

Rechnerstrukturen, Teil 1



Vorlesung 4 SWS WS 19/20

Prof. Dr. Jian-Jia Chen

Fakultät für Informatik – Technische Universität Dortmund

jian-jia.chen@cs.uni-dortmund.de

<http://ls12-www.cs.tu-dortmund.de>

1. Organisatorisches

1. Organisatorisches

1. Vorlesung

2. Übungen

3. Übungsanmeldung

4. HelpDesk

5. Klausur

2. Einleitung

3. Repräsentation von Daten

1.1 Vorlesung

Vorlesung **Rechnerstrukturen** (4V + 2Ü)

- Termine Mo 16:15–17:45 Uhr (SRG, HS1)
 Mi 16:15 – 17:45 Uhr (SRG, HS1)
- regelmäßige **Teilnahme** – **erwünscht**
- regelmäßige **Nachbereitung** – **erforderlich**
- Faustregel Vorlesung : Nachbereitung = 1 : 1
- Skript begleitend lesen – **empfehlenswert**
- Übungen dazu – **erforderlich**

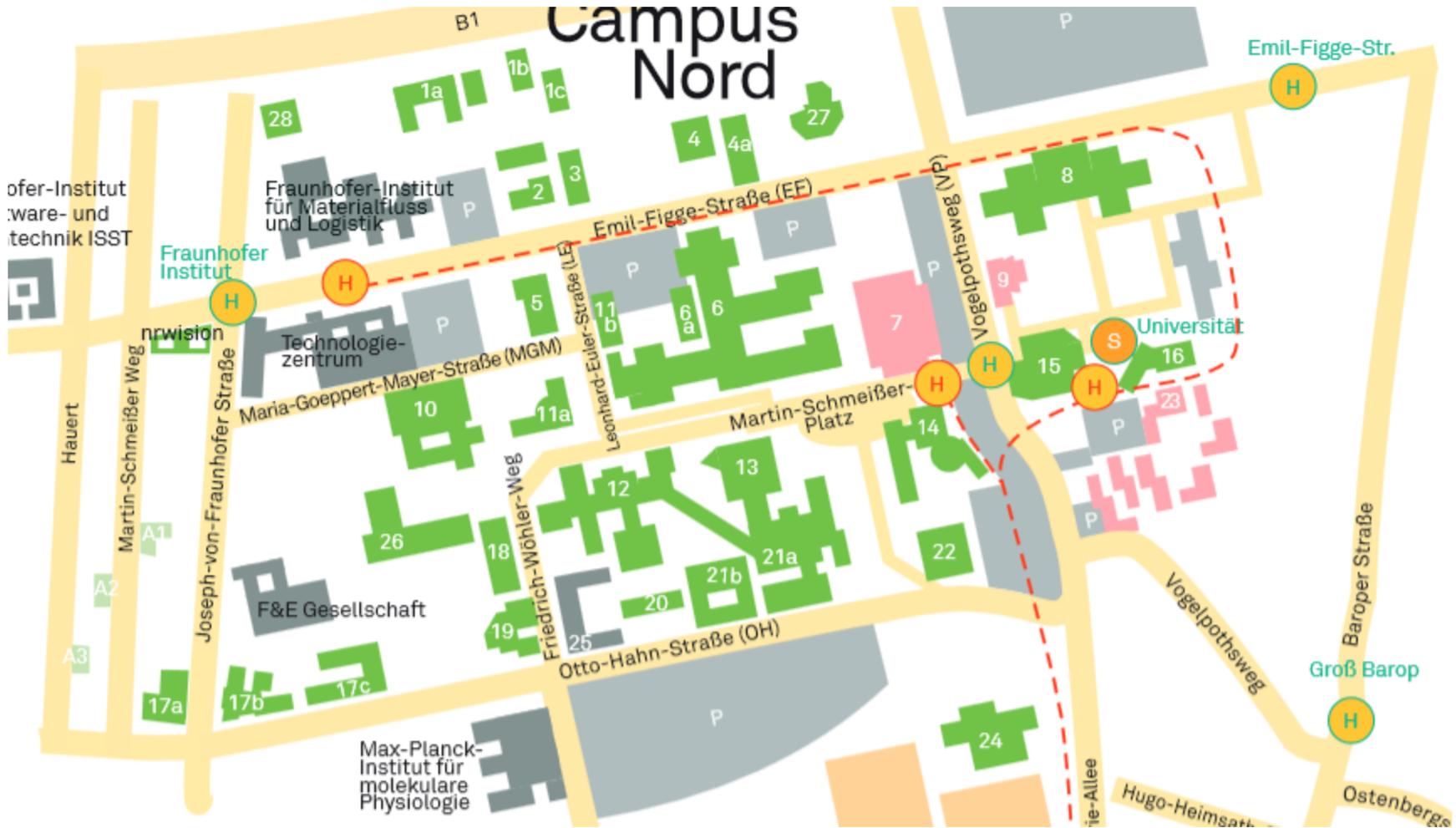
1.1 Vorlesung

Vorlesung Rechnerstrukturen (4V + 2Ü)

- Informationen zur Vorlesung
 - <http://ls12-www.cs.tu-dortmund.de/daes/de/lehre/lehrveranstaltungen/wintersemester-20192020/rechnerstrukturen.html>
- bei Fragen **Fragen!**
 - (in der Vorlesung),
 - **nach der Vorlesung**,
 - **in der Übung** oder
 - wenn sehr dringend: jian-jia.chen@cs.uni-dortmund.de
- Jian-Jia Chen, Otto-Hahn-Str. 16, Raum E23, Sprechstunde Mo. 11-12

1.1 Vorlesung

Lageplan



1. Organisatorisches

1. Organisatorisches

1. Vorlesung ✓

2. **Übungen**

3. Übungsanmeldung

4. HelpDesk

5. Klausur

2. Einleitung

3. Repräsentation von Daten

1.2 Übungen

Übungen zur Vorlesung **Rechnerstrukturen** (4V + 2Ü)

- Veranstalter
 - Marco Dürr
 - Roland Kühn
 - Ulrich Gabor
 - studentische Tutorinnen und Tutoren
- Anmeldung
 - erfolgt web-basiert über ASSESS (Details folgen)
- Teilnahme
 - **Regelmäßige** und **aktive** Teilnahme sehr wichtig!
 - *Übungsschein erforderlich zur Klausurteilnahme (Studienleistung)*

1.2 Übungen

Übungen zur Vorlesung **Rechnerstrukturen** (4V + 2Ü)

	Montag	Dienstag	Mittwoch	Donnerstag	Freitag
08 – 10		2 Gruppen			
10 – 12	5 Gruppen	3 Gruppen			
12 – 14	4 Gruppen			5 Gruppen	
14 – 16	7 Gruppen	4 Gruppen			
16 – 18	Vorlesung		Vorlesung		

1.2 Übungen

Studienleistung zur Vorlesung Rechnerstrukturen (4V + 2Ü)

- Anforderungen für erfolgreiche Übungsteilnahme
 - regelmäßige Teilnahme
 - erfolgreiche und regelmäßige Bearbeitung der Übungsaufgaben
 - ≥ 30 Punkte der Übungsblätter 1–4
 - ≥ 30 Punkte der Übungsblätter 5–8
 - ≥ 30 Punkte der Übungsblätter 9–12
- Hinweis Gruppenabgaben **möglich**
 - Gruppengröße ≤ 3

1. Organisatorisches

1. Organisatorisches

1. Vorlesung ✓

2. Übungen ✓

3. **Übungsanmeldung**

4. HelpDesk

5. Klausur

2. Einleitung

3. Repräsentation von Daten

1.3 Übungsanmeldung

Übungsanmeldung zur Vorlesung Rechnerstrukturen (4V + 2Ü)

1. In ASSESS *Neuen Account erstellen* wählen
2. Daten eintragen (insbes. Matrikelnummer, Name, **Email!**)
3. Mit neuem Account im ASSESS anmelden
4. Bei *Rechnerstrukturen* Option *anzeigen* wählen
5. Prioritäten für ≥ 4 Übungsgruppen-Slots angeben
6. Für Gruppenanmeldungen (max. 3 Studierende)
 1. *Clique* erstellen,
 2. *ID* und *Password* an andere Mitglieder weitergeben

Anmeldefrist

- Anmeldung bis zum **14.10.2019 20:00**
- Angabe von mind. 4 Prioritäten notwendig
- weitere Informationen: siehe Web-Seite

Benachrichtigung über Übungszuteilung per Email

1.3 Übungsanmeldung

Übungsanmeldung zur Vorlesung Rechnerstrukturen (4V + 2Ü)

The screenshot shows a web browser window with the address bar containing `ess.cs.uni-dortmund.de/ASSESS/`. The page header includes the logos for TU Dortmund, the Faculty of Informatics (fi), and ESS (Eingebettete Systemsoftware). A search bar and a 'Los' button are visible in the top right. The main content area is titled 'AnmeldungSSystem für ESS' and contains a breadcrumb trail: 'Sie sind hier: TU Dortmund > Informatik > LS12 > ESS > ASSESS'. On the left, there is a navigation menu with links for 'ASSESS', 'Übungsanmeldung', 'Abgabe', 'Profil', and 'Administration'. The main section is titled 'AsSESS' and includes a sub-header '(Probleme mit AsSESS, Verbesserungsvorschläge, Lob/Kritik? Mail an [die ASSESS-Admins!](#))'. Below this is a 'LOGIN' section with two input fields for 'Matrikelnr.' and 'Passwort', and a 'Login' button. At the bottom of the login section, there are links for 'Noch kein Account? [Neuen Account erstellen](#) | [Passwort vergessen?](#)'. The footer of the page indicates 'ASSESS v1.5.37 — Team: [Horst Schirmeier](#), Alexander Lochmann, Michael Lenz' and a timestamp 'Zuletzt geändert am 04.10.2012 11:17:32'.

1.3 Übungsanmeldung

Übungsanmeldung zur Vorlesung Rechnerstrukturen (4V + 2Ü)

The screenshot shows a web browser window with the URL `ess.cs.uni-dortmund.de/ASSESS/?do=createacc_show`. The page header includes the logos for TU Dortmund, the Faculty of Informatics (fi), and ESS (Eingebettete Systemssoftware). The main content area is titled "Anmeldungssystem für ESS" and shows the breadcrumb "Sie sind hier: TU Dortmund > Informatik > LS12 > ESS > ASSESS". The page is for "AsSESS" and includes a link for "ACCOUNT ERSTELLEN". A sidebar on the left lists navigation options: "ASSESS", "Übungsanmeldung", "Abgabe", "Profil", and "Administration". The main form area contains the following fields and options:

- Matrikelnr.:
- ODER
- Hast du noch keine Matrikelnummer? (weil du z.B. Schüler-Student bist) Pseudo-Matrikelnummer generieren
- Anrede: keine Angabe
- Vorname:*
- Nachname:*
- eMailadresse:
- FRLPoolKennung: (falls vorhanden)

1.3 Übungsanmeldung

Übungsanmeldung zur Vorlesung Rechnerstrukturen (4V + 2Ü)

The screenshot shows a web browser window with the URL `ess.cs.uni-dortmund.de/ASSESS/index.php?do=lecturelist`. The page header includes the logos for TU Dortmund, the Faculty of Informatics (fi), and ESS (Eingebettete Systemsoftware). A search bar is visible in the top right.

The main content area is titled "Anmeldungssystem für ESS" and shows the breadcrumb "Sie sind hier: TU Dortmund > Informatik > LS12 > ESS > ASSESS". A sidebar on the left contains navigation links: "ASSESS", "Übungsanmeldung", "Abgabe", "Profil", and "Administration".

The main content area features the "AsSESS" logo and a note: "(Probleme mit AsSESS, Verbesserungsvorschläge, Lob/Kritik? Mail an [die ASSESS-Admins!](#))". Below this is the section "LEHRVERANSTALTUNGEN" which contains a table of courses.

Name	Veranstalter	# Übungen	
Funktionale Programmierung WS 2012/2013 (Website)	Pascal Hof	8	anzeigen
Rechnerstrukturen WS2012/13 (Website)	Peter Marwedel	24	anzeigen
Seminar 'Softwarebasierte Fehlertoleranz' WS 2012/13 (Website)	Olaf Spinczyk	1	anzeigen
Betriebssystembau WS 2012/13 (Website)	Olaf Spinczyk	2	anzeigen
Übersetzerbau WS 2012/13 (Website)	Jens Lechner	4	anzeigen

1.3 Übungsanmeldung

Übungsanmeldung zur Vorlesung Rechnerstrukturen (4V + 2Ü)

The screenshot shows a web browser window with the URL `ess.cs.uni-dortmund.de/ASSESS/index.php?do=exerciselist&lectur`. The page header includes the logos for TU Dortmund, the Faculty of Informatics (fi), and ESS (Eingebettete Systemsoftware). The main content area is titled "Anmeldungssystem für ESS" and shows the user is logged in as Timon Kelter. The page displays a list of exercises for the course "RECHNERSTRUKTUREN WS2012/13".

Sie sind hier: TU Dortmund > Informatik > LS12 > ESS > ASSESS

ASSESS

- Übungsanmeldung
- Abgabe
- Profil
- Administration

AsSESS

(Probleme mit AsSESS, Verbesserungsvorschläge, Lob/Kritik? Mail an [die ASSESS-Admins!](#))

Eingeloggt als **Timon Kelter** (Mat.-Nr. 10071; Administrator). [Logout](#)

ÜBUNG ZU "RECHNERSTRUKTUREN WS2012/13"

Es sind mindestens 3 Prioritäten abzugeben. Ende der Prioritätenphase: 11.10.2012, 10:00:00

Die Spalte "Beliebtheit" gibt Auskunft über die Anzahl an Priorität-1-Anmeldungen für diese Übung.

Jedes Mitglied einer Clique kann die Prioritäten verändern.

Übung	Übungsgruppenleiter	Priorität	Beliebtheit (# Prio 1)
Mo. 10-12 (MSW16, E30, Gruppen-Nr. 1)	Timon Kelter	1 - höchste	0
Mo. 10-12 (MSW16, E32, Gruppen-Nr. 3)	Timon Kelter	1 - höchste	0
Mo. 10-12 (MSW16, E31, Gruppen-Nr. 5)	Timon Kelter	1 - höchste	0
Mo. 12-14 (MSW16, E31, Gruppen-Nr. 9)	Timon Kelter	1 - höchste	0

1.3 Übungsanmeldung

Übungsanmeldung zur Vorlesung Rechnerstrukturen (4V + 2Ü)

Gruppen-Nr. 4)	Timon Kelter	1 - höchste	
Do. 12-14 (MSW16, E30, Gruppen-Nr. 6)	Timon Kelter	1 - höchste	3
Do. 12-14 (MSW16, E31, Gruppen-Nr. 14)	Timon Kelter	1 - höchste	2
Do. 12-14 (MSW16, E29, Gruppen-Nr. 22)	Timon Kelter	1 - höchste	2

Um eine Clique anzulegen, gib unten ein Passwort für deine Clique ein und klicke auf "Anlegen". Anschließend wird dir in dieser Übersicht die ID deiner Clique angezeigt. Die ID sowie das Passwort gibst du nun den gewünschten Kommilitonen, damit diese der Clique beitreten können.

Möchtest du einer Clique beitreten, so gib unten die ID und das Passwort, die du von deinem Kommilitonen erhalten hast, ein. Bitte beachte, dass du mit dem Eintreten in eine Clique die Prioritäten dieser übernimmst.

Clique anlegen	Passwort: <input type="password"/>	<input type="button" value="Anlegen"/>
Clique beitreten ID: <input type="text" value="10071"/>	Passwort: <input type="password"/>	<input type="button" value="Beitreten"/>

[Zurück zu den Lehrveranstaltungen](#)

ASSESS v1.5.37 — Team: [Horst Schirmeier](#), Alexander Lochmann, Michael Lenz

Zuletzt geändert am 04.10.2012 11:17:32.

1.3 Zusätzliche Informationen

SchülerUni

- Email to Marco Dürr

Wichtige Information

- Erste Übung
 - **Ausgabe: nach der Vorlesung, 09.10.19 (Mittwoch)**
 - **Deadline: 16:00, 16.10.19 (Mittwoch)**
 - **Besprechungszeitraum: 21.10.-25.10.19**
 - **Gruppenverteilung: 14.10.19 (Montag)**

1. Organisatorisches

1. Organisatorisches

1. Vorlesung ✓
2. Übungen ✓
3. Übungsanmeldung ✓
4. HelpDesk
5. Klausur

2. Einleitung

3. Repräsentation von Daten

Der Rechnerstrukturen HelpDesk:

- Offene Sprechstunden
- Offener Lernraum
- Unterstützt von Tutorinnen/Tutoren
- Hilfe bei:
 - Aufgaben/Übungszetteln
 - Klausurvorbereitung
 - Fragen zum Stoff

RS HelpDesk - Öffnungszeiten

Ab 14.10.2019

	Montag	Dienstag	Mittwoch	Donnerstag	Freitag
08 – 10					
10 – 12	Help Desk	Help Desk	Help Desk	Help Desk	
12 – 14	Help Desk	Help Desk	Help Desk	Help Desk	
14 – 16	Help Desk	Help Desk	Help Desk	Help Desk	
16 – 18	Vorlesung		Vorlesung		

Lageplan



Oh12, 3. Stock bei den Aufzügen

1. Organisatorisches

1. Organisatorisches

1. Vorlesung ✓
2. Übungen ✓
3. Übungsanmeldung ✓
4. HelpDesk ✓
5. Klausur

2. Einleitung

3. Repräsentation von Daten

1.4 Klausur

- Übungsschein erforderlich zur Klausurteilnahme d.h. mindestens 30 Punkte in jedem der drei Blöcke sammeln
- 1. Klausur: Dienstag 26.02.2020 08:00-10:00
- 2. Klausur: Nach dem Sommersemester 2020, Termin wird noch bekanntgegeben
- Die Nettozeit für die Klausur beträgt jeweils 120 Minuten
- Die Anmeldung erfolgt zum Ende des Semesters für Studierende im Studiengang Informatik und Studiengang Angewandte Informatik über das BOSS-System

Übersicht

1. Organisatorisches
2. Einleitung
3. Repräsentation von Daten
4. Boolesche Funktionen und Schaltnetze
5. Rechnerarithmetik
6. Optimierung von Schaltnetzen
7. Programmierbare Bausteine
8. Synchrone Schaltwerke

2. Einleitung

1. Organisatorisches ✓

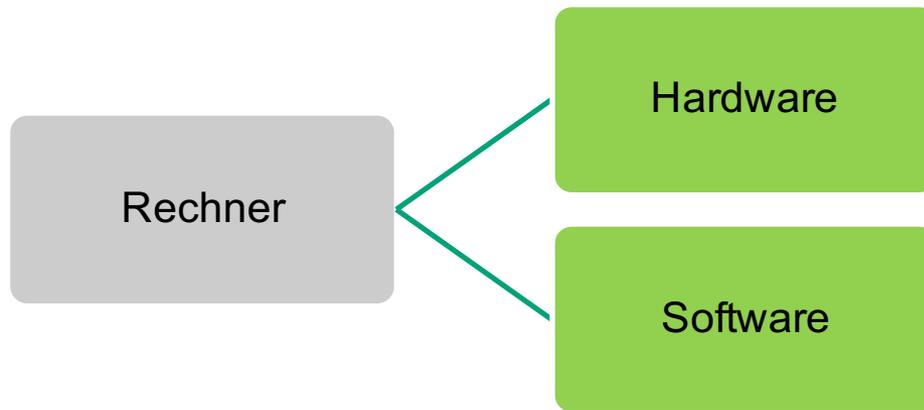
2. Einleitung

1. Historische Entwicklung von Rechnern (Anhang)

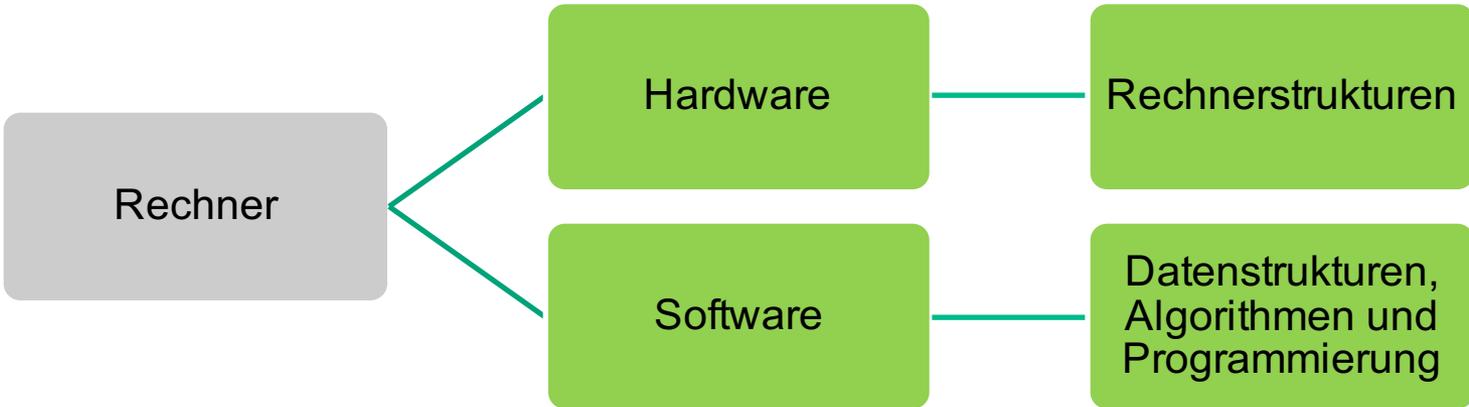
2. Einordnung

3. Repräsentation von Daten

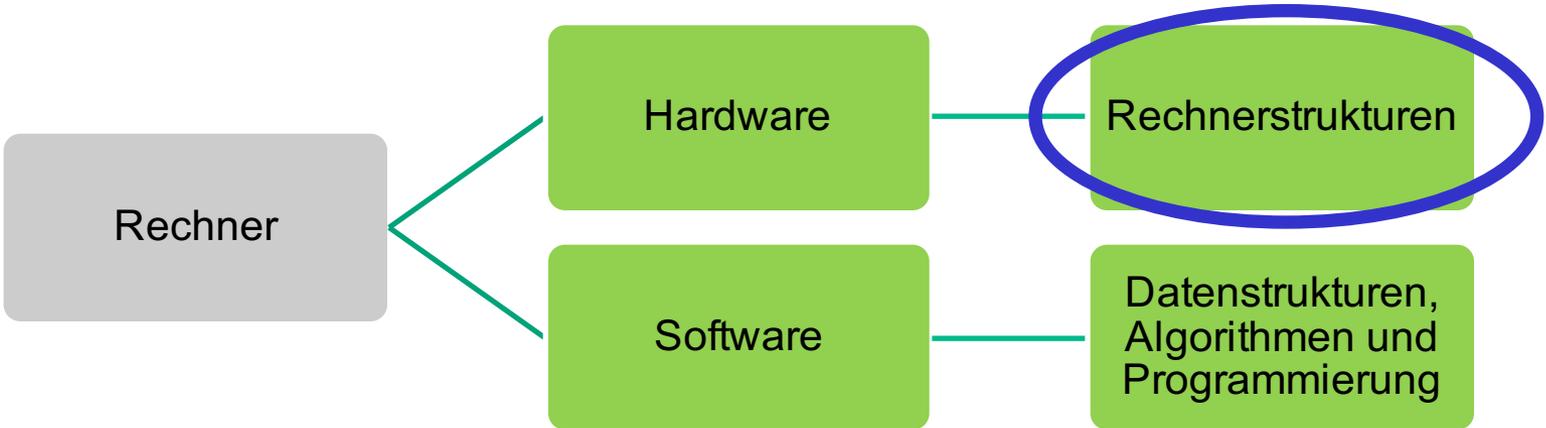
2.2 Einordnung



2.2 Einordnung

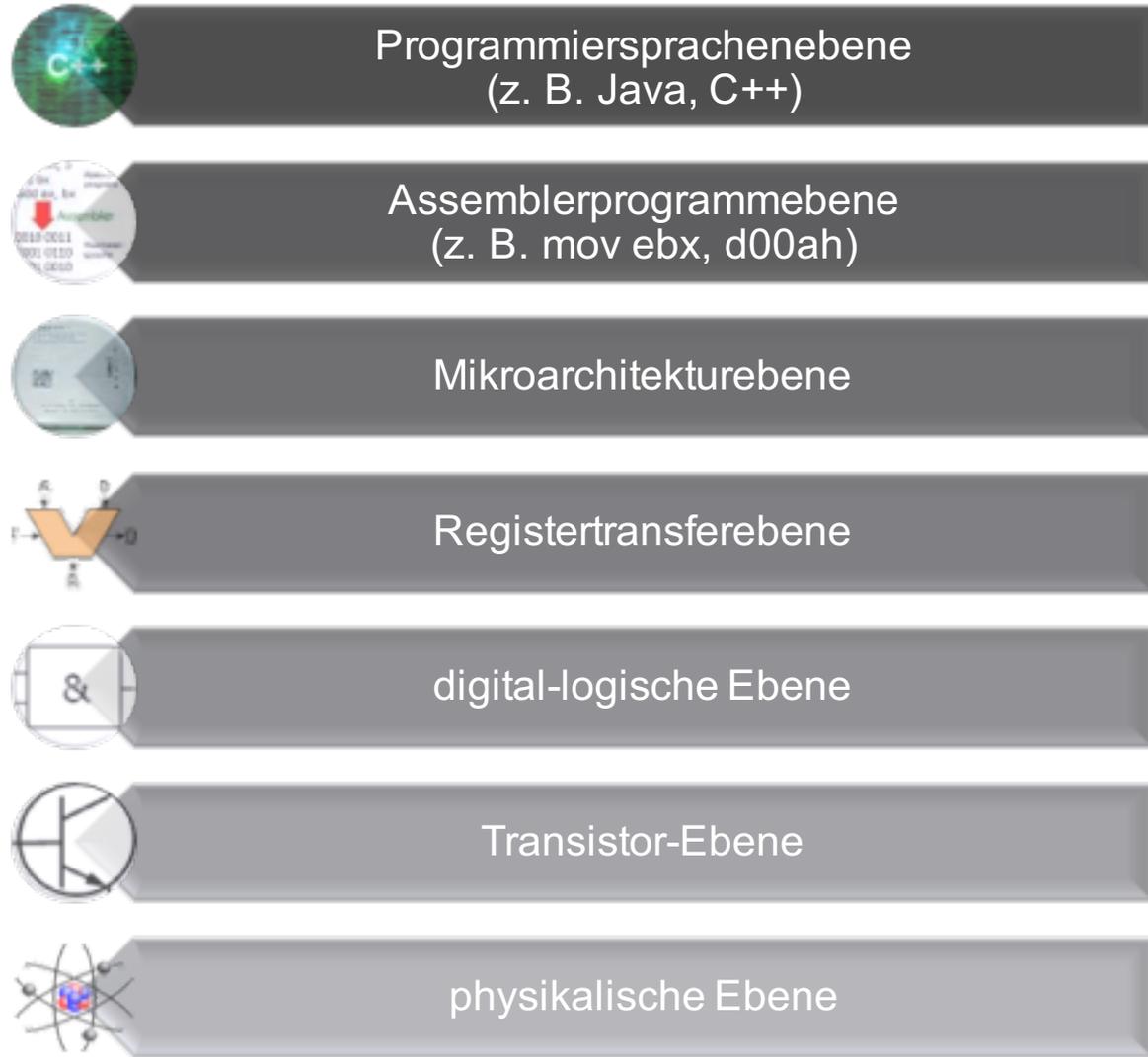


2.2 Einordnung



2.2 Einordnung

Hardware: Abstraktionsebenen



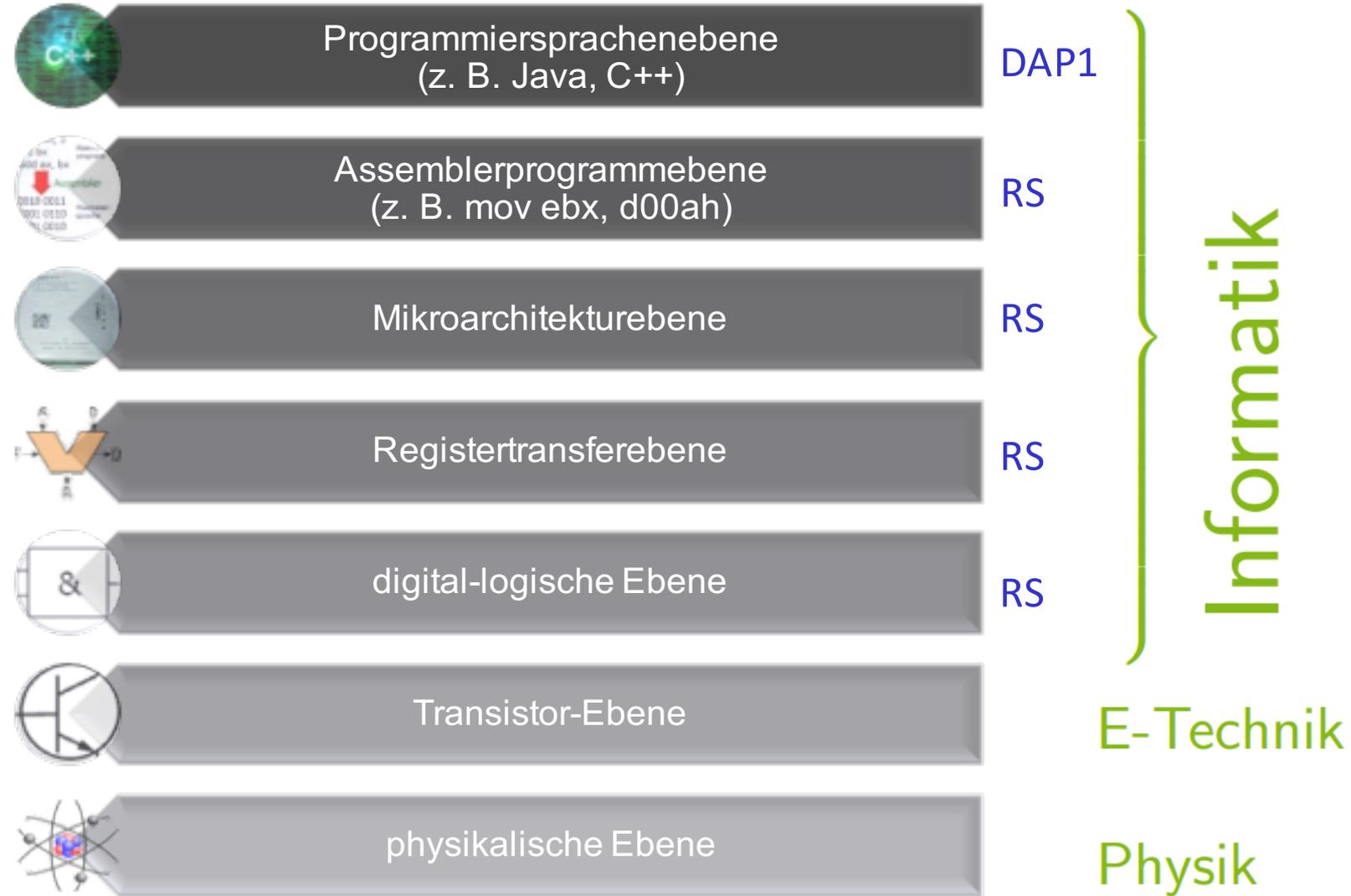
Informatik

E-Technik

Physik

2.2 Einordnung

Hardware: Abstraktionsebenen



3. Repräsentation von Daten

1. Organisatorisches ✓

2. Einleitung ✓

3. Repräsentation von Daten

1. Repräsentation von natürlichen Zahlen

3.1 Repräsentation von natürlichen Zahlen

Moderne Rechner arbeiten binär (→ Digitalrechner)

Grund

- technisch einfacher zu realisieren
- damit billiger zu realisieren
- bieten wohl-definierte Genauigkeit
- sind leichter zu programmieren

Problem

Wie repräsentiert man mit nur zwei Grundzuständen...

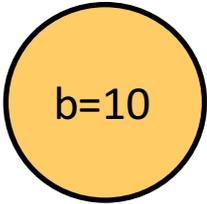
- Zahlen? ... natürliche, ganze, reelle Zahlen?
- Texte?
- andere Daten?

3.1 Repräsentation von natürlichen Zahlen

Stellenwertsysteme

- 10 Ziffern $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ — Basis 10
- Muss das so sein?
- Behauptung:
 - Es geht mit jeder Basis $b \in \mathbb{N} \setminus \{1\}$
- Basis b
 - Ziffern $\{0, 1, \dots, b - 1\}$
 - Schreibweise $(z_{\ell-1} z_{\ell-2} \dots z_1 z_0)_b$
 - Beispiel $(2\ 0\ 1\ 4)_{10} = 2014$
 - bei Basis $b > 10$ reichen unsere Zahlzeichen nicht
 - A für 10,
 - B für 11,
 - ...

3.1 Repräsentation von natürlichen Zahlen



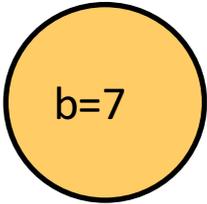
Umrechnung von Basis 10 in Basis 10

$$\begin{aligned}(3\ 7\ 4\ 8)_{10} &= 3 \times 10^3 + 7 \times 10^2 + 4 \times 10^1 + 8 \times 10^0 \\ &= 3 \times 1000 + 7 \times 100 + 4 \times 10 + 8 \times 1 \\ &= 3000 + 700 + 40 + 8 \\ &= 3748\end{aligned}$$

Stelle	3	2	1	0
Basis ^ Stelle	10^3	10^2	10^1	10^0
Wert der Stelle	1000	100	10	1
Ziffer	3	7	4	8
Wert	3000	700	40	8

3.1 Repräsentation von natürlichen Zahlen

Umrechnung von Basis 7 in Basis 10

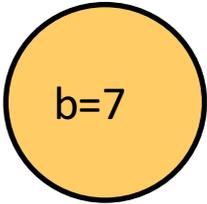


$(3\ 7\ 4\ 8)_7$

Bei einer Basis b ist der Ziffernvorrat: $\{0, 1, \dots, b - 1\}$

Stelle	3	2	1	0
Basis [^] Stelle				
Wert der Stelle				
Ziffer				
Wert				

3.1 Repräsentation von natürlichen Zahlen

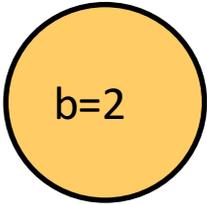


Umrechnung von Basis 7 in Basis 10

$$\begin{aligned}(2\ 1\ 6\ 4)_7 &= 2 \times 7^3 + 1 \times 7^2 + 6 \times 7^1 + 4 \times 7^0 \\ &= 2 \times 343 + 1 \times 49 + 6 \times 7 + 4 \times 1 \\ &= 686 + 49 + 42 + 4 \\ &= 781\end{aligned}$$

Stelle	3	2	1	0
Basis ^ Stelle	7^3	7^2	7^1	7^0
Wert der Stelle	343	49	7	1
Ziffer	2	1	6	4
Wert	686	49	42	4

3.1 Repräsentation von natürlichen Zahlen



Umrechnung von Basis 2 in Basis 10

$$\begin{aligned}(1011)_2 &= 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 \\ &= 1 \times 8 + 0 \times 4 + 1 \times 2 + 1 \times 1 \\ &= 8 + 0 + 2 + 1 \\ &= 11\end{aligned}$$

Stelle	3	2	1	0
Basis ^ Stelle	2^3	2^2	2^1	2^0
Wert der Stelle	8	4	2	1
Ziffer	1	0	1	1
Wert	8	0	2	1

3.1 Repräsentation von natürlichen Zahlen

Besonderheit der Basis $b=16$

$$(0)_{10} = (0)_{16} = (0000)_2$$

$$(1)_{10} = (1)_{16} = (0001)_2$$

$$(2)_{10} = (2)_{16} = (0010)_2$$

$$(3)_{10} = (3)_{16} = (0011)_2$$

$$(4)_{10} = (4)_{16} = (0100)_2$$

$$(5)_{10} = (5)_{16} = (0101)_2$$

$$(6)_{10} = (6)_{16} = (0110)_2$$

$$(7)_{10} = (7)_{16} = (0111)_2$$

$$(8)_{10} = (8)_{16} = (1000)_2$$

$$(9)_{10} = (9)_{16} = (1001)_2$$

$$(10)_{10} = (A)_{16} = (1010)_2$$

$$(11)_{10} = (B)_{16} = (1011)_2$$

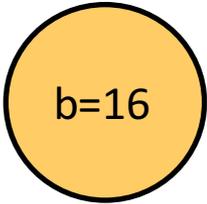
$$(12)_{10} = (C)_{16} = (1100)_2$$

$$(13)_{10} = (D)_{16} = (1101)_2$$

$$(14)_{10} = (E)_{16} = (1110)_2$$

$$(15)_{10} = (F)_{16} = (1111)_2$$

3.1 Repräsentation von natürlichen Zahlen



Umrechnung von Basis 16 in Basis 10

$$\begin{aligned}(3 \text{ E } \text{ C } 9)_{16} &= 3 \times 16^3 + \text{E} \times 16^2 + \text{C} \times 16^1 + 9 \times 16^0 \\ &= 3 \times 4096 + 14 \times 256 + 12 \times 16 + 9 \times 1 \\ &= 12\,288 + 3\,584 + 192 + 9 \\ &= 16\,073\end{aligned}$$

Stelle	3	2	1	0
Basis ^ Stelle	16^3	16^2	16^1	16^0
Wert der Stelle	4096	256	16	1
Ziffer	3	14	12	9
Wert	12 288	3 584	192	9

3.1 Repräsentation von natürlichen Zahlen

Besonderheit der Basis $b=16$

schnelle Umwandlung Basis 2 \leftrightarrow Basis 16

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

$$(0010\ 1011\ 0100\ 1001\ 1111)_2$$
$$= (2\ B\ 4\ 9\ F)_{16}$$

3.1 Repräsentation von natürlichen Zahlen

Namenskonventionen

- Basis 10 **Dezimaldarstellung**
- Basis 2 **Binärdarstellung** (Ziffern nennen wir **Bits**)
- Basis 16 **Hexadezimaldarstellung** (eigentlich Sedezimaldarstellung)

Offensichtliche Beobachtung: Binärdarstellung für die Repräsentation natürlicher Zahlen im Rechner geeignet

3.1 Repräsentation von natürlichen Zahlen

Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$ // ℓ ist die Repräsentationslänge
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$

$\lfloor x \rfloor$: untere Gaußklammer, die größte ganze Zahl, die kleiner als oder gleich wie x ist

3.1 Repräsentation von natürlichen Zahlen

Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$

b	16				
n	38 835				
l					
n_0					
n_1					
n_2					
n_3					

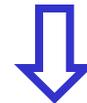
3.1 Repräsentation von natürlichen Zahlen

Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$



1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$

b	16				
n	38 835				
l	-1				
n_0					
n_1					
n_2					
n_3					

3.1 Repräsentation von natürlichen Zahlen

Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$



b	16				
n	38 835				
l	-1				
n_0					
n_1					
n_2					
n_3					

3.1 Repräsentation von natürlichen Zahlen

Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$



b	16	16			
n	38 835	38 835			
l	-1	0			
n_0					
n_1					
n_2					
n_3					

3.1 Repräsentation von natürlichen Zahlen

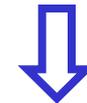
Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$



b	16	16			
n	38 835	38 835			
l	-1	0			
n_0		3			
n_1					
n_2					
n_3					

3.1 Repräsentation von natürlichen Zahlen

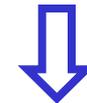
Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$



b	16	16			
n	38 835	2 427			
l	-1	0			
n_0		3			
n_1					
n_2					
n_3					

3.1 Repräsentation von natürlichen Zahlen

Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$



b	16	16			
n	38 835	2 427			
l	-1	0			
n_0		3			
n_1					
n_2					
n_3					

3.1 Repräsentation von natürlichen Zahlen

Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$



b	16	16	16		
n	38 835	2 427	2 427		
l	-1	0	1		
n_0		3	3		
n_1					
n_2					
n_3					

3.1 Repräsentation von natürlichen Zahlen

Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$



b	16	16	16		
n	38 835	2 427	2 427		
l	-1	0	1		
n_0		3	3		
n_1			B		
n_2					
n_3					

3.1 Repräsentation von natürlichen Zahlen

Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$



b	16	16	16		
n	38 835	2 427	151		
l	-1	0	1		
n_0		3	3		
n_1			B		
n_2					
n_3					

3.1 Repräsentation von natürlichen Zahlen

Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$



b	16	16	16		
n	38 835	2 427	151		
l	-1	0	1		
n_0		3	3		
n_1			B		
n_2					
n_3					

3.1 Repräsentation von natürlichen Zahlen

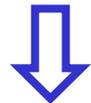
Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$



b	16	16	16	16	
n	38 835	2 427	151	151	
l	-1	0	1	2	
n_0		3	3	3	
n_1			B	B	
n_2					
n_3					

3.1 Repräsentation von natürlichen Zahlen

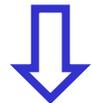
Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$



b	16	16	16	16	
n	38 835	2 427	151	151	
l	-1	0	1	2	
n_0		3	3	3	
n_1			B	B	
n_2				7	
n_3					

3.1 Repräsentation von natürlichen Zahlen

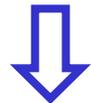
Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$



b	16	16	16	16	
n	38 835	2 427	151	9	
l	-1	0	1	2	
n_0		3	3	3	
n_1			B	B	
n_2				7	
n_3					

3.1 Repräsentation von natürlichen Zahlen

Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$



b	16	16	16	16	
n	38 835	2 427	151	9	
l	-1	0	1	2	
n_0		3	3	3	
n_1			B	B	
n_2				7	
n_3					

3.1 Repräsentation von natürlichen Zahlen

Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$



b	16	16	16	16	16
n	38 835	2 427	151	9	9
l	-1	0	1	2	3
n_0		3	3	3	3
n_1			B	B	B
n_2				7	7
n_3					

3.1 Repräsentation von natürlichen Zahlen

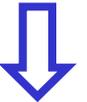
Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$



b	16	16	16	16	16
n	38 835	2 427	151	9	9
l	-1	0	1	2	3
n_0		3	3	3	3
n_1			B	B	B
n_2				7	7
n_3					9

3.1 Repräsentation von natürlichen Zahlen

Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$



b	16	16	16	16	16
n	38 835	2 427	151	9	0
l	-1	0	1	2	3
n_0		3	3	3	3
n_1			B	B	B
n_2				7	7
n_3					9

3.1 Repräsentation von natürlichen Zahlen

Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$



b	16	16	16	16	16
n	38 835	2 427	151	9	0
l	-1	0	1	2	3
n_0		3	3	3	3
n_1			B	B	B
n_2				7	7
n_3					9

3.1 Repräsentation von natürlichen Zahlen

Umrechnung von Basis 10 in Basis b

Algorithmus 1

Eingabe Basis $b \in \mathbb{N} \setminus \{1\}$, Betragszahl $n \in \mathbb{N}$

Ausgabe Repräsentation $(n_\ell n_{\ell-1} n_{\ell-2} \cdots n_1 n_0)_b$

1. $\ell := -1$
2. So lange $n > 0$ gilt
3. $\ell := \ell + 1$
4. $n_\ell := n - b \cdot \lfloor n / b \rfloor$
5. $n := \lfloor n / b \rfloor$

b	16	16	16	16	16
n	38 835	2 427	151	9	0
l	-1	0	1	2	3
n_0		3	3	3	3
n_1			B	B	B
n_2				7	7
n_3					9

$$\rightarrow (38\,835)_{10} = (9\,7\,B\,3)_{16}$$

3.1 Repräsentation von natürlichen Zahlen

b-adische Darstellung – Eigenschaften

Theorem

Für jede Basis $b \in \mathbb{N} \setminus \{1\}$ und jede natürliche Zahl $n \in \mathbb{N}$ lässt sich n eindeutig darstellen als

$$n = n_0 \cdot b^0 + n_1 \cdot b^1 + \dots + n_l \cdot b^l = \sum_{i=0}^l n_i \cdot b^i$$

mit $l \in \mathbb{N}_0$, $n_0, n_1, \dots, n_l \in \{0, 1, \dots, b-1\}$ und $n_l > 0$.

Fazit: Einschränkung bei binären Digitalrechnern in Bezug auf natürliche Zahlen stellt kein Problem dar.

3. Repräsentation von Daten

1. Organisatorisches ✓

2. Einleitung ✓

3. Repräsentation von Daten

1. Repräsentation von natürlichen Zahlen ✓

2. Repräsentation von Texten

3. Repräsentation ganzer Zahlen

4. Repräsentation rationaler Zahlen

5. Repräsentation anderer Daten

Historische Entwicklung von Rechnern (Anhang)

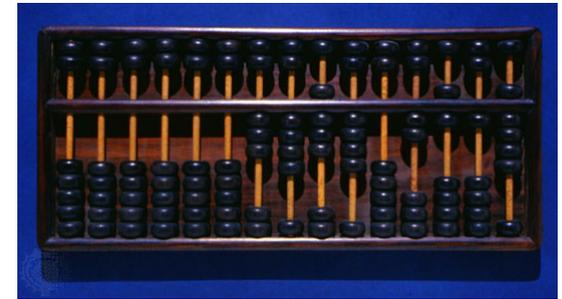
Zielsetzung

ursprünglich immer Automatisierung von Berechnungen

- schneller
- fehlerfreier

in der

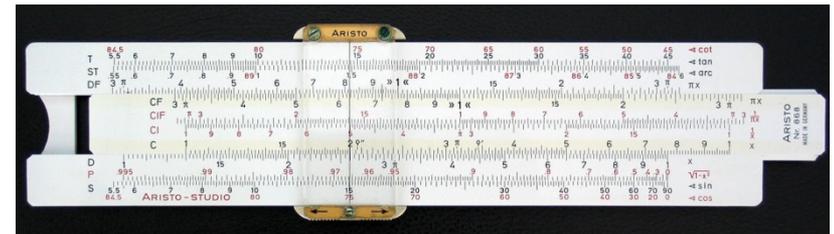
- Verwaltung,
- beim Militär,
- in der Wissenschaft



Quelle: Smithsonian Institution

Rechenhilfsmittel

- Abakus (ca. 200 n.Chr. In China)
- Rechenschieber (ca. 1620)



Quelle: Roger McLassus

Historische Entwicklung von Rechnern

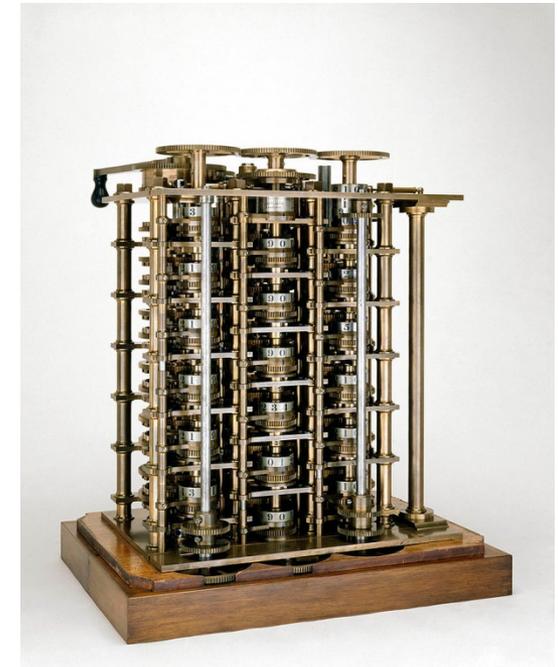
Mechanische Rechenmaschinen (1642-1945)

- *Blaise Pascal* (1623–1662)
Erste funktionsfähige Rechenmaschine für Addition/Subtraktion
- *Gottfried Wilhelm Leibniz* (1646–1716)
Rechenmaschine für 4 Grundrechenarten, Zahlen in Binärdarstellung



Quelle: Clemens Weller, Deutsches Museum München

- Charles Babbage (1791–1871)
 - Difference Engine (Add./Sub.),
 - Analytical Engine (programmierbar!)



Quelle: Science Museum, London

Historische Entwicklung von Rechnern

Mechanische Rechenmaschinen (1642-1945)

- *Herman Hollerith* (1860-1929)
Lochkarten zur Datenspeicherung
- *Konrad Zuse* (1910-1995)
Z1 (1937/38): erster programmgesteuerter Rechenautomat



Quelle: Adam Schuster, Wikipedia



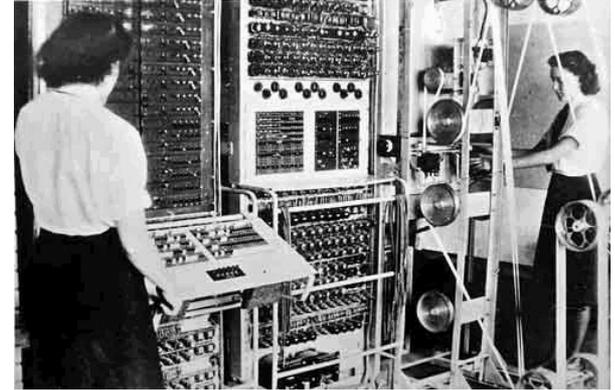
Quelle: Heinz Nixdorf MuseumsForum, Paderborn

Historische Entwicklung von Rechnern

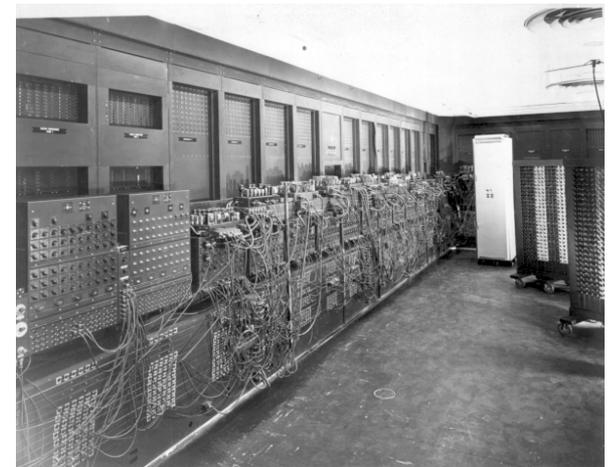
Die 1. Generation: Vakuumröhren (1945-1955)

- COLOSSUS (1943)
 - unter Mitwirkung von Alan Turing
 - Britisches Geheimprojekt zur
 - Entschlüsselung des ENIGMA-Codes

- ENIAC (1946)
erster elektronischer Universalrechner
 - 18000 Röhren,
 - 1500 Relais,
 - 30t Gewicht,
 - 174kW Leistungsaufnahme



Quelle: The National Archives , UK



Quelle: US Army

Historische Entwicklung von Rechnern

Die 1. Generation: Vakuumröhren (1945-1955)

- *John von Neumann*
Konzeption eines speicherprogrammierbaren Digitalrechners
 - Speicher,
 - Rechen-,
 - Steuer- und
 - E/A-Einheiten

- IBM 701 (1953)
 - erster wissenschaftlicher Rechner
 - arbeitete nach dem von-Neuman-Model



Quelle: TheFamousPeople.com



Quelle: IBM, USA

Historische Entwicklung von Rechnern

Die 2. Generation: Transistoren (1955-1965)

- TX-0 als erster transistor-basierter Rechner am M.I.T. entwickelt
- Digital Equipment Corporation gegründet (1957):
 - PDP-1: erster Minicomputer
 - PDP-8: verwendet erstmals ein Bus-System
- Control Data Corporation (CDC, Seymour Cray)
 - Modell 6600 (1964):
 - erster Supercomputer
 - 10-mal schneller als IBM-Top-Modell,
 - parallele Funktionseinheiten,
 - speziell für number crunching
- Burroughs B5000: Rechner
 - speziell für ALGOL 60



Quelle: ComputerHistory.org



Quelle: ComputerHistory.org

Historische Entwicklung von Rechnern

Die 3. Generation: Integrierte Schaltungen (1965-1980)

- IBM führt mit System/360 eine einzige Produktlinie ein
 - Mehrprogrammbetrieb,
 - Emulation anderer Rechner

- PDP-11
erfolgreichster Minicomputer im wissenschaftlichen Bereich
 - 16bit-Wortbreite
 - offenes Bussystem



Quelle: IBM, USA



Quellen: HighTechHistory.com

Historische Entwicklung von Rechnern

Die 4. Generation: VLSI-Integration (seit 1980)

höhere Integrationsdichte macht persönliche Rechner möglich

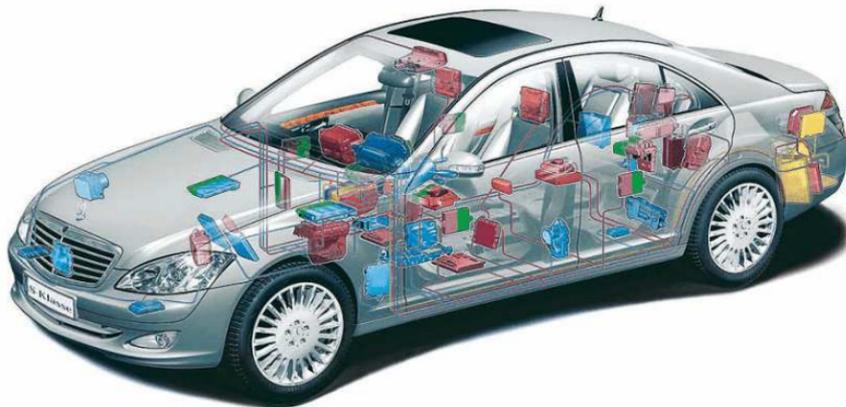
- Erster PC als Bausatz auf Basis des Intel 8080
- CP/M Betriebssystem für 8080-Rechner (diskettenbasiert)
- Steve Jobs & Steve Wozniak bauen den Apple
- IBM-PC auf Intel 8088-Basis mit Standardkomponenten (1981)
 - offener,
 - dokumentierter Standard wird von anderen Firmen geklont,
 - Betriebssystem MS-DOS [Rechte verbleiben bei Microsoft]
- Mitte der 80er Jahre: RISC-Rechner entstehen (SPARC, MIPS)
 - extrem kleiner Instruktionsspeicher,
 - Mikroprogrammebene wird abgeschafft

Historische Entwicklung von Rechnern

Pervasive / Ubiquitous Computing (heute)

extreme Integrationsdichte ermöglicht die Einbettung von Rechnern in quasi beliebige Artefakte des täglichen Lebens

- Mobiltelefone und persönliche Assistenten
- Kraftfahrzeuge
ca. 100 Steuergeräte [= Rechner] in Oberklasse-Fahrzeug



Quelle: freenet.de