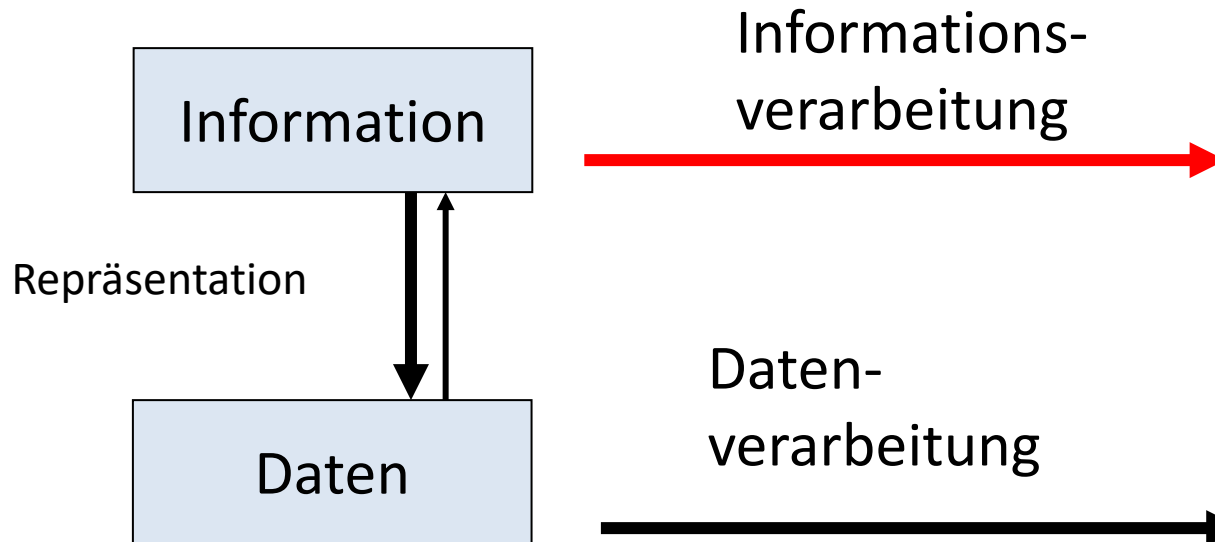


Quelle: TomsHardware.com

Wozu wird ein Rechensystem genutzt?

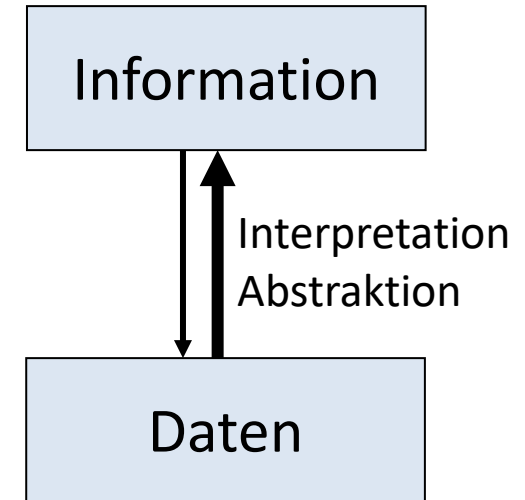
Wunsch:

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**

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

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.

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.

Byte

- ▶ Rechner behandeln keine einzelnen Bits: kleinste betrachtete Bitfolge ist das Byte = 8 Bits
- ▶ größere 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:

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

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

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

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

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

- ▶ 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.

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

EINI LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstel-
lung

20	0	30	@	40	P	50	`	60	p	70		A0	°	B0	À	C0	Ð	D0	à	E0	ð	F0
21	1	31	A	41	Q	51	a	61	q	71	¡	A1	±	B1	Á	C1	Ñ	D1	á	E1	ñ	F1
22	2	32	B	42	R	52	b	62	r	72	¢	A2	²	B2	Â	C2	Ò	D2	â	E2	ò	F2
23	3	33	C	43	S	53	c	63	s	73	£	A3	³	B3	Ã	C3	Ó	D3	ã	E3	ó	F3
24	4	34	D	44	T	54	d	64	t	74	¤	A4	´	B4	Ä	C4	Ô	D4	ä	E4	ô	F4
25	5	35	E	45	U	55	e	65	u	75	¥	A5	µ	B5	Å	C5	Õ	D5	å	E5	õ	F5
26	6	36	F	46	V	56	f	66	v	76	¦	A6	¶	B6	Æ	C6	Ö	D6	æ	E6	ö	F6
27	7	37	G	47	W	57	g	67	w	77	§	A7	·	B7	Ç	C7	×	D7	ç	E7	÷	F7
28	8	38	H	48	X	58	h	68	x	78	¨	A8	¸	B8	È	C8	Ø	D8	è	E8	ø	F8
29	9	39	I	49	Y	59	i	69	y	79	©	A9	¹	B9	É	C9	Ù	D9	é	E9	ù	F9
2A	:	3A	J	4A	Z	5A	j	6A	z	7A	ª	AA	º	BA	Ê	CA	Ú	DA	ê	EA	ú	FA
2B	;	3B	K	4B	[	5B	k	6B	{	7B	«	AB	»	BB	Ë	CB	Û	DB	ë	EB	û	FB
2C	<	3C	L	4C	\	5C	l	6C		7C	¬	AC	¼	BC	Ì	CC	Ü	DC	ì	EC	ü	FC
2D	=	3D	M	4D	]	5D	m	6D	}	7D	–	AD	½	BD	Í	CD	Ý	DD	í	ED	ý	FD
2E	>	3E	N	4E	^	5E	n	6E	~	7E	®	AE	¾	BE	Î	CE	Þ	DE	î	EE	þ	FE
2F	?	3F	O	4F	_	5F	o	6F		7F	–	AF	¿	BF	Ï	CF	ß	DF	ï	EF	ÿ	FF

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**

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Lateinische Schriften und Symbole
10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	
20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F	Lautschriften
30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F	
40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	Andere europäische Schriften
50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F	
60	61	62	63	64	65	66	67	68	69	6A	6B	6C	6D	6E	6F	Nahost- und Südwestasiatische Schriften
70	71	72	73	74	75	76	77	78	79	7A	7B	7C	7D	7E	7F	
80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F	Afrikanische Schriften
90	91	92	93	94	95	96	97	98	99	9A	9B	9C	9D	9E	9F	
A0	A1	A2	A3	A4	A5	A6	A7	A8	A9	AA	AB	AC	AD	AE	AF	Südasiatische Schriften
B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	BA	BB	BC	BD	BE	BF	
C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF	Südostasiatische Schriften
D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	DA	DB	DC	DD	DE	DF	
E0	E1	E2	E3	E4	E5	E6	E7	E8	E9	EA	EB	EC	ED	EE	EF	Ostasiatische Schriften
F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	FA	FB	FC	FD	FE	FF	

Lateinische Schriften und Symbole

Lautschriften

Andere europäische Schriften

Nahost- und Südwestasiatische Schriften

Afrikanische Schriften

Südasiatische Schriften

Südostasiatische Schriften

Ostasiatische Schriften

CJK-Ideogramme

Kanadische Silben

Symbole

Diakritika

UTF-16-Surrogates und privater Nutzungsbereich

Verschiedene Zeichen

Nicht belegte Codebereiche

Kanadische Silben: *Unified Canadian Aboriginal Syllabics*

$$\sigma \Delta \leq \Delta C$$

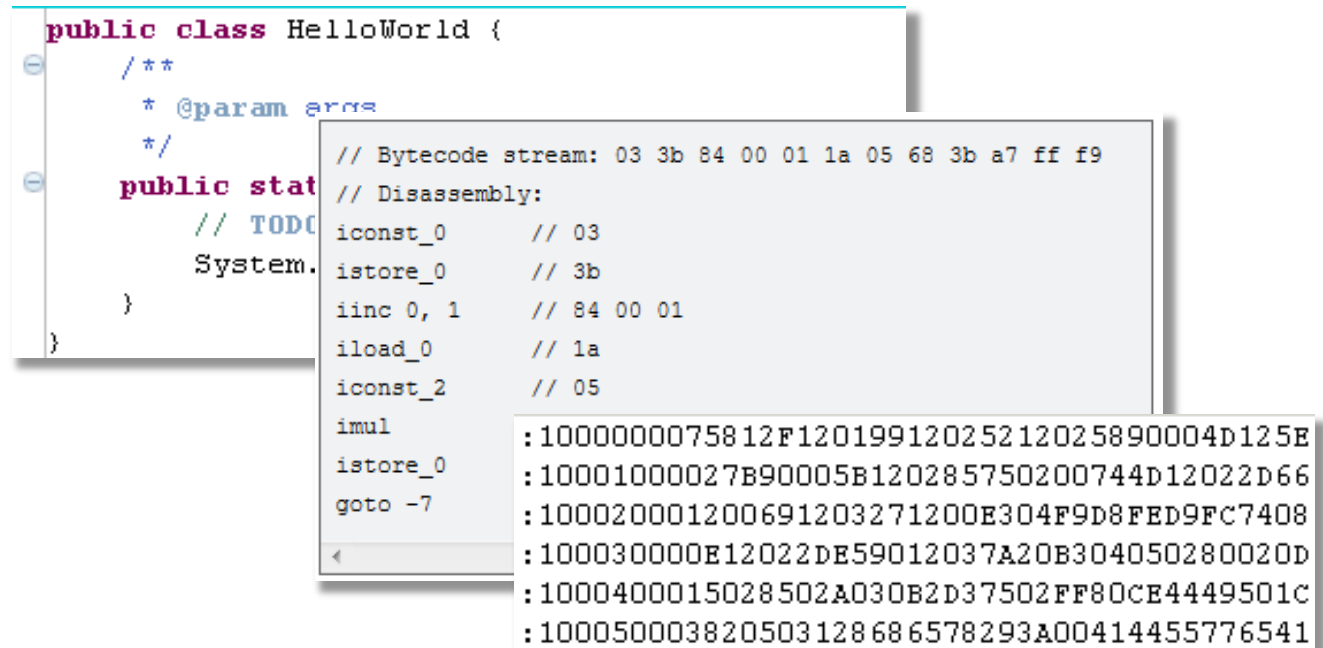
tu technische universität
dortmund

20

- Prolog
- Rechensysteme
- **Datendarstellung**

► Programme

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



The screenshot shows a Java IDE with a code editor on the left and a bytecode viewer on the right. The code editor displays the following Java code:

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

The bytecode viewer on the right shows the following information:

```
// 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  
istore_0      // 3b  
goto -7       // a7 ff f9
```

Below the disassembly, a hex dump of the bytecode stream is shown:

```
: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,

Datendarstellung: Logische/Boolesche Werte

▶ Boolesche Algebra

- ▶ **Trägermenge** = {false,true} (oft auch als {0,1}) mit

▶ 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: $p=8$

Umrechnung: Dezimal in Binär

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

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

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

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

also **fortgesetztes Dividieren**.

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

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

$4711 / 2 = 2355$ mit Rest 1 -> „**rechteste**“ 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.

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,...,127	8 Bit	byte
-32768,...,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 2020

EINI LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

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

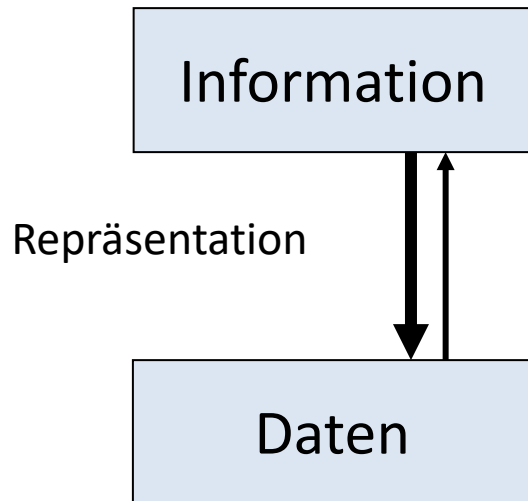
In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Wozu wird ein Rechensystem genutzt?

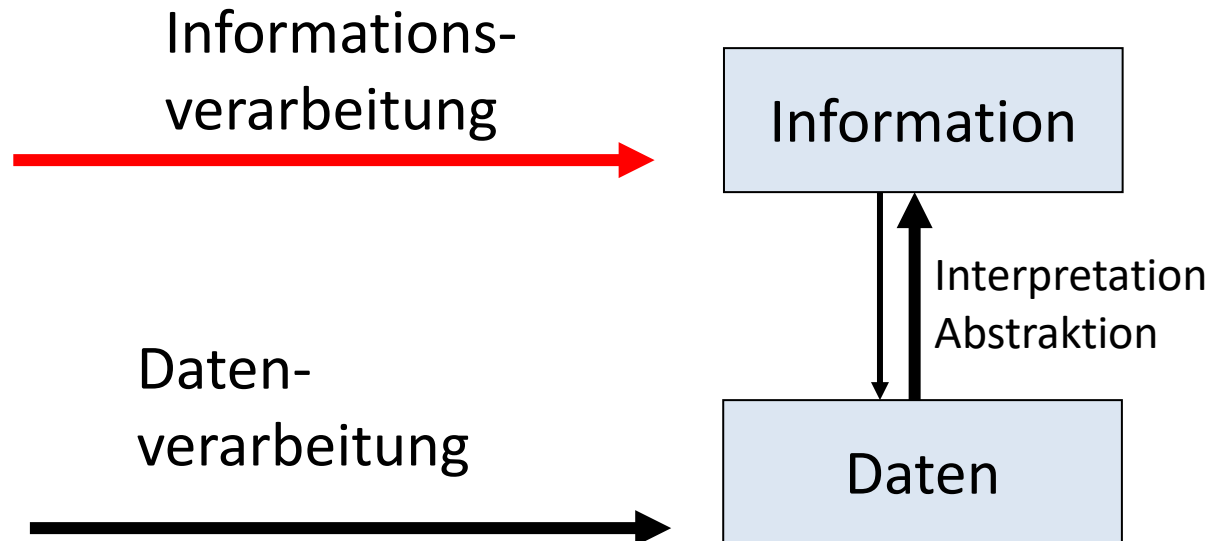
Wunsch:

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

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**



Vielen Dank für Ihre Aufmerksamkeit!

Nächste Termine

- ▶ Nächste Vorlesung – WiMa 29.10.2026, 08:15
- ▶ Beginn Praktikum 26.10.2026