

EINI LW

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

Vorlesung 2 SWS WS 14/15

Dr. Lars Hildebrand

Fakultät für Informatik – Technische Universität Dortmund

lars.hildebrand@tu-dortmund.de

<http://ls1-www.cs.tu-dortmund.de>

Eini LogWing / WiMa

Kapitel 1

Rechensysteme und Datendarstellung

In diesem Kapitel:

- **Prolog**
- Rechensysteme
- Datendarstellung

Übung	Übungsgruppenleiter	Priorität	Beliebtheit (# Prio 1)
01, Mo., 8:00-11:00, OH-12 R3.032	Henning Timm	1 - höchste	9
02, Mo., 8:00-11:00, OH-12 R3.033	Bartho Rudak	1 - höchste	8
03, Mo., 11:00-14:00, OH-12 R3.032	Sebastian Hauer	1 - höchste	207
04, Mo., 14:00-16:15, OH-12 R3.032	Henning Timm	1 - höchste	20
05, Mo., 14:00-16:15, OH-12 R3.033	Sebastian Hauer	1 - höchste	17
06, Mo., 16:15-18:30, OH-12 R3.032	Ronald Hund	1 - höchste	40
07, Di., 8:00-11:00, OH-12 R3.032	Marcel Preuß	1 - höchste	17
08, Di., 8:00-11:00, OH-12 R3.033	Bartho Rudak	1 - höchste	15
09, Di., 11:00-14:00, OH-12 R3.032	Iman Kamehkhosh	1 - höchste	62
10, Di., 11:00-14:00, OH-12 R3.033	Thomas Schmitz	1 - höchste	60
11, Di., 14:00-16:15, OH-12 R3.032	Iman Kamehkhosh	1 - höchste	23
12, Di., 14:00-16:15, OH-12 R3.033	Thomas Schmitz	1 - höchste	23
13, Di., 16:15-18:30, OH-12 R3.032	David Sturm	1 - höchste	13
14, Mi., 8:00-11:00, OH-12 R3.032	Alexander Lochmann	1 - höchste	33
15, Mi., 8:00-11:00, OH-12 R3.033	Marcel Preuß	1 - höchste	22
16, Mi., 11:00-14:00, OH-12 R3.032	Viktoria Hübert	1 - höchste	15
17, Mi., 11:00-14:00, OH-12 R3.033	Sergej Karsten	1 - höchste	14
18, Mi., 14:00-16:15, OH-12 R3.032	Viktoria Hübert	1 - höchste	65
19, Mi., 14:00-16:15, OH-12 R3.033	David Sturm	1 - höchste	55
20, Do., 11:00-14:00, OH-12 R3.032	Alexander Lochmann	1 - höchste	63
21, Do., 14:00-16:15, OH-12 R3.032	Solveig Ahorner	1 - höchste	38
22, Fr., 11:30-14:00, OH-12 R3.032	Sergej Karsten	1 - höchste	18
23, Fr., 14:00-16:15, OH-12 R3.032	Ronald Hund	1 - höchste	18

Praktikum & Übung

- ▶ es existieren noch Gruppen mit freien Plätzen
- ▶ Benachrichtigung per EMail

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- **Prolog**
- Rechensysteme
- Datendarstellung

Praktikum & Übung

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

- ▶ es existieren noch Gruppen mit freien Plätzen
- ▶ Benachrichtigung per EMail
- ▶ an Alle
 - ▶ bis auf einige wenige Studierende
 - ▶ persönlich nachfragen



In diesem Kapitel:

- Prolog
- Rechensysteme
- Datendarstellung

In diesem Kapitel:

- **Prolog**
- Rechensysteme
- Datendarstellung

▶ **Rechensysteme**

- ▶ Was macht ein Rechensystem aus?
- ▶ Schichten eines Computer Systems
- ▶ Wozu werden Rechensysteme benutzt?

▶ **Datendarstellung**

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

Was macht ein Rechengesystem aus?

Hardware

- ▶ physikalische Komponenten,
 - ▶ die spezielle Leistungen erbringen,
 - ▶ Funktionen verfügbar machen

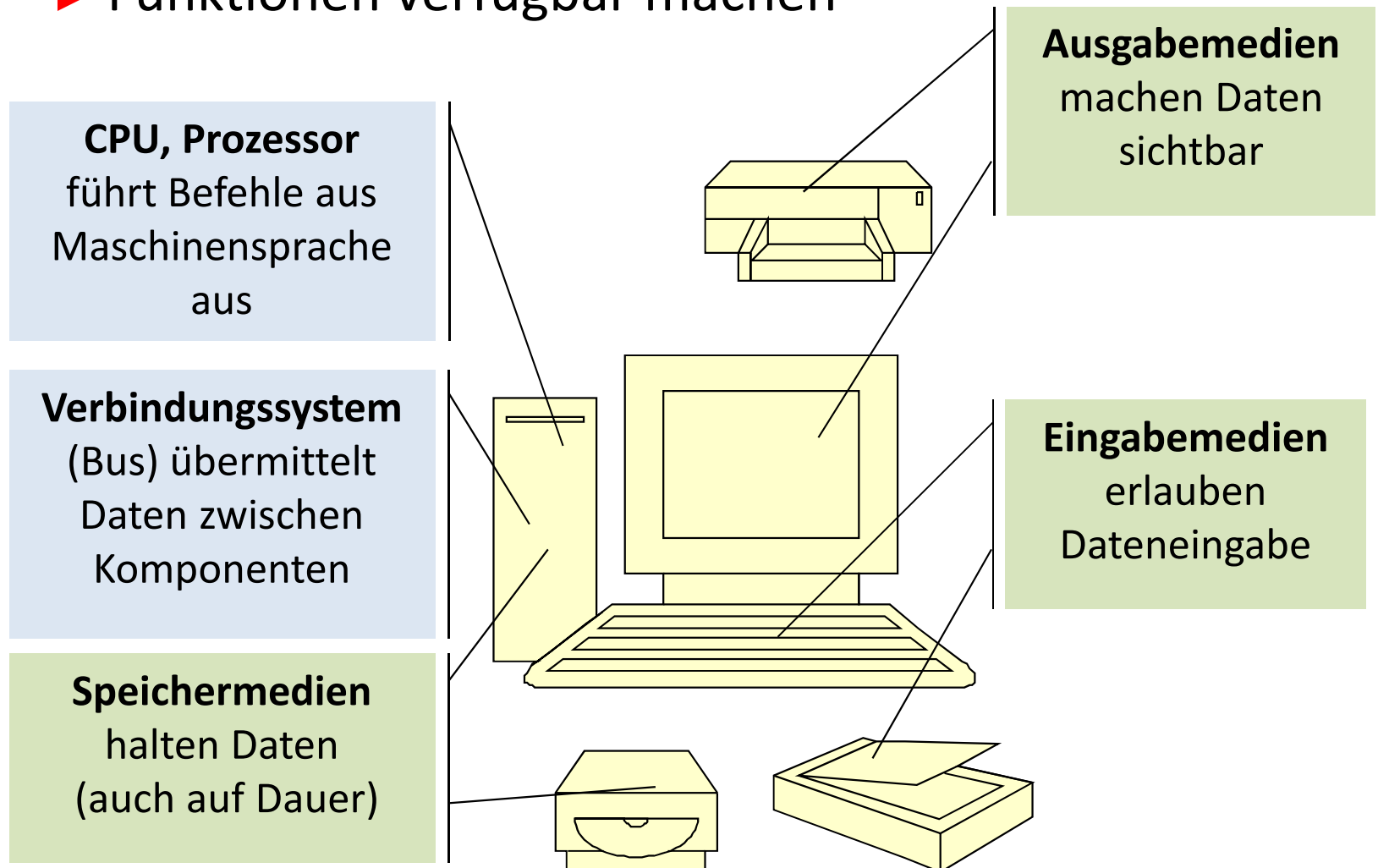
Eini LogWing /
WiMa

Kapitel 1

Rechengesysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- **Rechengesysteme**
- Datendarstellung



Was macht ein Rechensystem aus?

Wofür wird es verwendet?

- ▶ Von Privatanwendern:
 - ▶ Textverarbeitung
 - ▶ Tabellenkalkulation
 - ▶ Elektronische Post (Email)
 - ▶ Internet surfen, Informationsbeschaffung, ...

- ▶ Von Firmen:
 - ▶ zusätzlich zu den vorherigen Punkten,
 - ▶ Verwaltung von Firmendaten und Arbeitsvorgängen, Produktionsplanung und -steuerung, Buchhaltung, ...
 - Datenbankapplikationen,
 - Enterprise Ressource Systeme wie SAP R/3
 - ▶ Steuerung automatisierter Fertigungsanlagen

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstel-
lung

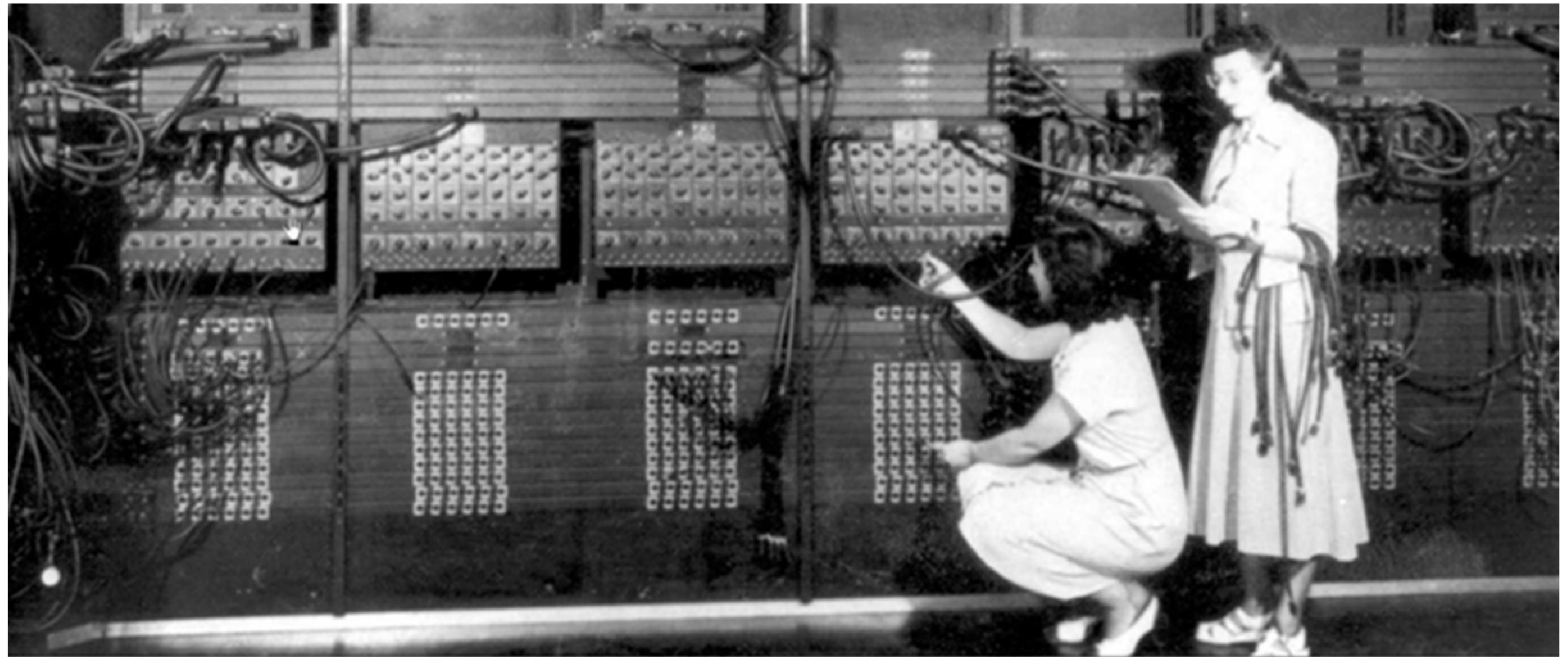
In diesem Kapitel:

- Prolog
- **Rechensysteme**
- Datendarstellung

Warum geht das?

Frei programmierbare Rechner

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



Eini LogWing /
WiMa

Kapitel 1

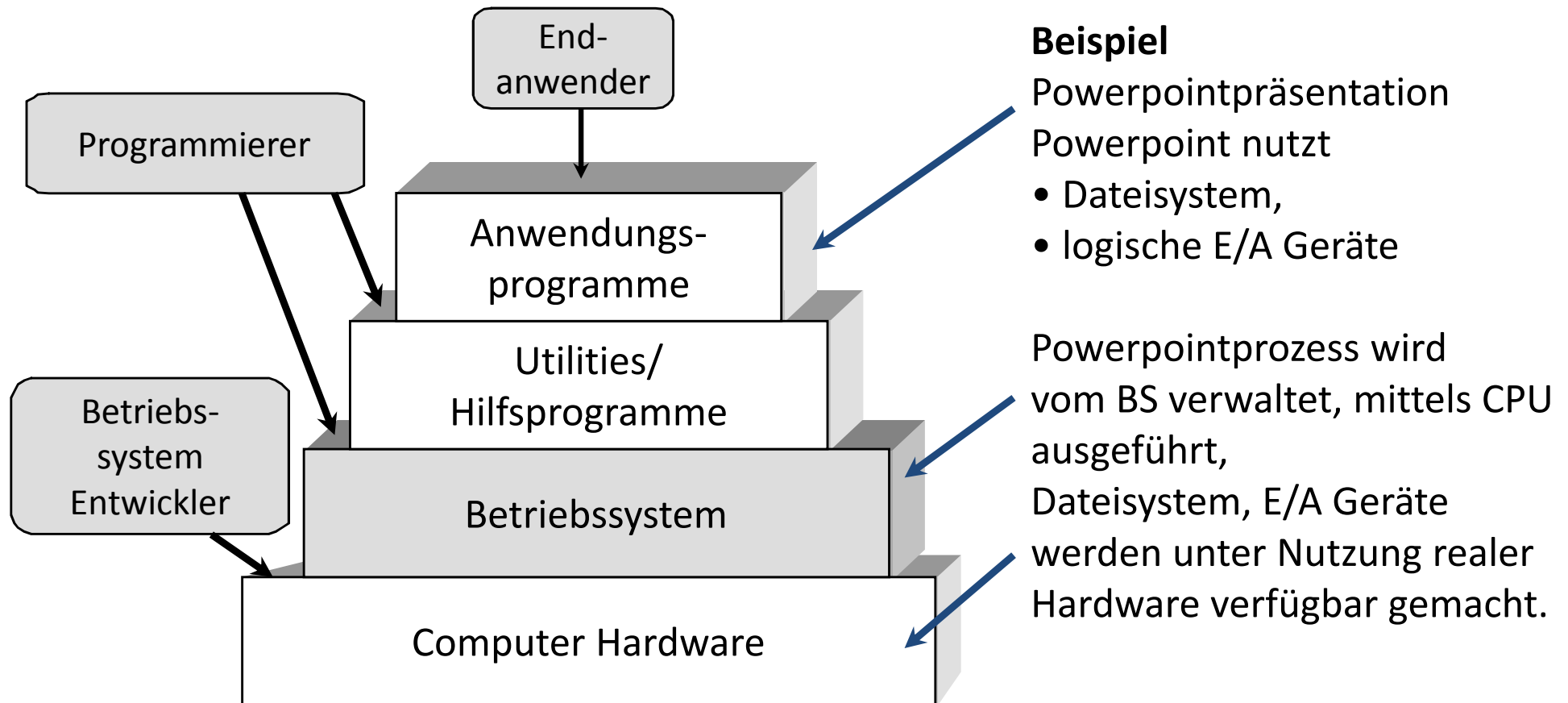
Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- **Rechensysteme**
- Datendarstellung

Quelle: TomsHardware.com

Schichten (Sichten) eines Rechensystems



Das Betriebssystem vermittelt zwischen Anwendungsprogrammen und realer Hardware. Z.B. hat Powerpoint eine „Save“ Funktion zum Speichern einer Präsentation in einer Datei. Das Betriebssystem stellt hierzu elementare Operationen zum Lesen / Schreiben von Dateien zur Verfügung und verwaltet Dateien. Es realisiert eine **virtuelle Maschine**.

Virtuelle Maschine

Eine **virtuelle Maschine** ist eine Menge von Funktionen /Diensten, die sich real nutzen lassen, jedoch durch Software hergestellt werden.

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

- ▶ häufig verwendetes Konzept der Informatik zur Beherrschung von Komplexität durch **Zerlegen in beherrschbare Teilprobleme** und **Kopplung der Teillösungen** mittels Schnittstellen (Teile&Herrsche), wobei eine Teillösung (VM) dann in unterschiedlichen Kontexten genutzt werden kann.
- ▶ Z.B. **Betriebssystem**: erzeugt eine **Schnittstelle** festen Funktionsumfangs (=>VM), so dass Anwendungsprogramme auf dieser VM trotz **unterschiedlicher HW Plattformen** ablaufen können. Eine Leistung ist z.B., die real begrenzte Menge Hauptspeicher für das Anwendungsprogramm als quasi beliebig groß erscheinen zu lassen.
- ▶ z.B. **Kommunikationsprotokolle**: erzeugen eine Schnittstelle zur **Übermittlung** von Daten. Z.B. TCP/IP bewirkt zuverlässige Verbindung zwischen Kommunikationspartnern, obwohl in Wirklichkeit einzelne Pakete über Zwischenstationen und mit unsicheren Verbindungen durch ein Netzwerk transferiert werden.

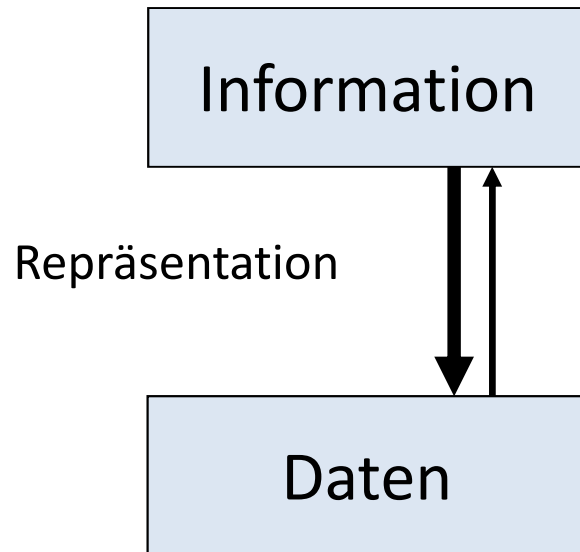
In diesem Kapitel:

- Prolog
- **Rechensysteme**
- Datendarstellung

Wozu wird ein Rechensystem genutzt?

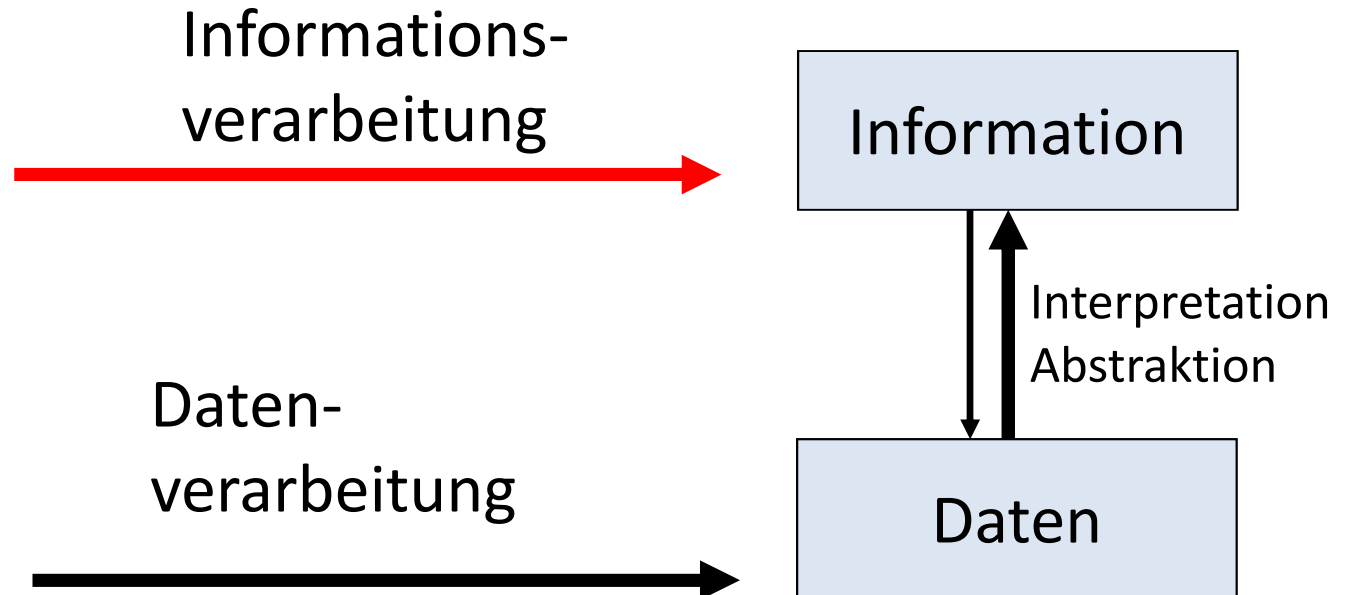
Wunsch:

Informationsverarbeitung



Wirklichkeit:

Datenverarbeitung



Anmerkung 1:

Die grundsätzliche Crux der Informatik besteht darin, dass ein System ohne eigenes Verstehen, 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 im direkten Anschluss mit der elementarsten Ebene, nämlich durch Werte 0 und 1, befassen.

Wo stehen wir jetzt?

- ▶ Was ist Informatik ?
- ▶ Was macht ein Rechensystem aus ?
- ▶ Daten vs. Informationen führt zur Frage: Wie werden Daten in einem Rechner dargestellt ?
 - ▶ Buchstaben, Zeichenketten, Texte, ...
 - ▶ Grafiken
 - ▶ Algebren
 - Boolesche Algebra mit Operationen AND, OR, NOT
 - Natürliche Zahlen, ganze Zahlen, reellwertige Zahlen mit Operationen Addition, Subtraktion, Multiplikation, Division, Modulo, ...
 - Achtung: Genauigkeit der Darstellung und damit auch von Berechnungen ist begrenzt!
Wertebereiche für Zahlen sind beschränkt!
- ▶ Wir wollen uns letztlich jedoch 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 für Datendarstellungen
- ▶ Datendarstellung Überblick
- ▶ Texte
- ▶ Programme, Graphiken
- ▶ Logische, bzw. 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 für Datendarstellung

Bits

- ▶ kleinstmögliche Einheit der Information(darstellung)
- ▶ erlaubt Antwort auf eine Frage mit nur zwei Antwortmöglichkeiten,
 - ▶ z.b. {ja,nein}, {wahr,falsch}, {schwarz,weiß}, {links,rechts},
 - ▶ meist durch {0,1} codiert
- ▶ technisch umgesetzt 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.

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **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 mögliche 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 nicht einzelne 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

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

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

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

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

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

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

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

Für Längen, Zeiten:

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

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

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

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

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

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Datendarstellung: Komplexe Datentypen

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

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

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Datendarstellung: Texte

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

- ▶ **Texte:** Zeichenfolgen aus Buchstaben und Satzzeichen
 - ▶ Darstellung mittels Bitfolgen
 - ▶ 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“ hat Nummer 97, „A“ hat Nummer 65, „?“ hat Nummer 63
 - ▶ Klein- und Großbuchstaben nach Alphabet durchnummeriert
 - ▶ übliche Erweiterung auf PCs: 8bit, 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>
 - ▶ 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	¢	Â	Ò	â	ò
0023	#	3	C	S	c	s	£	Ã	Ó	ã	ó
0024	\$	4	D	T	d	t	¤	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		¬	Ì	Ü	ì	ü
002D	-	=	M]	m	}	¯	Í	Ý	í	ý
002E	.	>	N	^	n	~	®	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

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
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
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
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
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
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
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
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
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

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Datendarstellung: Texte

► Kanadische Silben: Unified Canadian Aboriginal Syllabics

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

$\Delta \sigma^b \cap \text{C}$
 Inuktitut Syllabarium
 $\text{C} \sigma^b \text{C} \text{C} \text{C} \text{C}$

Anlaute	Silben				Auslaute
	ai	i	u	a	
∅	▽	△	▷	◁	
p	∇	∧	>	<	◁◁
t	∪	∩	⊃	⊂	◁◁ ^c
k	∩	∪	∂	∂	◁◁ ^b
g	∩	∪	∂	∂	◁◁ ^l
m	∩	∪	∂	∂	◁◁ ^L
n	∅	∅	∅	∅	◁◁ ^o
s	∩	∪	∂	∂	◁◁ ^s
l	∩	∪	∂	∂	◁◁ ^c
j	∩	∪	∂	∂	◁◁ ^l
v	∩	∪	∂	∂	◁◁ ^e
r	∩	∪	∂	∂	◁◁ ^s
q	∩	∪	∂	∂	◁◁ ^{so}
ng	∩	∪	∂	∂	◁◁ ^o
nng	∩	∪	∂	∂	◁◁ ^{so}
∩		∩	∪	∂	◁◁ ^s

Datendarstellung: Programme

▶ Programme

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

```
public class HelloWorld {  
    /**  
     * @param args  
     */  
    public static void main(String[] args) {  
        // TODO: write the logic here  
        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 :1000000075812F12019912025212025890004D125E
istore_0 :10001000027B90005B120285750200744D12022D66
goto -7 :100020001200691203271200E304F9D8FED9FC7408
:100030000E12022DE59012037A20B304050280020D
:1000400015028502A030B2D37502FF80CE4449501C
:100050003820503128686578293A00414455776541

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Datendarstellung: Grafiken

▶ Grafiken, Bilder

▶ Rastergrafik

- ▶ Grafik wird aus einer Folge einzelner Rasterpunkte dargestellt
- ▶ Einzelner Rasterpunkt durch 1 Bit, 1 oder mehrere Bytes (wg. Farbe) codiert



▶ Vektorgrafik

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



Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

- Prolog
- Rechensysteme
- **Datendarstellung**

▶ Boolesche Algebra:

- ▶ **Trägermenge** = {false,true} (oft auch als {0,1}) mit
- ▶ **Operationen** (Auswahl)
 - 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 im Rechner ist 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

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

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

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: $p=8$

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- 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

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- 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 man eine Fallunterscheidung für die Addition treffen:

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

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

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Datendarstellung: Ganze Zahlen

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

- ▶ 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

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 Variable, analog zu $X \in \mathbb{N}$ wird vereinbart: `int x`
- ▶ in der Programmiersprache Java werden folgende Bereiche angeboten (unabhängig von 32- oder 64-bit Versionen)

Bereich	Größe	Java
$-128, \dots, 127$	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

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Datendarstellung: Festpunktzahlen

- ▶ Bisher natürliche und ganze Zahlen (binär, 2-er Komplement)
- ▶ Bei gebrochenen Zahlen zeigen sich 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 gibt es bereits im Dezimalsystem
 - π (irrational), keine endliche Darstellung im Dezimalsystem
 - periodische, gebrochene Dezimalzahl bei $1/3$
- ▶ bei Binärzahlen tritt Problem auch bei Zahlen mit endlicher Dezimaldarstellung auf:
 - dezimal 0.1 wird binär zu $0.00011001100110011\dots$
- ▶ wiederum stehen natürlich nur eine endliche und feste Anzahl Bits zur Verfügung ...

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstel-
lung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Überlegungen zur Jahreszahl 2013

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Datendarstellung: Gleitpunktzahlen

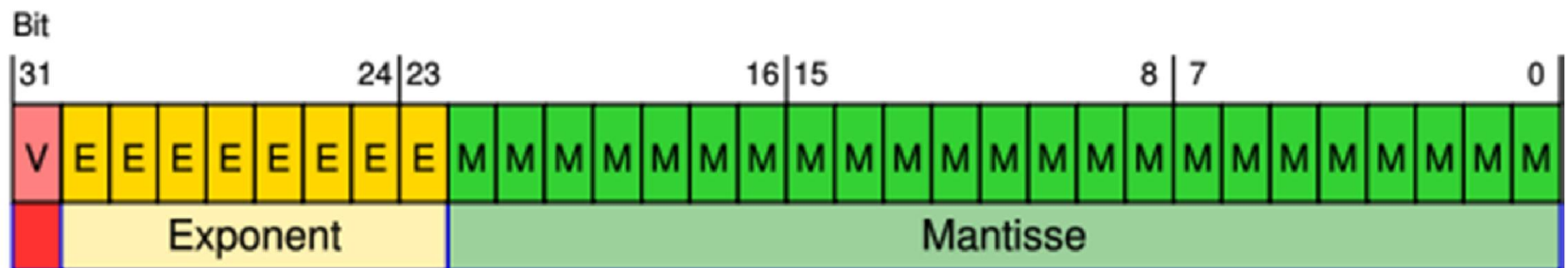
Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

- ▶ Gleitpunktzahlen bestehen aus 3 Teilen
 - ▶ Vorzeichenbit: **V**, gibt an, ob die Zahl positiv oder negativ ist
 - ▶ Exponent: **E**, gibt für eine Basis (typisch: $p=2$), einen Exponenten als Binärzahl an, mit der die Mantisse zu multiplizieren ist
 - ▶ Mantisse: **M**, ist eine Folge von Binärziffern m_1, \dots, m_n , die interpretiert wird als $m_1 \cdot 2^{-1} + m_2 \cdot 2^{-2} + \dots + m_n \cdot 2^{-n}$

$$w = (-1)^v * \left(1 + \sum_{i=-1}^{-23} m_i * 2^i\right) * \left(2^{\sum_{i=0}^7 e_i * 2^i - 127}\right)$$



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“ wie Exponent bedeutet:

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

und entstammt Norm IEEE 754, ist in Programmiersprachen üblich.

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

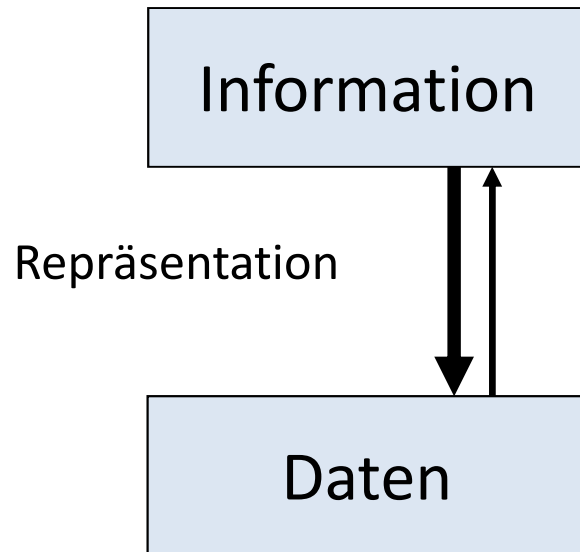
In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

Wozu wird ein Rechensystem genutzt?

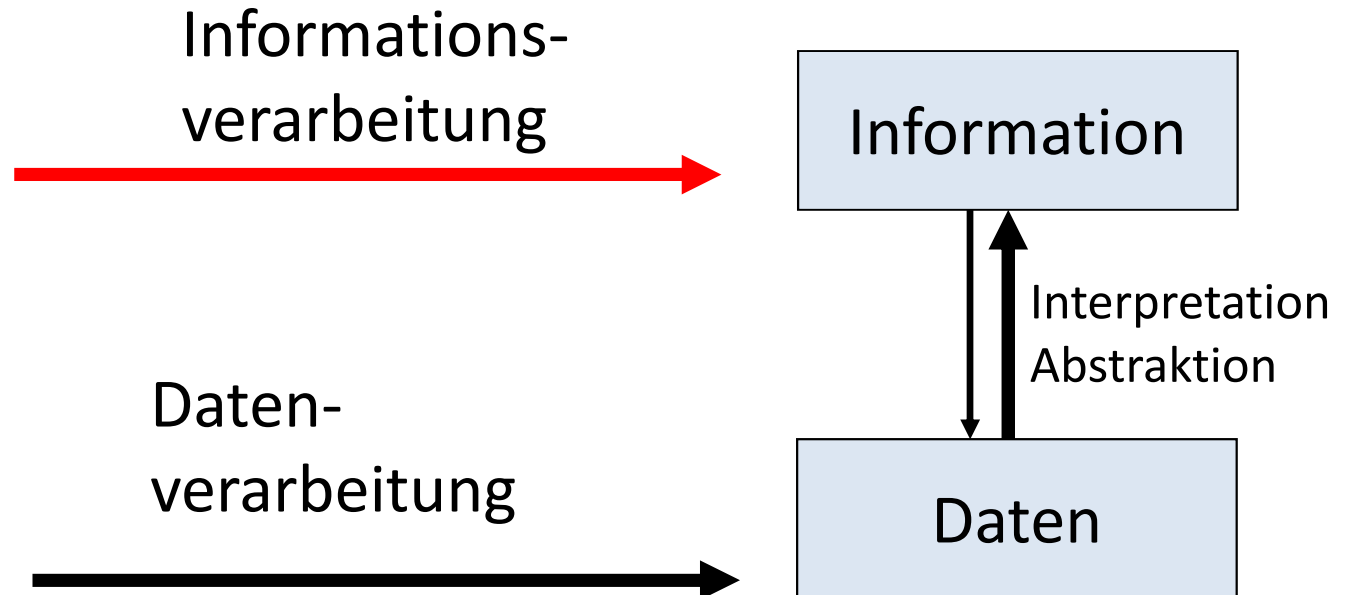
Wunsch:

Informationsverarbeitung



Wirklichkeit:

Datenverarbeitung



Bisher betrachtet:

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

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

Wo stehen wir jetzt?

Eini LogWing /
WiMa

Kapitel 1

Rechensysteme
und Datendarstellung

In diesem Kapitel:

- Prolog
- Rechensysteme
- **Datendarstellung**

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



Vielen Dank für Ihre Aufmerksamkeit!

Nächste Termine

- ▶ Nächste Vorlesung – WiMa 29.10.2014, 08:15
- ▶ Nächste Vorlesung – LogWIng 30.10.2014, 08:15
- ▶ Beginn Praktikum 27.10.2014