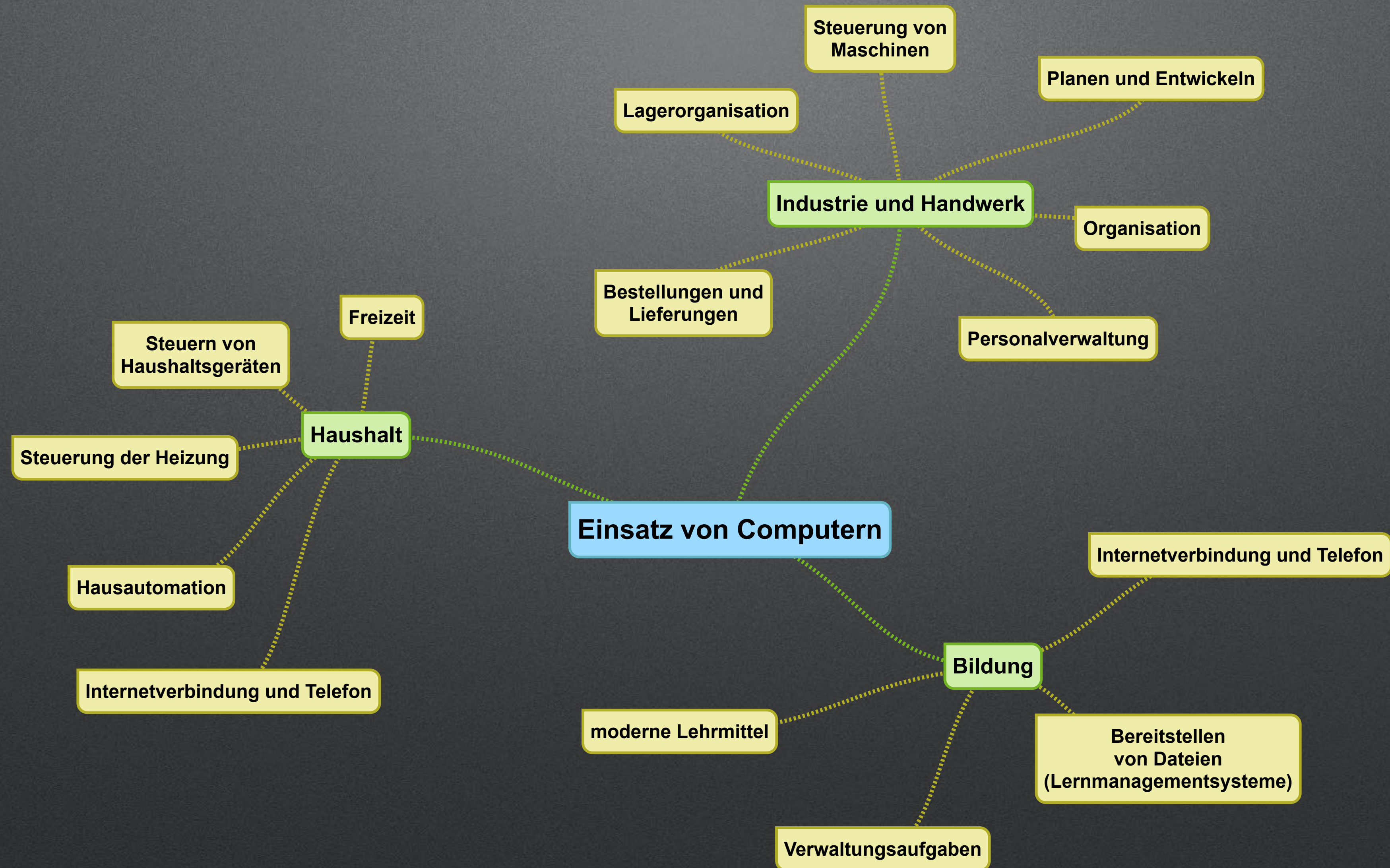


# 1 Computer verstehen

von Daten und Strukturen



# 1.1 Einsatzgebiete von Computern





# 1.2 Bauformen von Computern

- Mikroprozessoren zur Steuerung von Geräten für z.B. Internet of Things (Internet der Dinge) - Waage, Waschmaschine
- Kleinrechner, z.B. Raspberry Pi
- Smartphone
- Tablett PC
- Notebook oder Laptop
- eBook Reader
- Grafiktablett
- Desktop PC
- Gaming PC / Spielkonsolen
- Industrie PC, z.B. zur Verwaltung und Steuerung von Leitsystemen und Maschinen.
- Serversysteme, z.B. zur Verwaltung von Benutzern und Rechnern oder die Bereitstellung von Netzwerkdiensten.
- Großrechner und Superrechner, z.B. für die Verarbeitung riesiger Datenmengen in der Forschung



# 1.3 Der Computerarbeitsplatz

Hardware:

Monitor, Rechner, Tastatur, evtl. Lautsprecher, Kamera, Mikrofon

Peripheriegeräte/Zubehör:

Drucker, Plotter

Wichtige Schnittstellen:

HDMI (alt DVI, VGA) für Monitor, USB für Lautsprecher, Mikrofon und Kamera sowie externe Festplatten

Kabellose Schnittstellen:

Bluetooth zum Datenaustausch, Anschluss von Maus und Tastatur

Infrarot (veraltet) zum Datenaustausch mit anderen Geräten



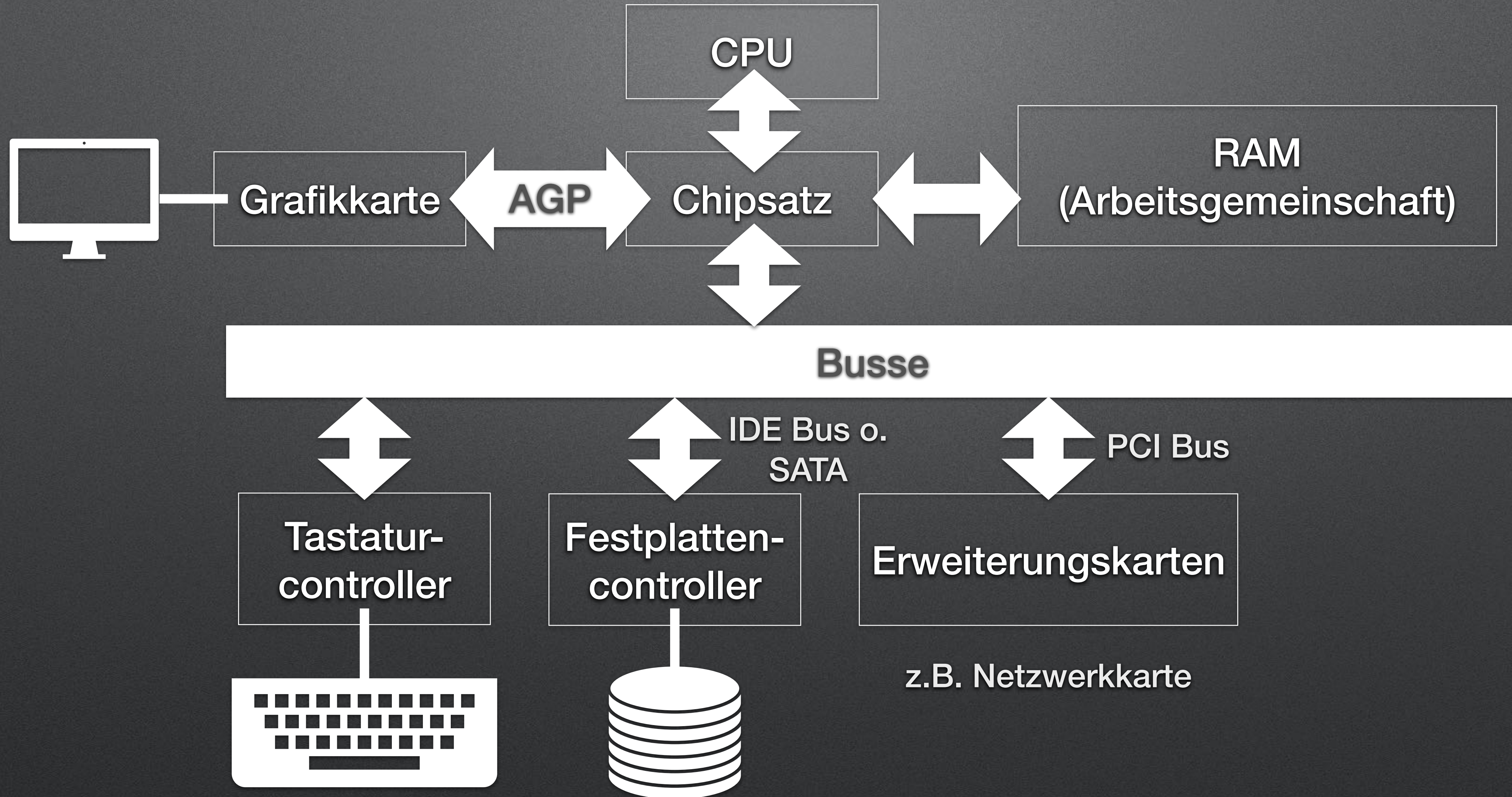
# 1.3 Der Computerarbeitsplatz

Wichtige interne Komponenten im Rechnergehäuse:

- Festplatte zur persistenten (dauerhaften) Datenspeicherung
- Arbeitsspeicher zur kurzzeitigen Speicherung von veränderbaren Programmdateien; kann Daten nur im eingeschalteten Zustand speichern
- Prozessor mit Rechen- und Steuerwerk
- Motherboard (Hauptplatine) mit Bussverbindungen zu den Komponenten
- Netzteil zur Versorgung mit elektrischer Energie



# 1.4 Funktionsweise eines Computers





# 1.4 Funktionsweise eines Computers

- Bus: mehrere Leitungen zum Transport von Daten und Steuersignalen, evtl. auch der Spannungsversorgung. Beispiel: SATA-Stecker und Leitung um Anschluss der Festplatte
- Grafikkarte: steuert in einem Computer die Grafikausgabe des Monitorbildes; berechnet evtl. 3D-Ausgaben
- Controller: steuern bestimmte Vorgänge der Ein- und Ausgabe von Daten, Beispiel: Der Tastaturcontroller steuert die Tastaturein- und ausgaben
- RAM (Random-Access Memory) ist ein flüchtiger Speicher für die Arbeit der CPU, der seinen Inhalt beim Ausschalten verliert.



# 1.5 Ergonomie (gesunde PC-Nutzung)

- Stuhl sollte drehbar und höhenverstellbar sein
  - Achte darauf: Füße stehen auf dem Boden oder einer Unterlage. Beine bilden ca. einen Winkel von  $90^\circ$ . Arme in einem Winkel  $>90^\circ$ .
- Tastatur sollte frei beweglich platzierbar sein.
  - Achte darauf: Gut leserliche Buchstaben ohne großen Kontrastunterschied. Die Handballen sollten vor der Tastatur aufliegen und die Tastatur sollte möglichst flach sein.



# 1.5 Ergonomie (gesunde PC-Nutzung)

- Monitor sollte in Höhe und Neigung verstellbar sein
  - Achte darauf: Die Oberkante des Monitors sollte nicht höher als deine Augen sein. Licht vom Fenster sollte seitwärts liegen (kein spiegeln oder blenden)
- Sitze nicht zu lange am PC
  - Achte darauf: Stehe immer mal auf und bewege dich.



# 1.6 Daten und ihre Darstellung im Computer

- Zahlen werden im Computer im binären Zahlensystem verarbeitet und gespeichert. z.B.  $1001 = 9$
- Buchstaben werden mit Hilfe von Zeichensätzen als Folge von binären Zahlen gespeichert. z.B.  $0100\ 0001 = A$



## 1.6.1 Zahlen im Rechner

### 1.6.1.1 Zahlensysteme

- Wir Menschen verwenden das Dezimalsystem (Zehnerzahlensystem) im Alltag. Dieses System besteht aus genau zehn Symbolen, welche als arabische Ziffern mit 0,1,2,3,4,5,6,7,8 und 9 dargestellt werden.
- Rechner verwenden das Binärsystem (Zweierzahlensystem). Dieses System verwendet nur die zwei Symbole 0 (LOW) und 1 (HIGH), welche sich im Rechner als „Spannung aus“ oder „Spannung ein“ gut darstellen lassen.



# 1.6.1.2 Wie zählt ein Rechner?

Zuerst wie wir im Alltag 0 für keins, dann 1 und nun? - Es braucht eine weitere Stelle, genau wie im Zehnersystem, wenn wir für den Nachfolger von 9 eine weitere Stelle links hinzufügen.

also:

binär	dezimal
0	0
1	1
10	2
11	3
100	4
101	5

binär	dezimal
110	6
111	7
1000	8
1001	9
1010	10
1011	11

binär	dezimal
1100	12
1101	13
1110	14
1111	15



# 1.6.1.3 Wir rechnen, wie ein PC

$$\begin{array}{r} 11 \\ + 1 \\ \hline 100 \end{array}$$

$$\begin{array}{r} 11 \\ + 10 \\ \hline 101 \end{array}$$

$$\begin{array}{r} 11 \\ + 11 \\ \hline 110 \end{array}$$

$$\begin{array}{r} 110011 \\ + 101010 \\ \hline 1011101 \end{array}$$

$$\begin{array}{r} 1110101 \\ + 101011 \\ \hline 10100000 \end{array}$$

$$\begin{array}{r} 11101 \\ + 11 \\ \hline 100000 \end{array}$$



# Übung

$$1+1=$$

$$1000 + 10 =$$

$$1010 + 10 =$$

$$110 + 11 =$$

$$111 + 1 =$$



# Übung

$$1+1=10$$

$$1000 + 10 = 1010$$

$$1010 + 10 = 1100$$

$$110 + 11 = 1001$$

$$111 + 1 = 1000$$



# 1.6.1.3 Wir rechnen, wie ein PC

$$\begin{array}{r} 11 \\ - 01 \\ \hline \end{array}$$

10

$$\begin{array}{r} 11 \\ - 10 \\ \hline \end{array}$$

01

$$\begin{array}{r} 11 \\ - 11 \\ \hline \end{array}$$

00

$$\begin{array}{r} 11101 \\ - 00011 \\ \hline \end{array}$$

11010

$$\begin{array}{r} 101101 \\ - 101000 \\ \hline \end{array}$$

000101

$$\begin{array}{r} 1101 \\ - 1010 \\ \hline \end{array}$$

0011



# Übung

$$111111 - 1 =$$

$$10110 - 11 =$$

$$11010 - 1101 =$$

$$1000011 - 11100 =$$

$$110011 - 101010 =$$



# Übung

$$111111 - 1 = 111110$$

$$10110 - 11 = 10011$$

$$11010 - 1101 = 1101$$

$$1000011 - 11100 = 100111$$

$$110011 - 101010 = 1001$$



# 1.6.2 Wie speichert ein Rechner Text?

- Den Textzeichen werden binäre Zahlenmuster zugeordnet (codiert).
- Eine wichtige Codierung eines Zeichenvorrates in einen Binärcode wurde 1963 als ASCII-Code definiert. (American Standard Code for Information Interchange, deutsch: Amerikanischer Standard-Code für den Informationsaustausch)
- Beispiel: Hallo

ASCII - Zeichensatz

binär (hex)	002 (0)	003 (1)	010 (2)	011 (3)	100 (4)	101 (5)	110 (6)	111 (7)
LSB	Steuerzeichen		Druckbuchstaben		Kleinbuchstaben			
0000 (0)	NUL	DLE	SP	@	P	-	p	
0001 (1)	SOH	DC1	!	A	Q	a	q	
0010 (2)	STX	DC2	"	B	R	b	r	
0011 (3)	ETX	DC3	#	C	S	c	s	
0100 (4)	EOT	DC4	\$	D	T	d	t	
0101 (5)	ENQ	NAK	%	E	U	e	u	
0110 (6)	ACK	CAN	&	F	V	f	v	
0111 (7)	BEL	ETB	'	G	W	g	w	
1000 (8)	BS	CAN	(	H	X	h	x	
1001 (9)	HAT	EM	)	I	Y	i	y	
1010 (A)	LF	SUB	*	J	Z	j	z	
1011 (B)	VT	ESC	+	K	[	k	[	
1100 (C)	FF	FS	,	L	\	l	\	
1101 (D)	CR	GS	-	M	]	m	]	
1110 (E)	SO	RS	.	N	^	n	^	
1111 (F)	SI	US	/	O	_	o	_	DEL

LSB ... least significant bit (niedrigwertige Bits)  
MSB ... most significant bit (hochwertige Bits)

einige Abkürzungen der Steuerzeichen  
CR ... carriage return, engl. Wagenrücklauf  
LF ... line feed, engl. Zeilenvorschub  
DEL ... delete, engl. Löschen

Abkürzungsbeispiele  
SP (Space, engl. Leerzeichen) == binär: 010 0000 oder hexadezimal: 20  
H == binär: 100 1000 oder hexadezimal: 48  
w == binär: 111 0111 oder hexadezimal: 77

Copyright © 2008, Pearson Education, Inc. Alle Rechte vorbehalten.

Komplexe Anwendungssysteme  
S. 1



1.

• D

• E

V

I

O

• E

	MSB ...most significant bit (hochwertige Bits)							
binär (hex)	000 (0)	001 (1)	010 (2)	011 (3)	100 (4)	101 (5)	110 (6)	111 (7)
LSB	Steuerzeichen				Großbuchstaben		Kleinbuchstaben	
0000 (0)	NUL	DLE	SP	0	@	P	`	p
0001 (1)	SOH	DC1	!	1	A	Q	a	q
0010 (2)	ATX	DC2	"	2	B	R	b	r
0011 (3)	ETX	DC3	#	3	C	S	c	s
0100 (4)	EOT	DC4	\$	4	D	T	d	t
0101 (5)	ENQ	NAK	%	5	E	U	e	u
0110 (6)	ACK	SYN	&	6	F	V	f	v
0111 (7)	BEL	ETB	'	7	G	W	g	w
1000 (8)	BS	CAN	(	8	H	X	h	x
1001 (9)	HAT	EM	)	9	I	Y	i	y
1010 (A)	LF	SUB	*	:	J	Z	j	z
1011 (B)	VT	ESC	+	;	K	[	k	{
1100 (C)	FF	FS	,	<	L	\	l	
1101 (D)	CR	GS	-	=	M	]	m	}
1110 (E)	SO	RS	.	>	N	^	n	~
1111 (F)	SI	US	/	?	O	_	o	DEL

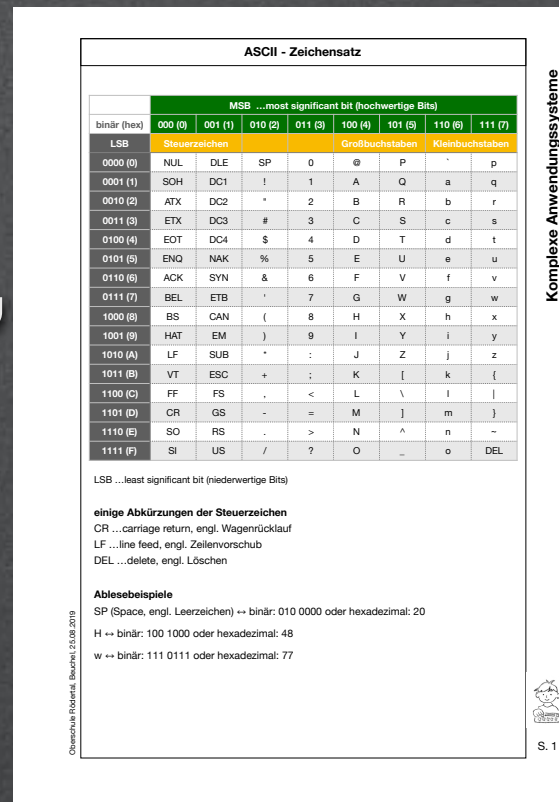
Komplexe Anwendungssysteme

?



# 1.6.2 Wie speichert ein Rechner Text?

- Den Textzeichen werden binäre Zahlenmuster zugeordnet (codiert).
- Eine wichtige Codierung eines Zeichenvorrates in einen Binärcode wurde 1963 als ASCII-Code definiert. (American Standard Code for Information Interchange, deutsch: Amerikanischer Standard-Code für den Informationsaustausch)
- Beispiel: Hallo



The image shows a snippet of the ASCII standard table. It lists characters from NUL (0) to DEL (127) in a grid. The columns represent different bit positions (0-7). Below the table, there are notes about control characters and abbreviations.

Binär (hex)	002 (2)	001 (1)	010 (2)	011 (3)	100 (4)	101 (5)	110 (6)	111 (7)
0000 (0)	NUL	DLE	SP	0	1	2	3	4
0001 (1)	SOH	DC1	!	1	A	G	a	h
0010 (2)	STX	DC2	"	2	B	H	b	i
0011 (3)	ETX	DC3	#	3	C	I	c	j
0100 (4)	EOT	DC4	\$	4	D	J	d	k
0101 (5)	ENQ	NAK	%	5	E	K	e	l
0110 (6)	ACK	CAN	&	6	F	L	f	m
0111 (7)	BEL	ETB	'	7	G	M	g	n
1000 (8)	BS	CAN	(	8	H	N	h	o
1001 (9)	HAT	EM	)	9	I	O	i	p
1010 (A)	LF	SUB	*	:	J	P	j	q
1011 (B)	VT	ESC	+	;	K	Q	k	r
1100 (C)	FF	FS	,	<	L	R	l	s
1101 (D)	CR	GS	-	=	M	S	m	t
1110 (E)	SO	RS	.	>	N	T	n	u
1111 (F)	SI	US	/	?	O	U	o	v

LSB ... least significant bit (rechterste Bit)  
MSB ... most significant bit (linkerster Bit)  
SP (Space, engl. Leerzeichen) = binär: 010 0000 oder hexadezimal: 20  
H = binär: 100 1000 oder hexadezimal: 48  
W = binär: 111 0111 oder hexadezimal: 77

einige Abkürzungen der Steuerzeichen  
CR ... carriage return, engl. Wagenrücklauf  
LF ... line feed, engl. Zeilenverschub  
DEL ... delete, engl. Löschen

Abkürzungen  
SP (Space, engl. Leerzeichen) = binär: 010 0000 oder hexadezimal: 20  
H = binär: 100 1000 oder hexadezimal: 48  
W = binär: 111 0111 oder hexadezimal: 77

Hallo = P1001000 P1101101 P1101100 P1101100 P1101111

P..Paritätsbit zur Fehlererkennung

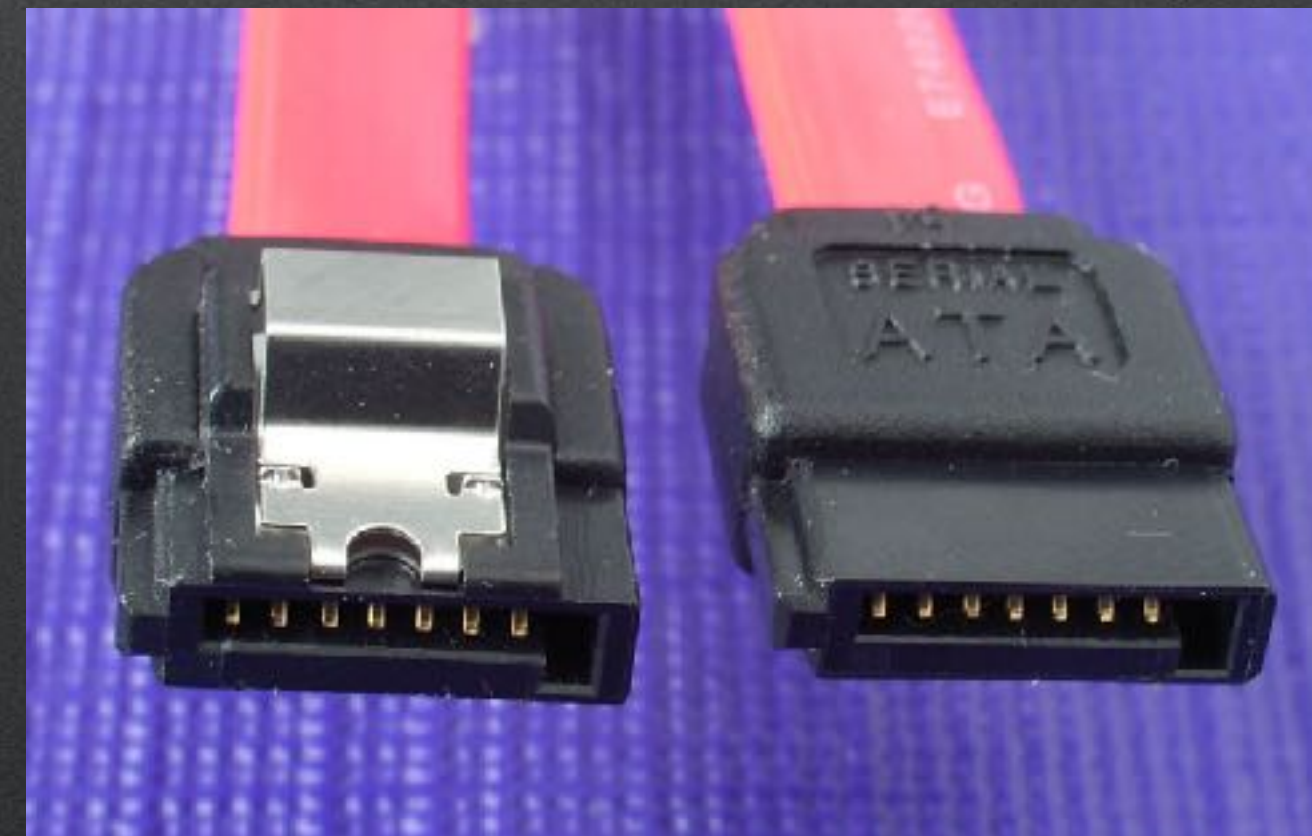
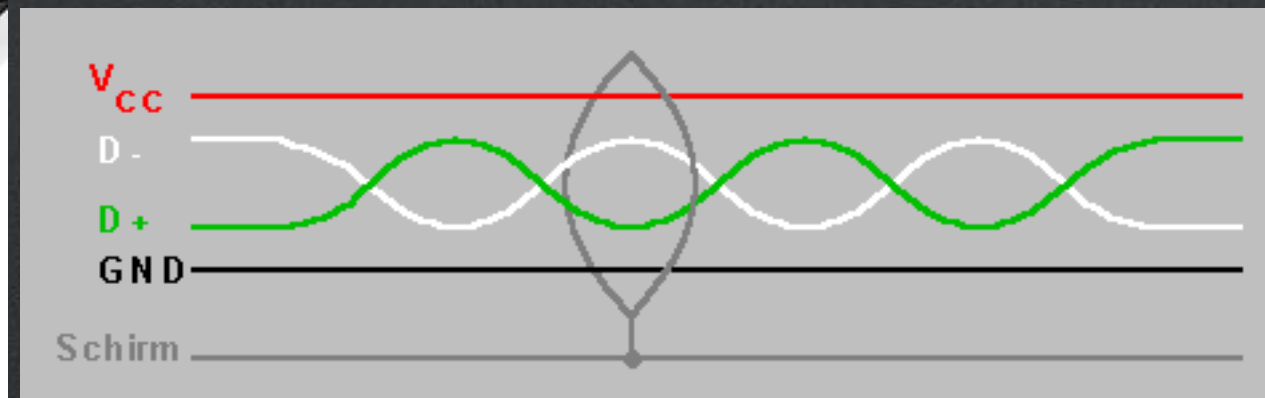
z.B. even = Anzahl der Einsen geradzahlig

Hallo = 01001000 11101101 01101100 01101100 01101111



# 1.7 Datenübertragung in seriellen und parallelen Bussen

- Serieller Bus:
  - Daten werden Bitweise nacheinander übertragen
  - es werden nur wenige Leitungen, z.B. USB zwei Leitungen benötigt
  - für weite Entfernungen geeignet
  - Beispiele: USB (Universal Serial Bus), SATA-Bus



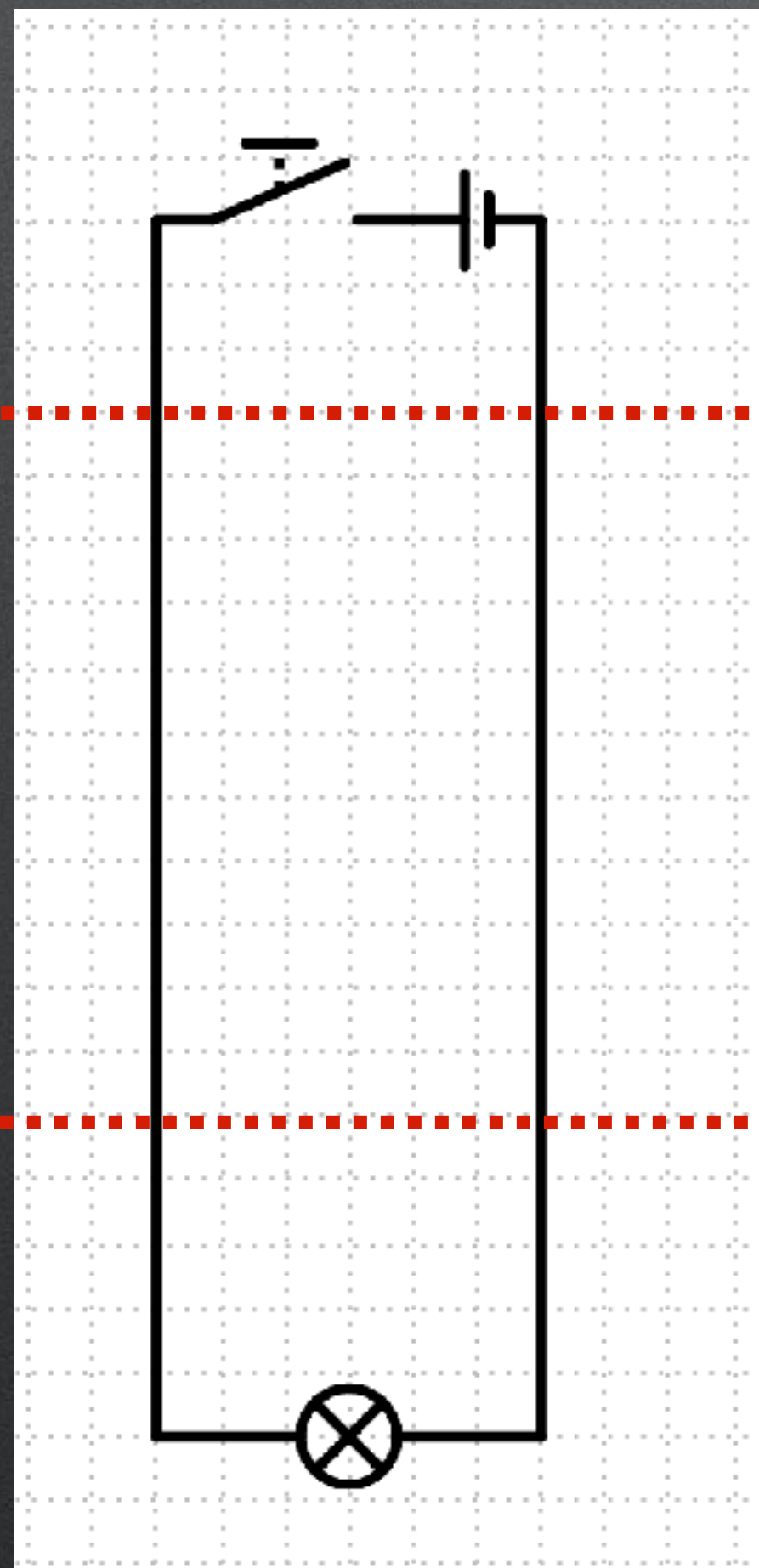
	Funktion
1	Masse
2	A+ (Senderichtung)
3	A- (Senderichtung)
4	Masse
5	B- (Empfangsrichtung)
6	B+ (Empfangsrichtung)
7	Masse



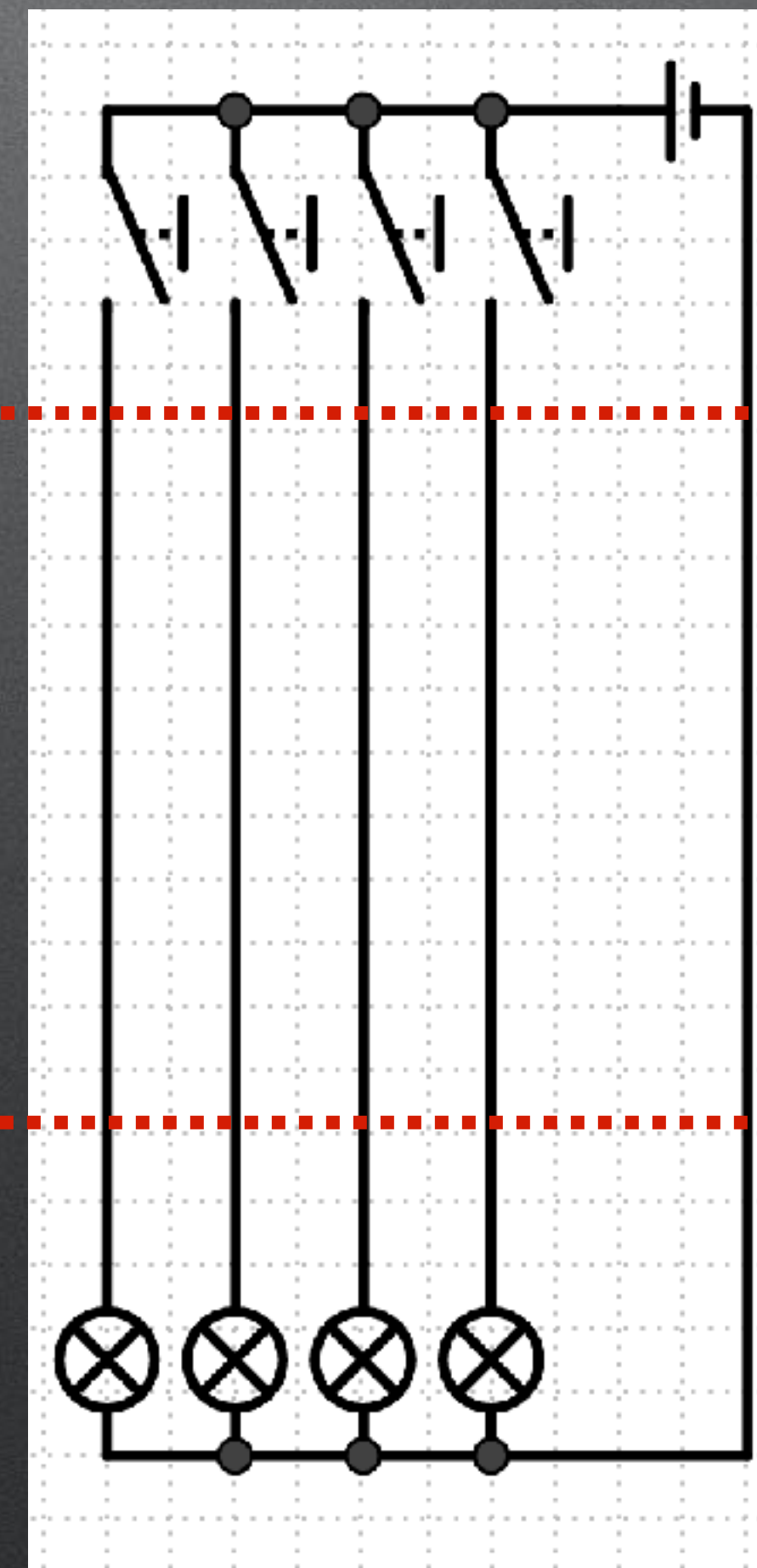
# 1.7 Datenübertragung in seriellen und parallelen Bussen

Prinzip

seriell:



parallel:



Sender

Übertragungsweg

Empfänger



# 1.7 Datenübertragung in seriellen und parallelen Bussen

- paralleler Bus:
  - Daten werden Wortweise (mehrere Bit gleichzeitig) übertragen
  - es wird für jedes bit eine Leitung benötigt, z.B. für 64 bit 64 Leitungen und eine Leitung für den Nullpol. (Spannung gibt es nur zwischen zwei Potentialen!)
  - nur für kurze Entfernungen geeignet
  - Beispiele: Bus zwischen CPU und Arbeitsspeicher, IDE-Bus



# Übung

Ordne zu:

parallele Busse

viele Leitungen  
wenige Leitungen  
für weite Strecken geeignet  
nur kurze Entfernungen  
mit einem Schritt werden mehrere Bit übertragen  
Daten werden Bit für Bit nacheinander übertragen  
Beispiel: USB  
Beispiel: Bus zwischen CPU und Motherboard

serielle Busse



# Übung

Ordne zu:

parallele Busse

viele Leitungen

wenige Leitungen

für weite Strecken geeignet

nur kurze Entfernungen

mit einem Schritt werden mehrere Bit übertragen

Daten werden Bit für Bit nacheinander übertragen

serielle Busse

Beispiel: USB

Beispiel: Bus zwischen CPU und Motherboard



# 1.8 EVA-Prinzip

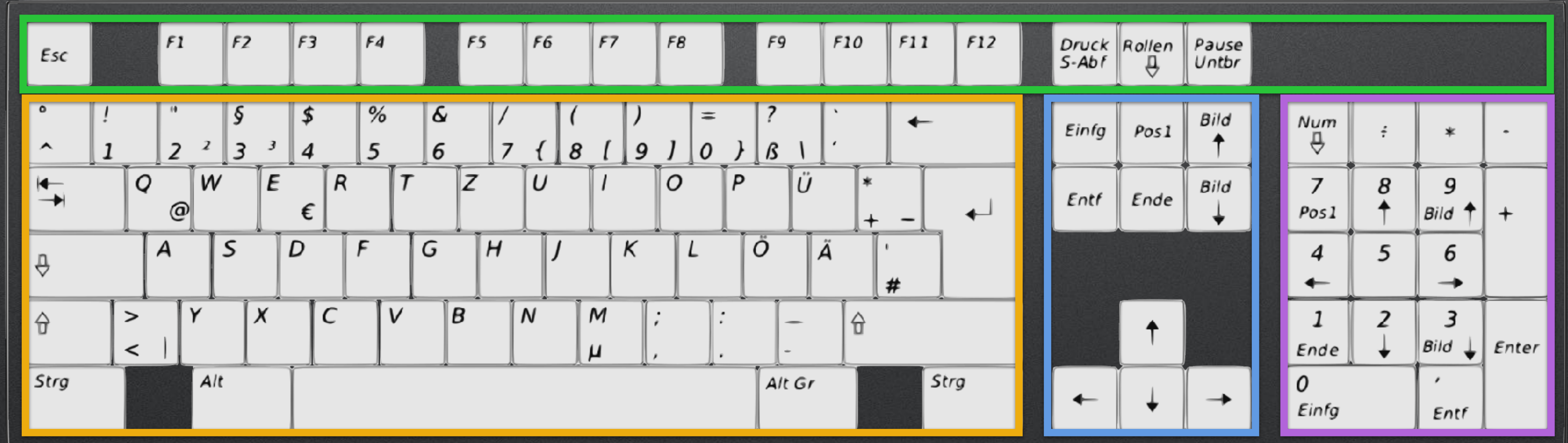
EINGABE	VERARBEITUNG	AUSGABE
<ul style="list-style-type: none"><li>• Tastatur</li><li>• Maus</li><li>• Touchscreen</li><li>• Mikrofon/Kamera</li><li>• Dateispeicher</li></ul>	<ul style="list-style-type: none"><li>• CPU</li><li>• (Rechenwerk/ Steuerwerk)</li><li>• Arbeitsspeicher</li></ul>	<ul style="list-style-type: none"><li>• Bildschirm</li><li>• Drucker</li><li>• Dateispeicher</li></ul>



# 1.8.1 Eingabegerät - Tastatur

## Aufbau

Eine Tastatur besteht aus dem **Steuerblock** (ESC und Funktionstasten), **Schreibmaschinenblock** (Tastenbelegung nach DIN 2137), **Bewegungsblock** (Cursortasten), **Numerikblock** (Zahlen und Rechenzeichen)





# 1.8.1 Eingabegerät - Tastatur

Bezeichnung einiger Tasten

^ Accent Circonflexe

' Accent Aigu (Akut)

` Accent Grave (Gravis)

' Apostroph

\* Asterisk (Stern)

° Grad

# Hash (Raute, Doppelkreuz)

@ At (bei)

& Et (kaufm. Und)

/ Slash; \ Backslash; | Bar; - Bindestrich; \_ Unterstrich



# 1.8.1 Eingabegerät - Tastatur

Bezeichnung einiger Tasten

esc: Escape (Flucht)

F1...F12: Funktionstasten

Umschalttaste

Feststelltaste

Strg (Control): Steuerungstaste

Fn: Funktion

Alt (Option): Alternative

ALT Gr: alternative Grafik



# 1.9 Verzeichnisbaum

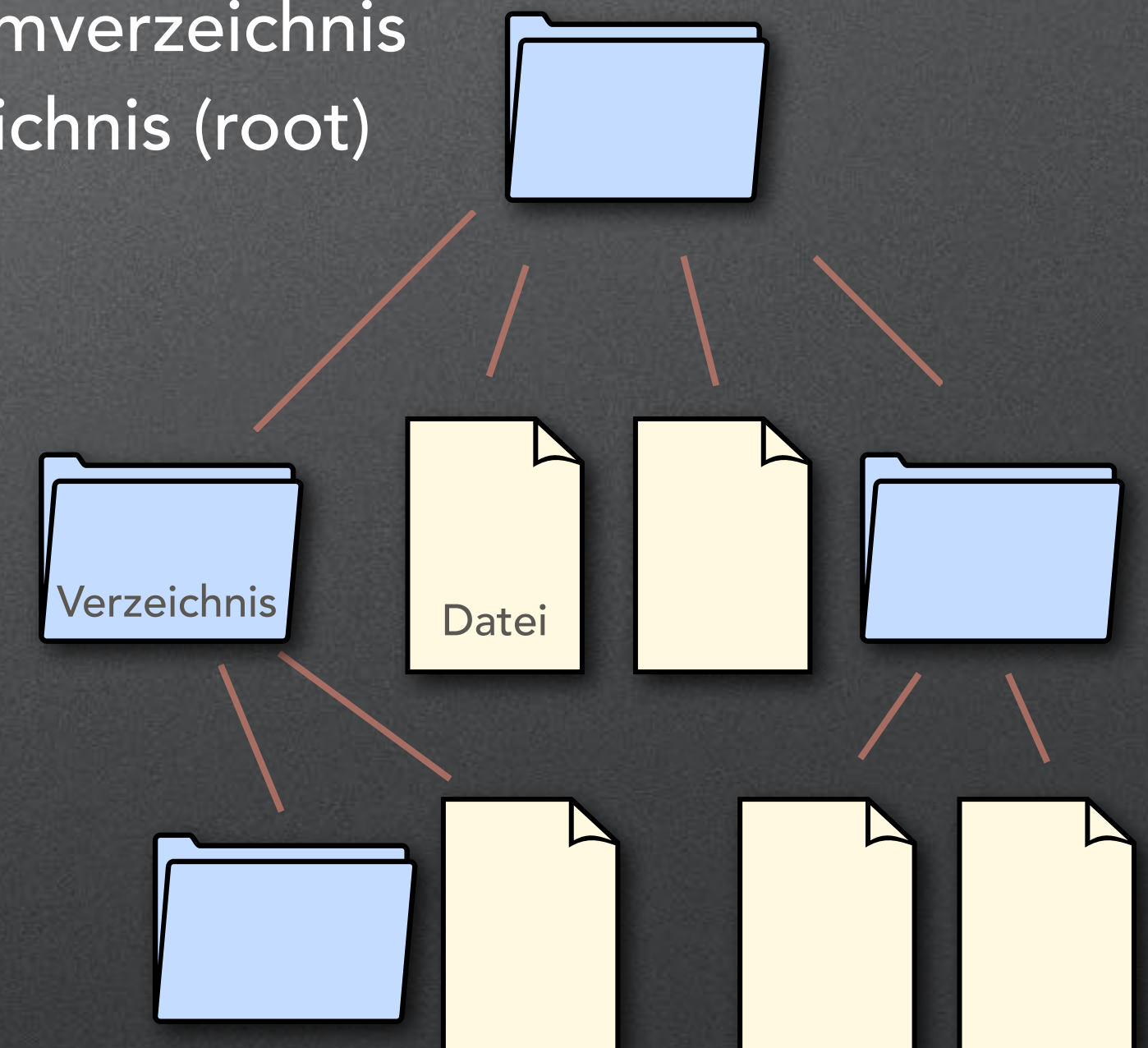




# 1.9 Verzeichnisbaum

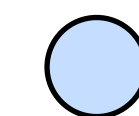


Laufwerk mit Stammverzeichnis  
oder Wurzelverzeichnis (root)



Legende des Baumes:

Kante



Knoten



Blatt



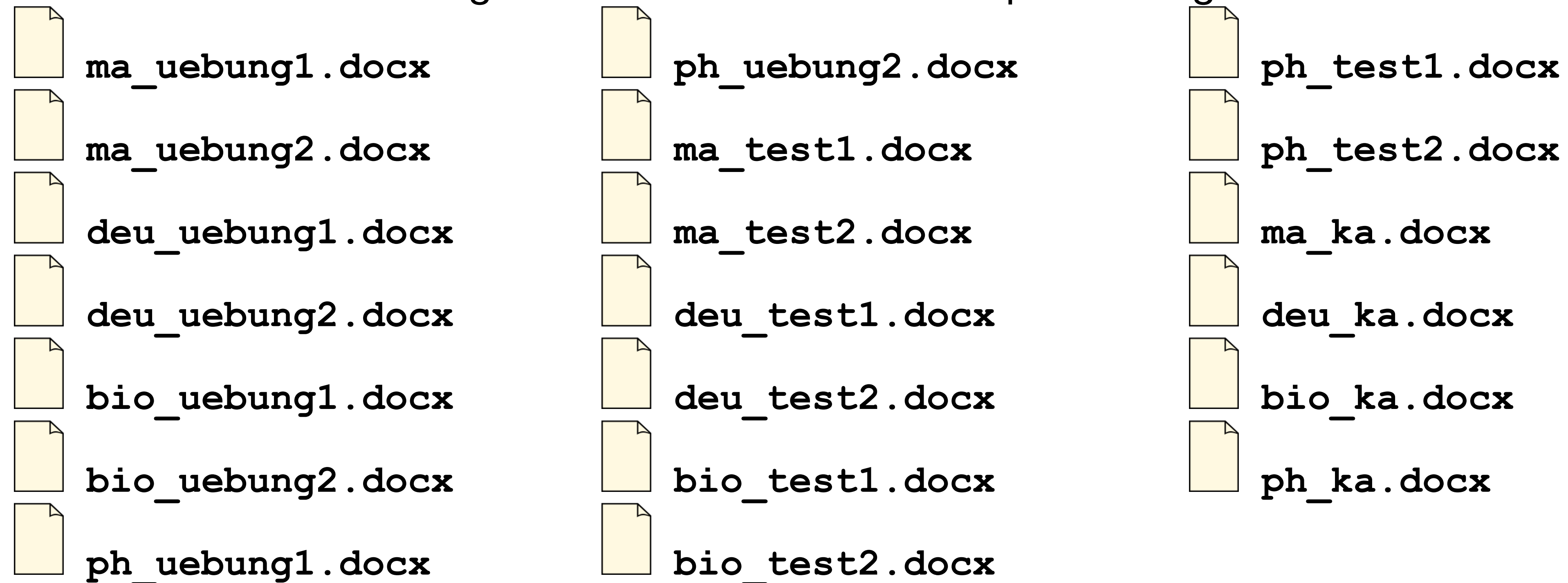
# Umgang mit Dateien und Verzeichnissen

---

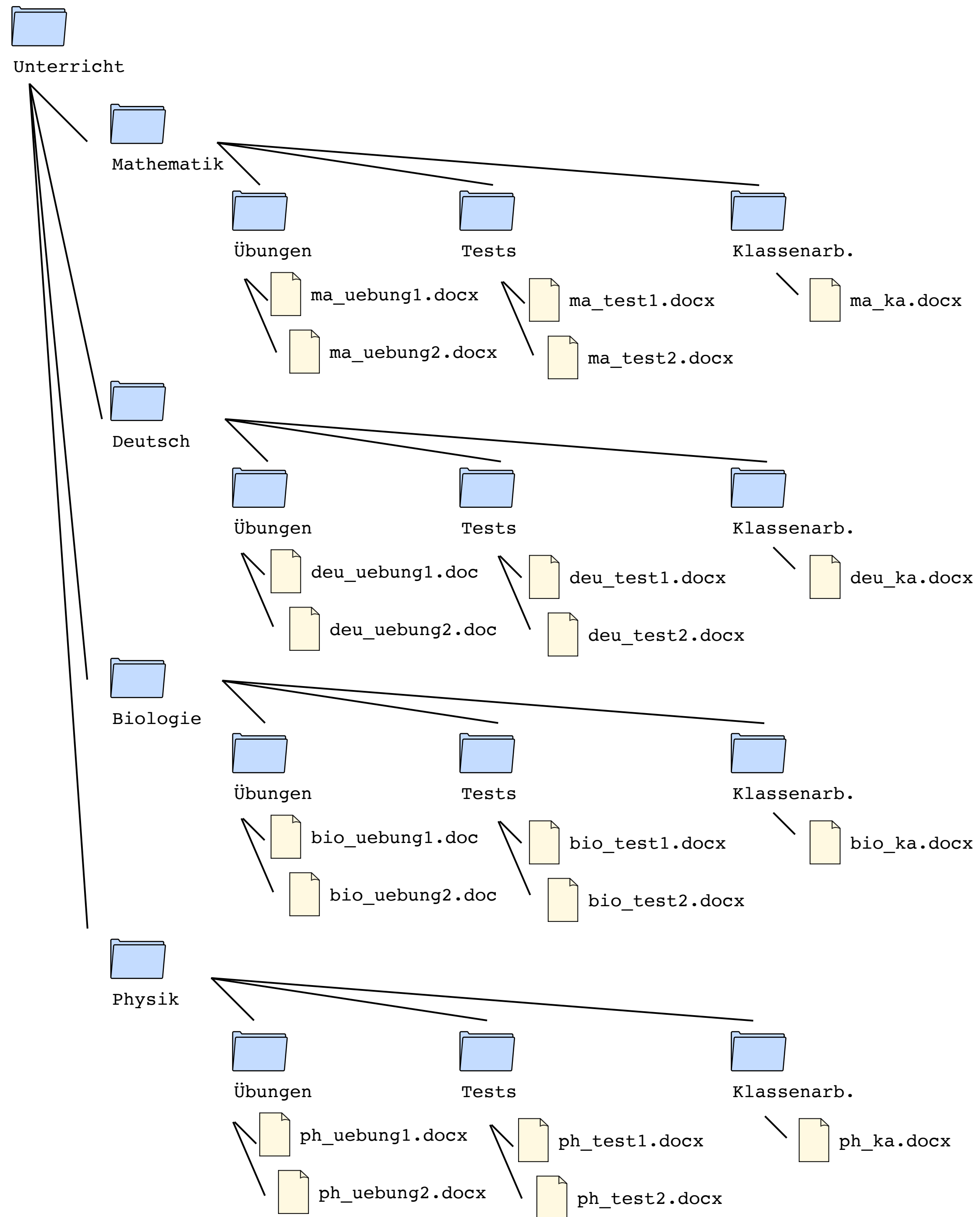
## Aufgaben:

1. Es sollen jeweils für die Unterrichtsfächer „Mathematik“, „Deutsch“, „Biologie“ und „Physik“ zwei Übungsaufgabenblätter, zwei Tests und eine Klassenarbeit übersichtlich in einem Verzeichnis Unterricht gespeichert werden.

Plane in einem Baumdiagramm zur übersichtlichen Speicherung der Dateien.









# 1.9.1 Datei

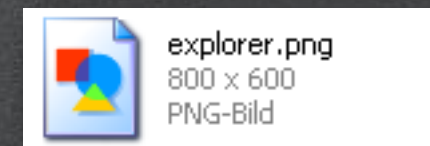
## Dateityp (Type of File)

### Reguläre Datei

- Anwendungsdatei  
(Text, Bild, Film, Audio...)
- Programm



Übungsaufgaben.txt  
Textdokument  
2 KB



explorer.png  
800 x 600  
PNG-Bild



Internet Explorer



PAP.pdf

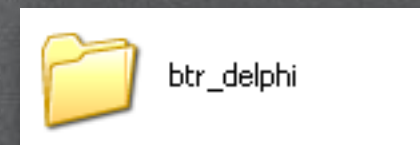


Übungsaufgaben

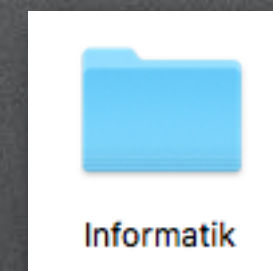
### Verzeichnis



Eigene Dateien

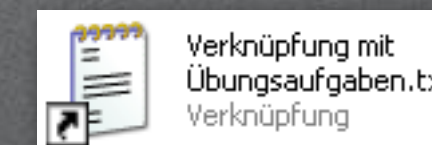


btr\_delphi

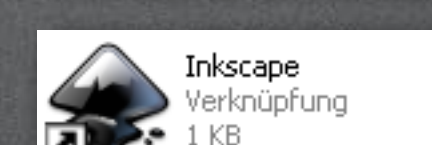


Informatik

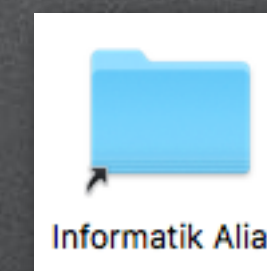
### Link / Verknüpfung



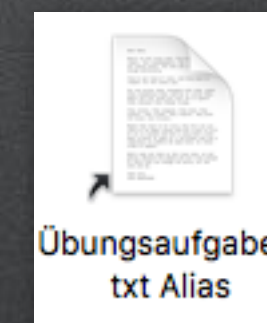
Verknüpfung mit  
Übungsaufgaben.txt  
Verknüpfung



Inkscape  
Verknüpfung  
1 KB



Informatik Alias



Übungsaufgaben.  
txt Alias

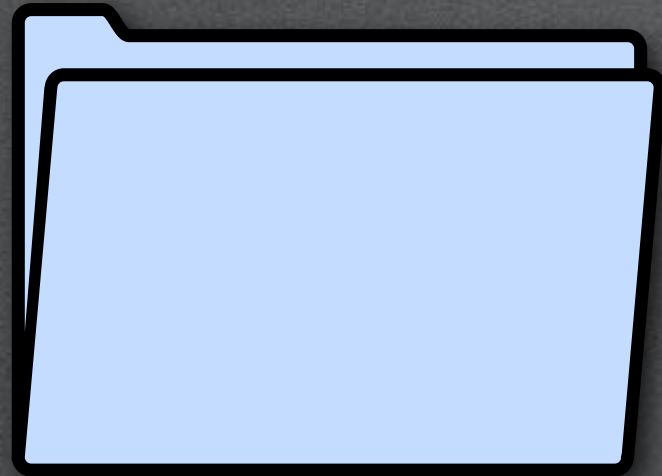


# Dateierweiterungen

Dateierweiterung	Name	Verwendung
txt	Text	Speichern von Texten.
jpg, jpeg		
bmp		
docx		
exe		
rtf		
odt		
pdf		
cal		
vcf		



# 1.9.2 Verzeichnis



Ein Verzeichnis ist eine besondere Datei, sie enthält eine Auflistung von Dateien und Unterverzeichnissen. Die Funktion eines Verzeichnisses im Dateisystem ähnelt dem eines Buches, es wird hier aber auf Speicherorte verwiesen.

## Index

Eingabegerät (Forts.)

*Tastatur* 157

Eingabesteuerung

*durch das Betriebssystem* 281

Eingabeumleitung

*in Unix-Shells* 381

*Windows* 335

Emacs, Texteditor (Forts.)

*Text markieren* 419

E-Mail

*Attachment* 264

*Entwicklung* 173

*Hoax* 1070

*HTML-Hyperlink auf* 883

Entwicklungsprozess, Software  
(Forts.)

*Unified Process* 650

Entwurf, Software-Engineering  
646

*Schnittstelle* 647

*Standalone-System* 646



# Wie arbeitet man im Verzeichnisbaum?

- Das Programm zur Verwaltung der Dateien im Verzeichnisbaum nennt sich:
  - Beispiele
    - Windows:
    - Mac:
    - Tablet:



# Arbeit mit Dateien

Wie erstellt man ...

- eine Datei?
- ein Verzeichnis?



# Arbeit mit Dateien

- ausführen oder öffnen
- umbenennen:
- verschieben (ausschneiden):
- löschen:
- kopieren:
- hineinwechseln in ein Verzeichnis:



# Hinweise zur Verwendung von Dateinamen

Dateien und Verzeichnisse:

- sollten deskriptiv (beschreibend) sein
- keine Leer- und Sonderzeichen enthalten (/,\,\$,@,#,“,’,?,!,^ usw.)
- möglichst nur aus Kleinbuchstaben bestehen



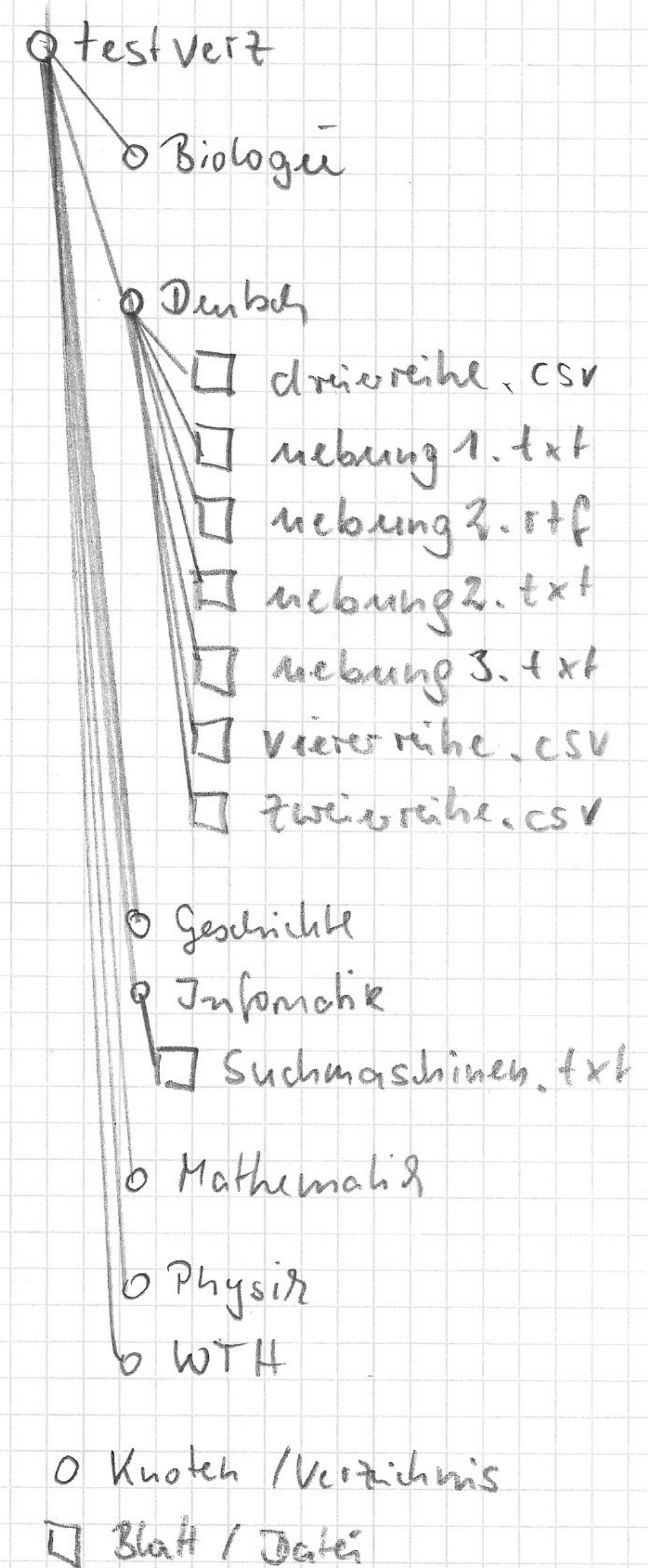
# Arbeiten mit dem Dateimanager (engl. File Manager)

- verschiedene Darstellungen (Symbolansicht, Listenansicht mit mehreren Eigenschaften, Vorschau)
- Möglichkeiten der Sortierung nach Dateimerkmalen (z.B. Erstellungsdatum, Änderungsdatum, Art, Dateiname)
- Navigatorische Elemente zum Bewegen im Verzeichnisbaum
- Suchmöglichkeiten
- Werkzeuge zum
  - Umbenennen des Dateinamens
  - zum Verschieben
  - zum Kopieren
  - Erstellen und Entpacken von (komprimierten) Archiven



# Auswertung d. Übung

- 1. grafische Struktur erstellen

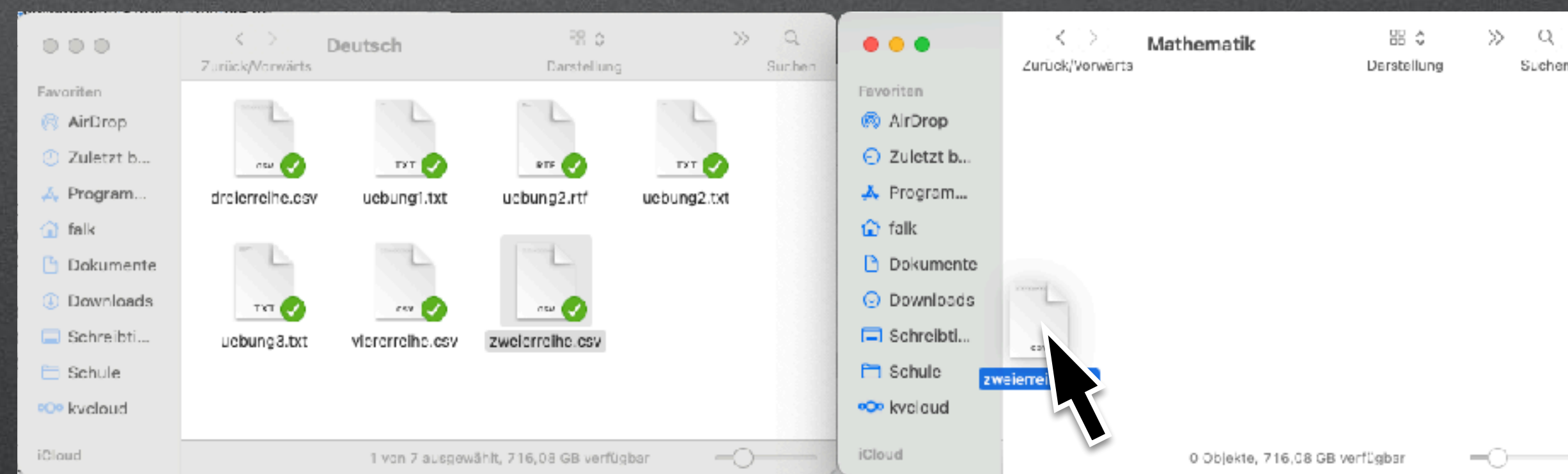
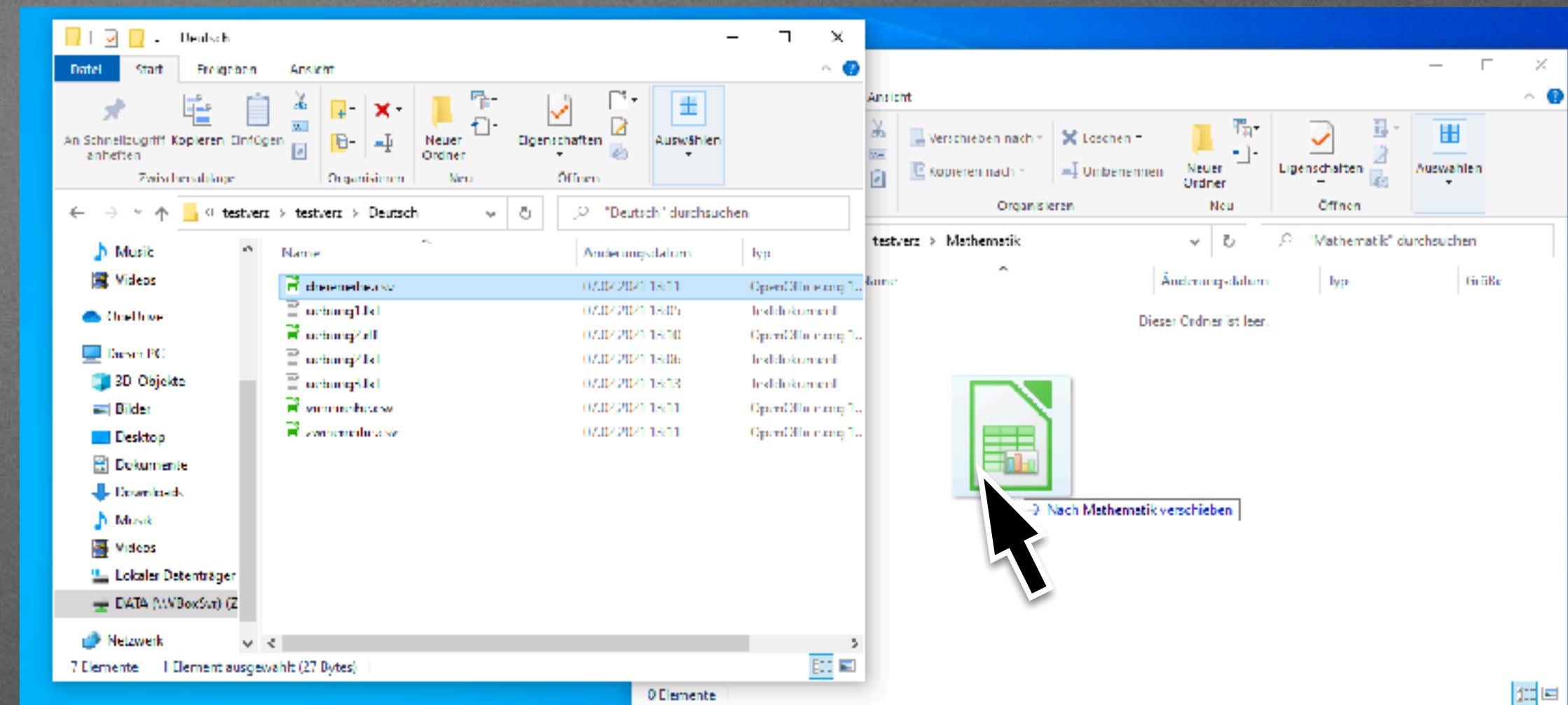




# Auswertung d. Übung

Verschiebe die Dateien

- zweierreihe.csv,
  - dreierreihe.csv und
  - viererreihe.csv
- in das Verzeichnis Mathematik

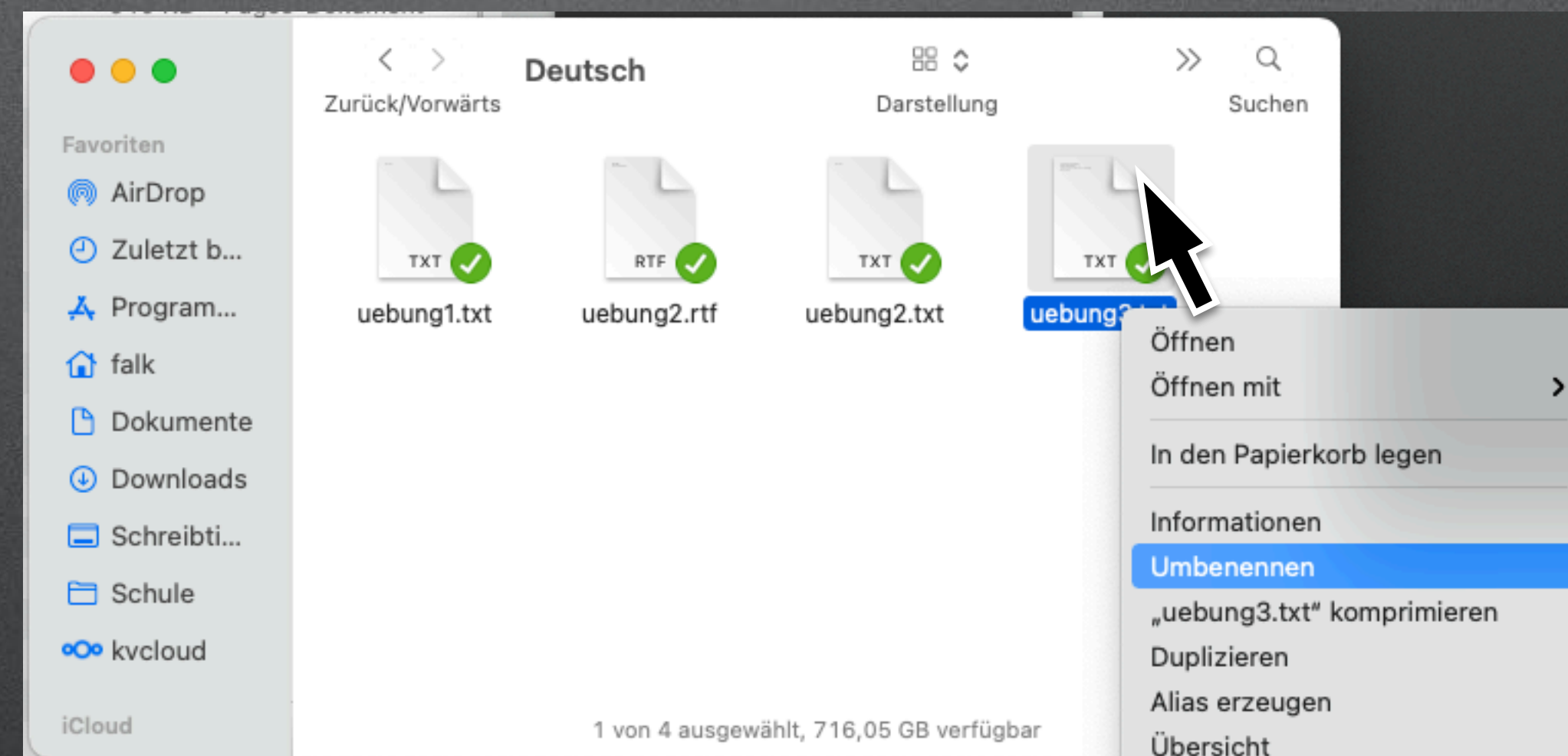


Mac: +command (in ein anderes Dateisystem)



# Auswertung d. Übung

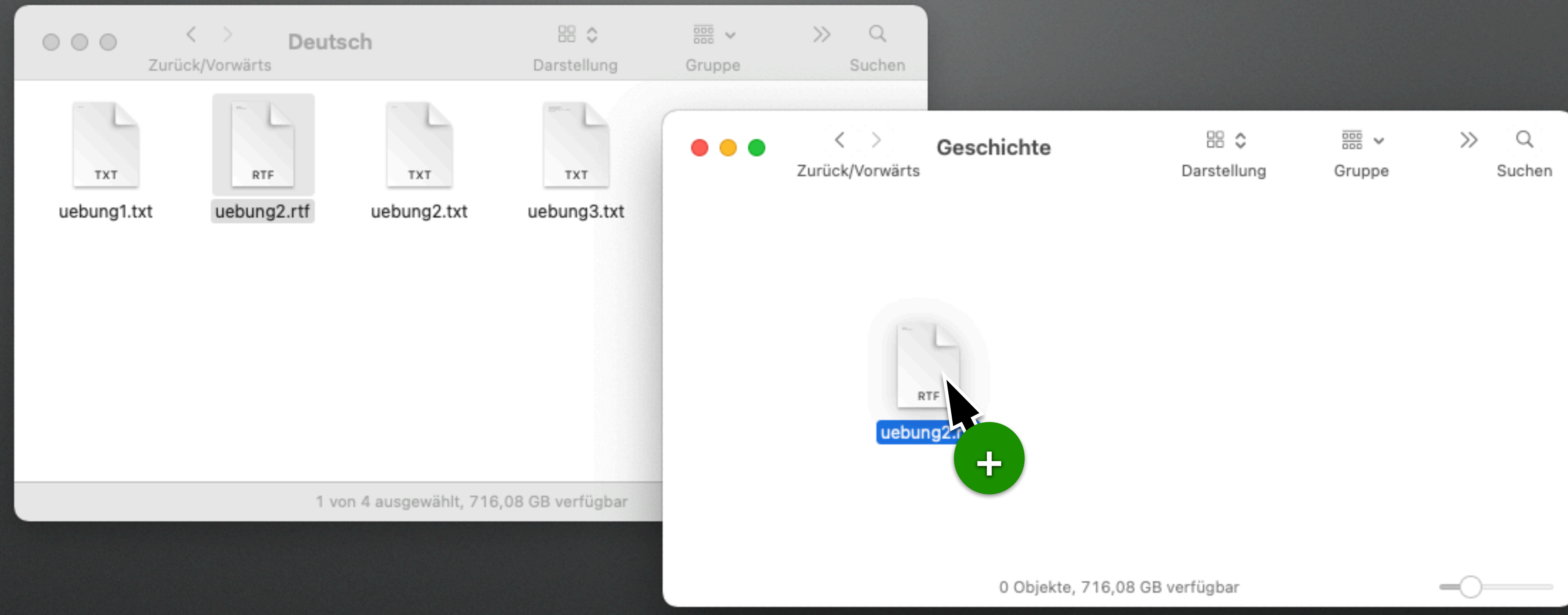
Wechsle in das Verzeichnis „Deutsch“ und benenne dort die Datei uebung3.txt in gedicht.txt um.





# Auswertung d. Übung

Kopiere die Datei uebung2.rtf im Verzeichnis Deutsch in den Ordner Geschichte.

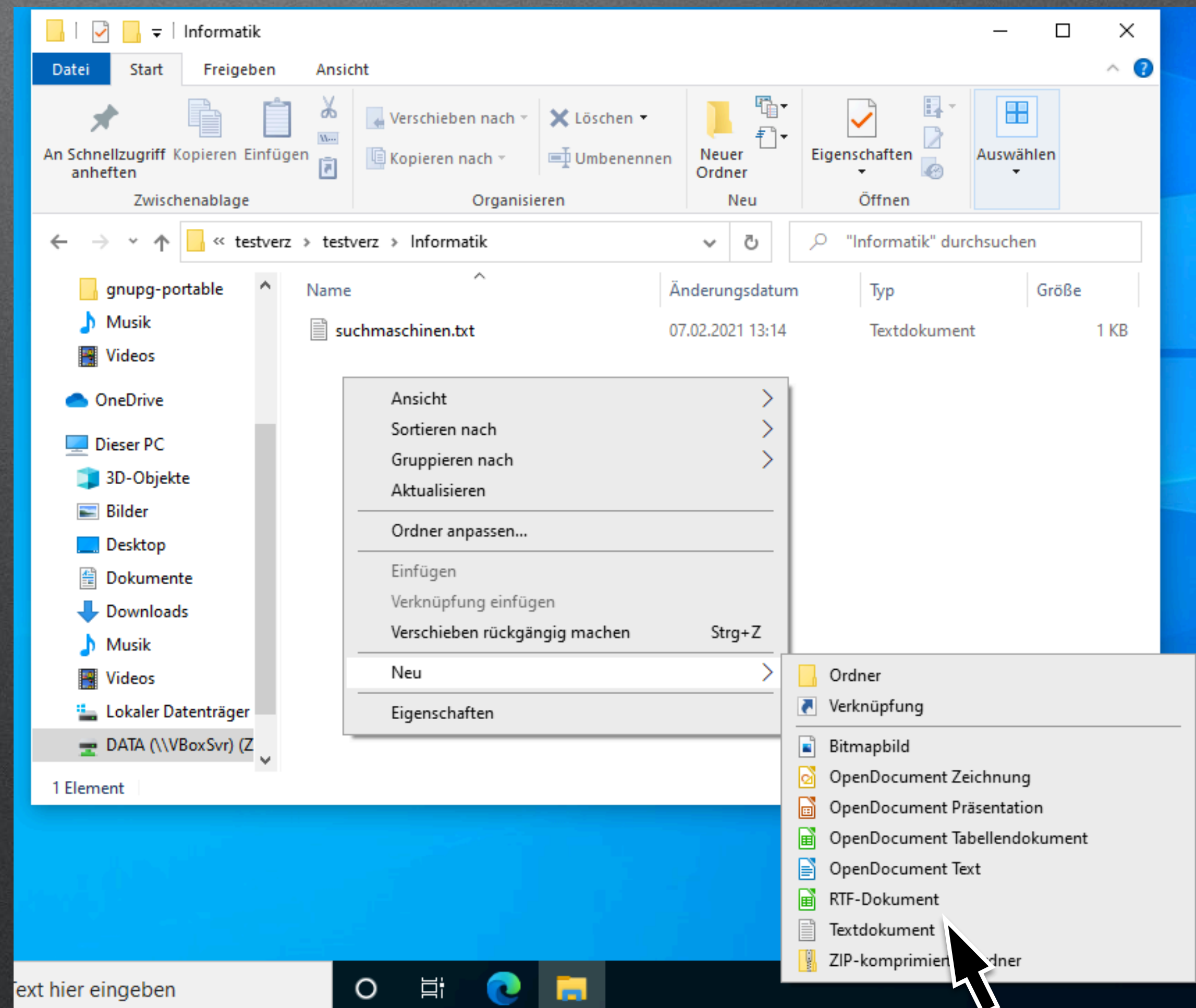


Mac +Umschalttaste  
Win: + Strg



# Auswertung d. Übung

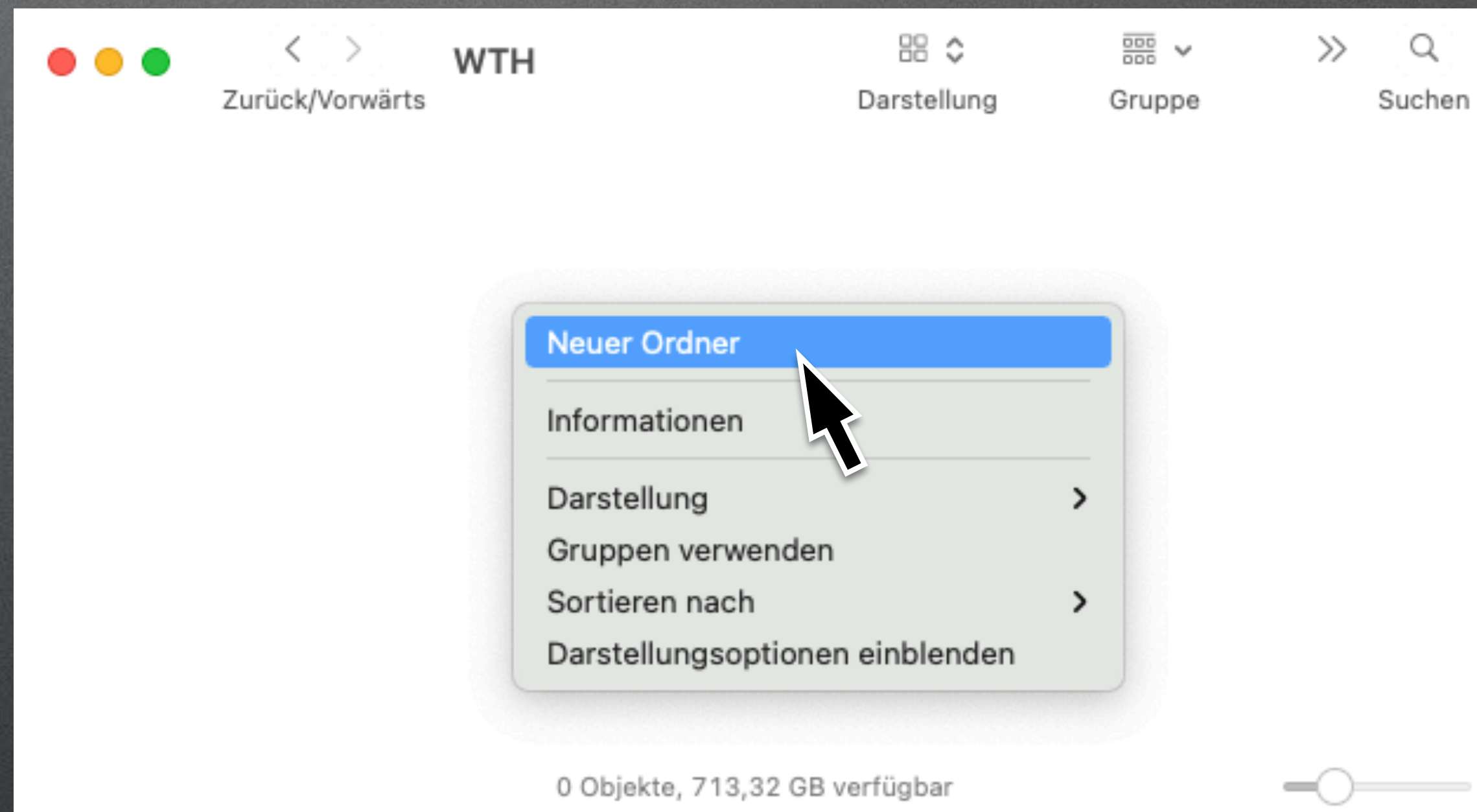
Bitte lege eine Textdatei „eingabegeräte.txt“ im Verzeichnis „H:\testverz\Informatik“ an. Schreibe in die Datei zwei Eingabegeräte.





# Auswertung d. Übung

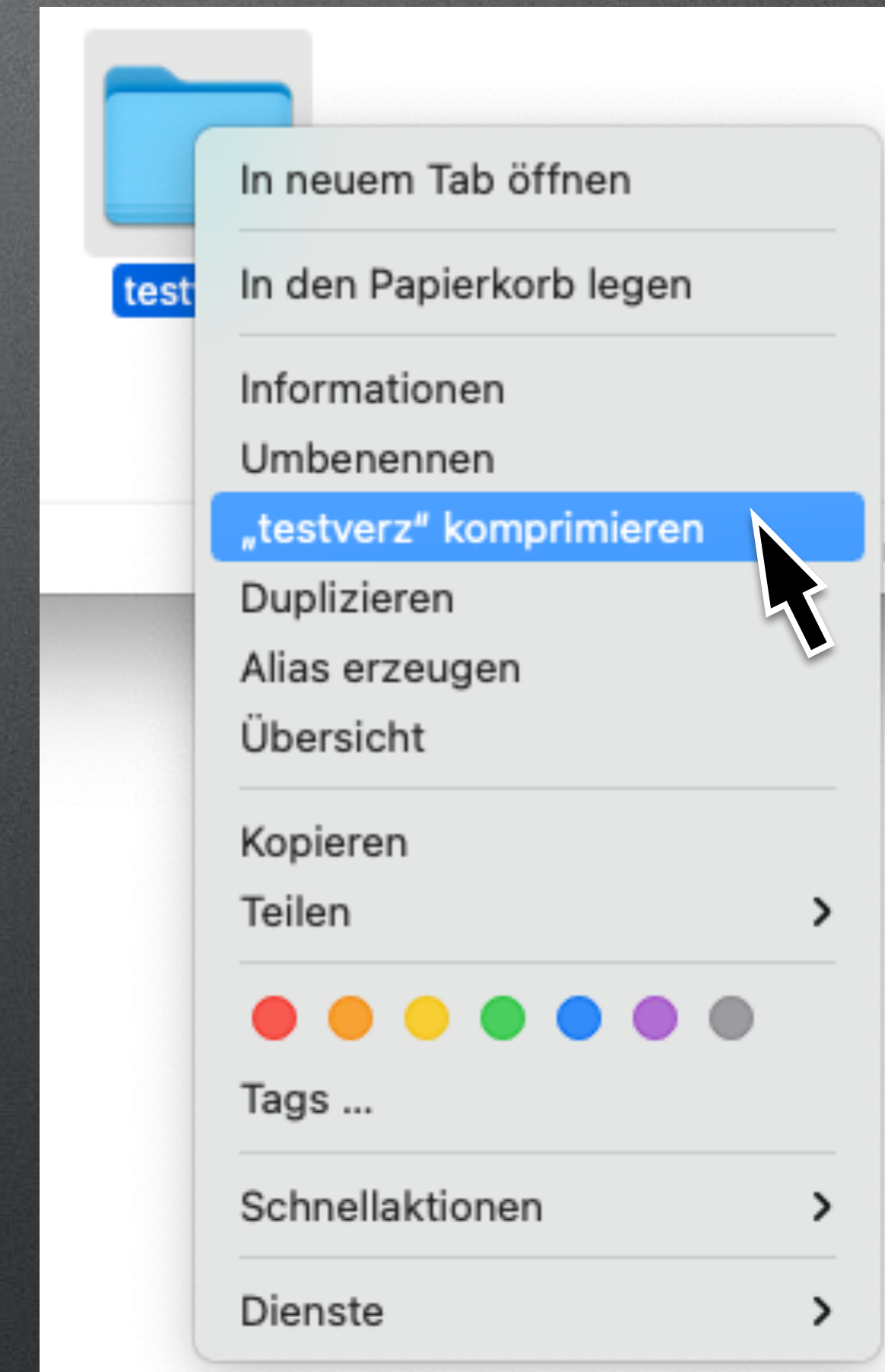
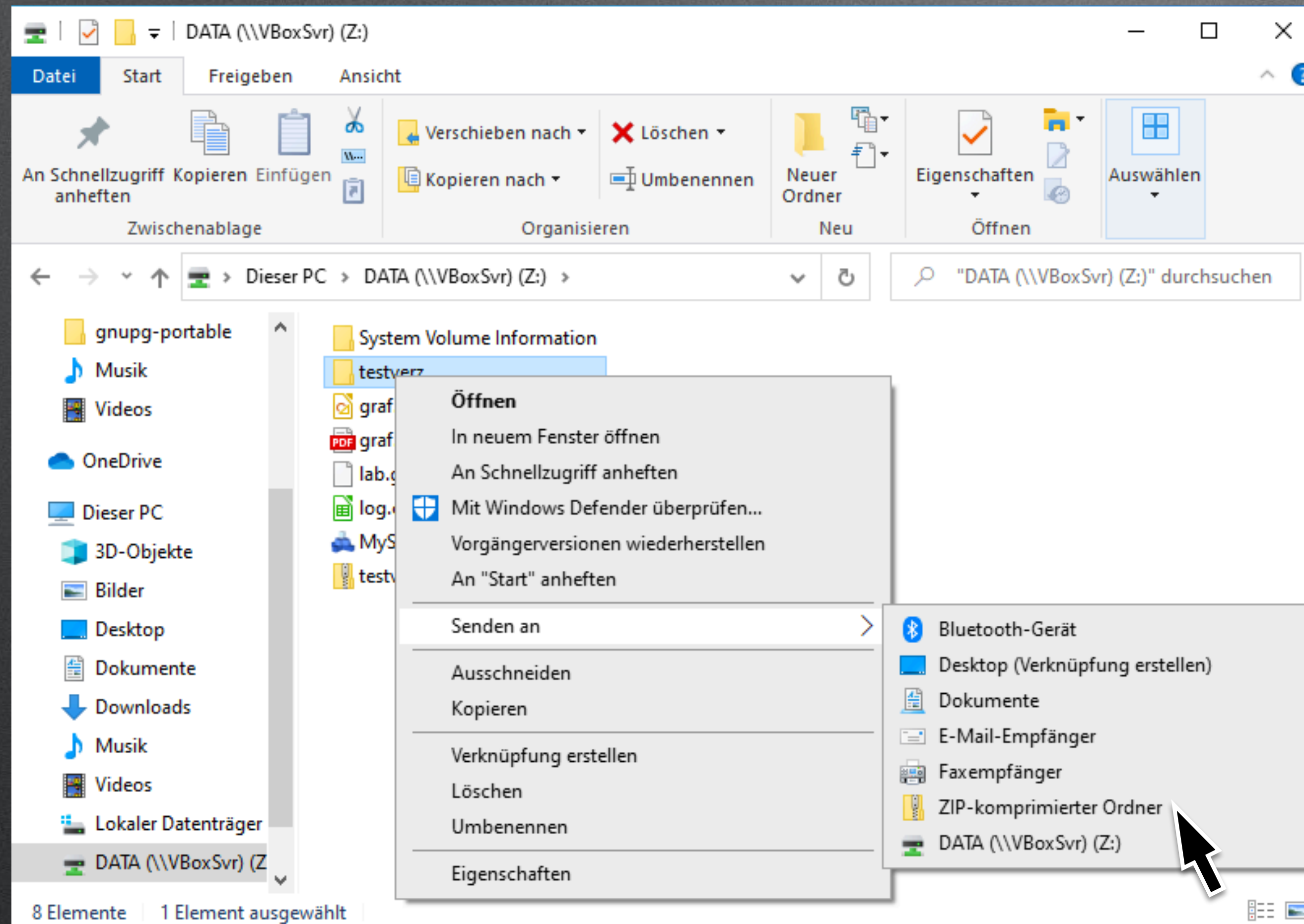
Lege im Verzeichnis „H:\testverz\WTH“ ein Unterverzeichnis "Nähen" an.





# Auswertung d. Übung

## Zip-Archiv erstellen



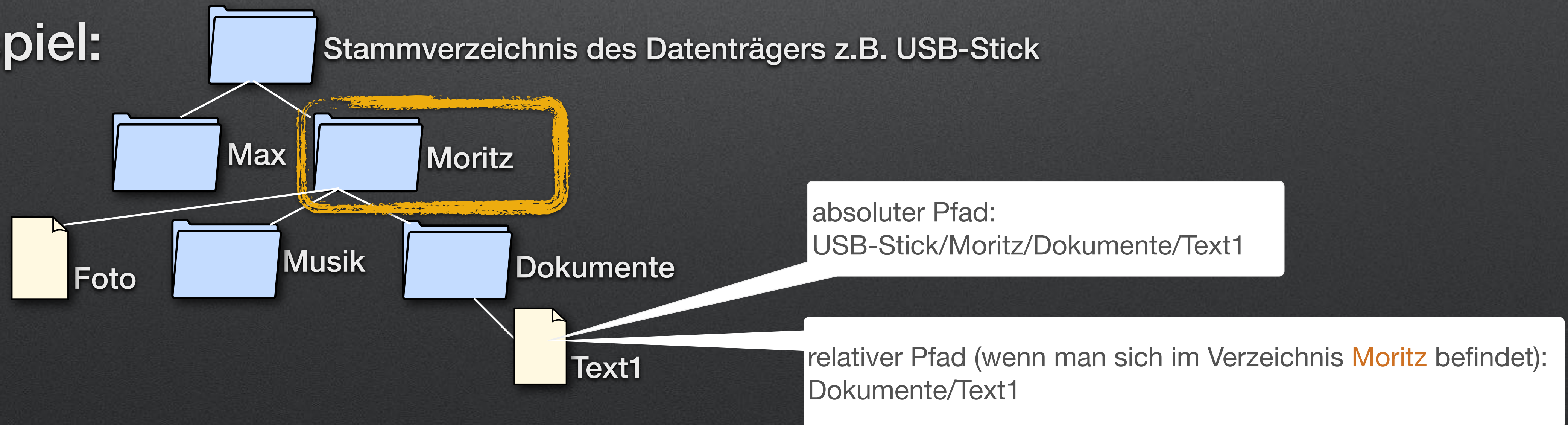


# 1.9.3 Dateipfad

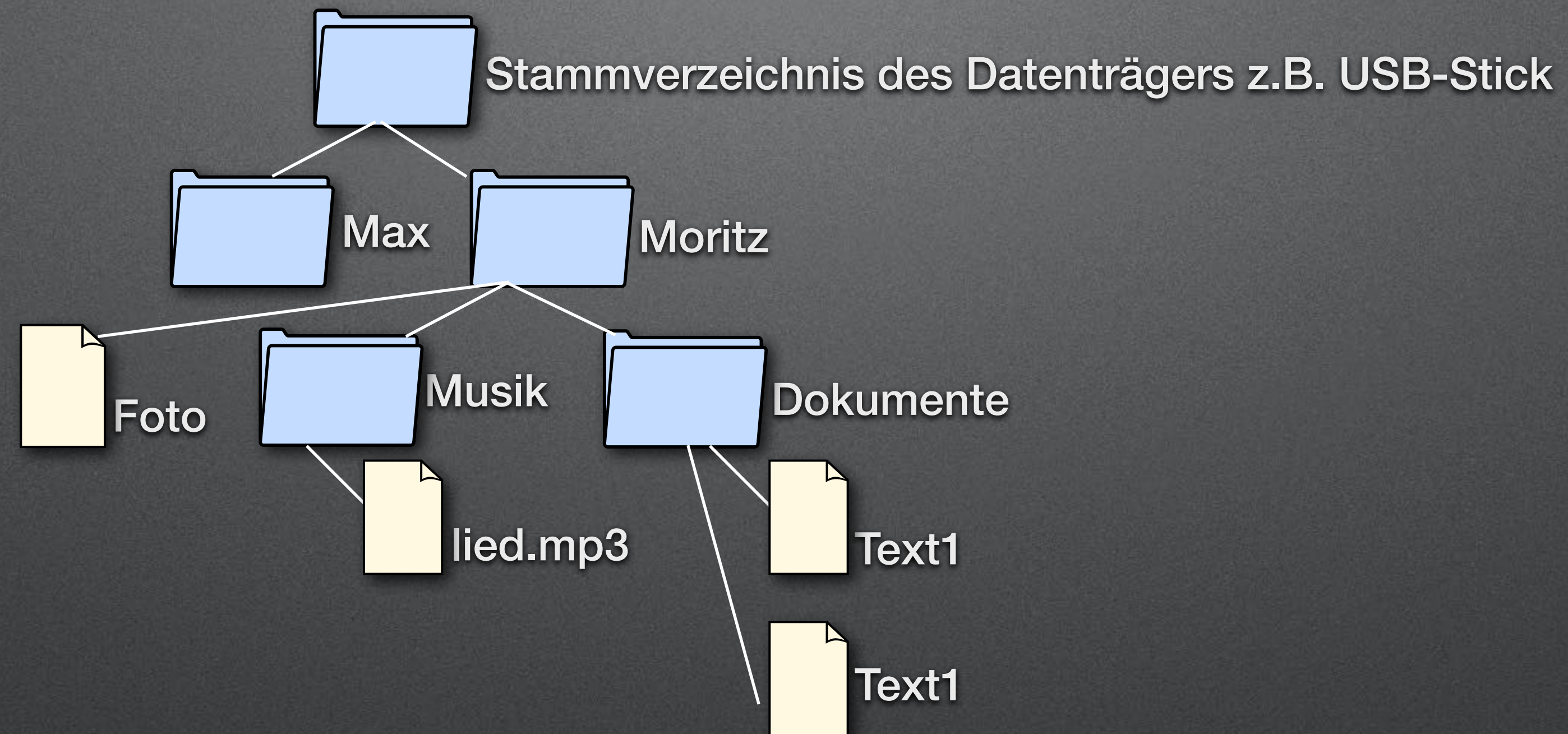
absoluter Pfad: gibt den Speicherort einer Datei unabhängig vom gerade verwendeten Verzeichnis (Arbeitsverzeichnis) an.

relativer Pfad: gibt den Weg zur Datei in Abhängigkeit vom aktuellen Verzeichnis an. Um eine Datei zu finden, muss man das Arbeitsverzeichnis kennen.

Beispiel:











SD-Karte



Bilder



Musik



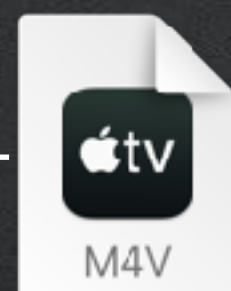
fruehling.mp3



sommer.mp3



Filme



kobold.m4v

  Arbeitsverzeichnis

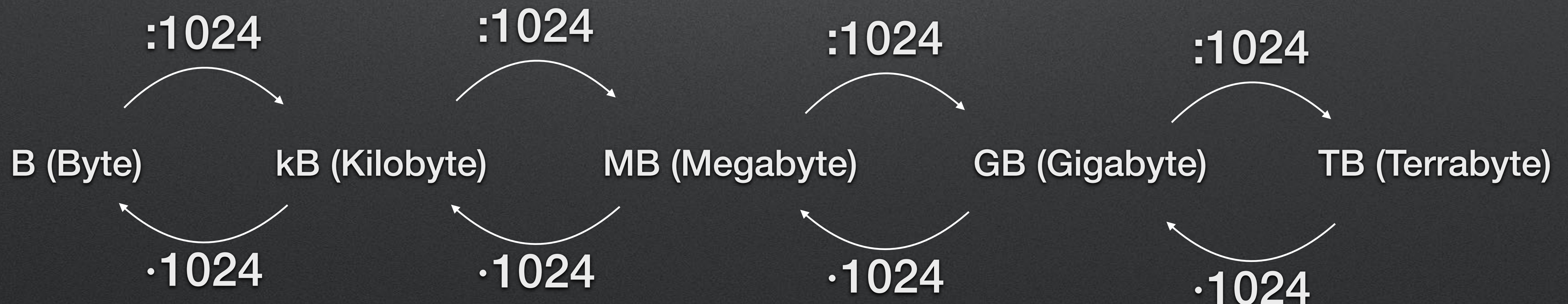


## 1.9.4 Dateigröße

Die Speichergröße einer Datei wird in Byte angegeben. Dabei gilt, dass 8 Bit einem Byte entsprechen. (1 Bit  $\hat{=}$  Information Eins oder Null)

*1 Byte = 8 bit*

Für die Umrechnung in größere Einheiten gilt anders als in der Physik oder Mathematik nicht der Faktor 1000 sondern  $2^{10}=1024$ .





# Berechnung Dateigröße

Übung:

1. Aus wie vielen Bits besteht eine 20 Byte große Datei?
2. Eine Datei besitzt eine Größe von 2048 Byte. Rechne die Speichergröße um in kByte um.
3. Rechne um:
  - a) 1,5 TB = GB
  - b) 440320 MB = TB
  - c) 1,5 kB = B
4. Ein Buchstabe wird mit 8 bit codiert (z.B. A=0100 0001). Ein Text besteht aus 3584 Textzeichen. Welche Speichergröße benötigt diese Datei in kB?



# 1.10. Dateisystem

## 1.10.1 Aufgaben

Ein Dateisystem

- organisiert der Dateiablage
- speichert die Dateinamen und den Speicherort
- verwaltet die Zugriffe auf die Dateien (Berechtigungen, Zugriffszeitpunkte usw.)

## 1.10.1 einige Eigenschaften

- Zeichensatz und Länge von Dateinamen
- max. Anzahl von Dateien
- Dateiattribute festlegen, z.B. schreibgeschützt, versteckt
- max. Speichergröße für Dateien
- usw.



## 1.10.3 Vom Datenträger zum Dateisystem

Datenträger



Einteilung in  
Speicherbereiche  
(partitionieren)



Erstellen des  
Dateisystems  
in jeder Partition  
(formatieren)

FAT-FS (ohne Benutzerrechte)  
NTFS (Windows)  
APFS (Apple)  
EXT-FS (Linux)

Dateiarbeit

- speichern
- aufrufen
- löschen
- umbenennen



# 2 Computer nutzen und anwenden

(Objekte - Attribute - Methoden)



# 2.1 Textverarbeitung

- Aufgabe der Textverarbeitung ist das Erstellen und Bearbeiten von Texten in der elektronischen Datenverarbeitung

## 2.1.1 Programme zur Textverarbeitung:

Name	Firma oder Team	Lizenz	WYSIWYG what you see is what you get
Word	Microsoft	kommerziell	
LibreOffice	The Document Founation	freie Nutzung	
Pages		kommerziell	

WYSIWYG: Der Ausdruck soll dem angezeigten Text in der Textverarbeitung entsprechen.



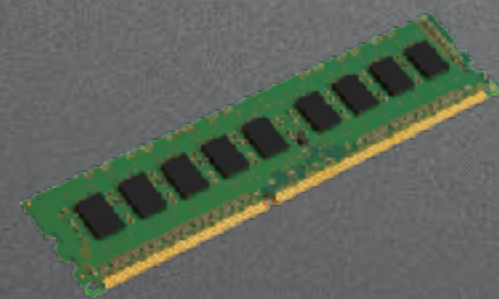
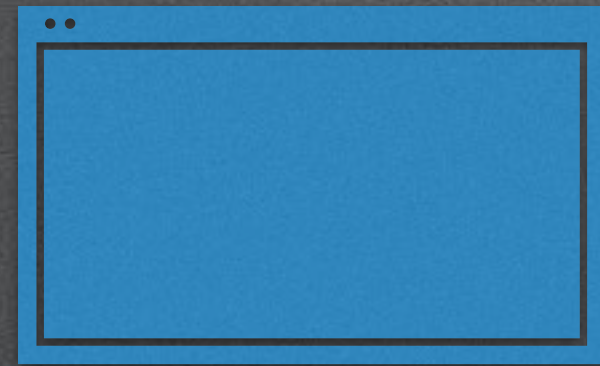
## 2.1.2 Dateiformate zur Textformatierung:

Dateiformat	Firma oder Team	Lizenz	Verbreitung
docx	Microsoft	frei verfügbar	hoch
pages	Apple	kommerziell	gering
HTML	W3C	RFC, frei verfügbar	sehr hoch
PDF	Adobe	seit 2008 offener Standard	sehr hoch



## 2.1.3 Datenfluss in der Textverarbeitung

Textverarbeitung → Daten im Arbeitsspeicher → reguläre Datei



Hinweis: Datei rechtzeitig und regelmäßig sichern (Strg-S oder ⌘-S)