



64-040 Modul InfB-RSB

Rechnerstrukturen und Betriebssysteme

[https://tams.informatik.uni-hamburg.de/
lectures/2019ws/vorlesung/rsb](https://tams.informatik.uni-hamburg.de/lectures/2019ws/vorlesung/rsb)

– Kapitel 5 –

Andreas Mäder



Universität Hamburg
Fakultät für Mathematik, Informatik und Naturwissenschaften
Fachbereich Informatik

Technische Aspekte Multimodaler Systeme

Wintersemester 2019/2020



Zeichen und Text

Ad-Hoc Codierungen

ASCII und ISO-8859

Unicode

Tipps und Tricks

Base64-Codierung

Literatur

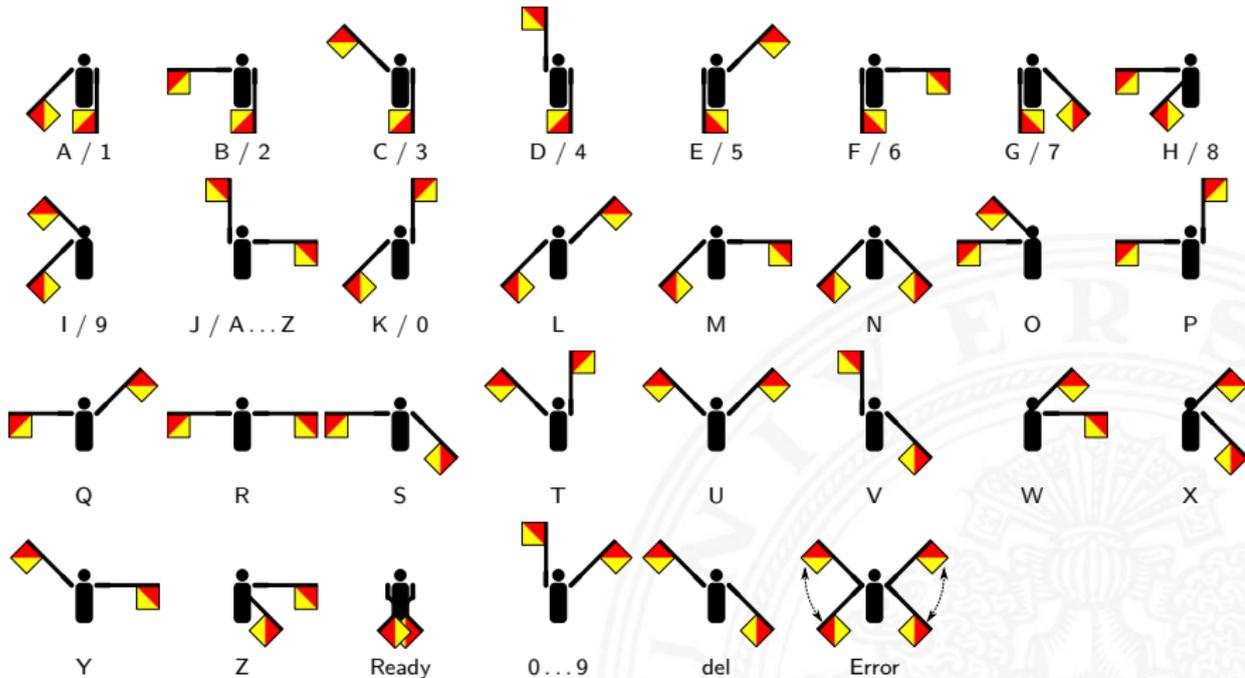


- ▶ **Zeichen:** engl. *character*
Element z aus einer zur Darstellung von Information vereinbarten, einer Abmachung unterliegenden, endlichen Menge Z von Elementen
- ▶ Die Menge Z heißt **Zeichensatz** oder **Zeichenvorrat**
engl. *character set*
- ▶ **Binärzeichen:** engl. *binary element, binary digit, bit*
Jedes der Zeichen aus einem Vorrat / aus einer Menge von zwei Symbolen
- ▶ **Numerischer Zeichensatz:** Zeichenvorrat aus Ziffern und/oder Sonderzeichen zur Darstellung von Zahlen
- ▶ **Alphanumerischer Zeichensatz:** Zeichensatz aus (mindestens) den Dezimalziffern und den Buchstaben des gewöhnlichen Alphabets, meistens auch mit Sonderzeichen (Leerzeichen, Punkt, Komma usw.)

Wiederholung: Zeichen (cont.)

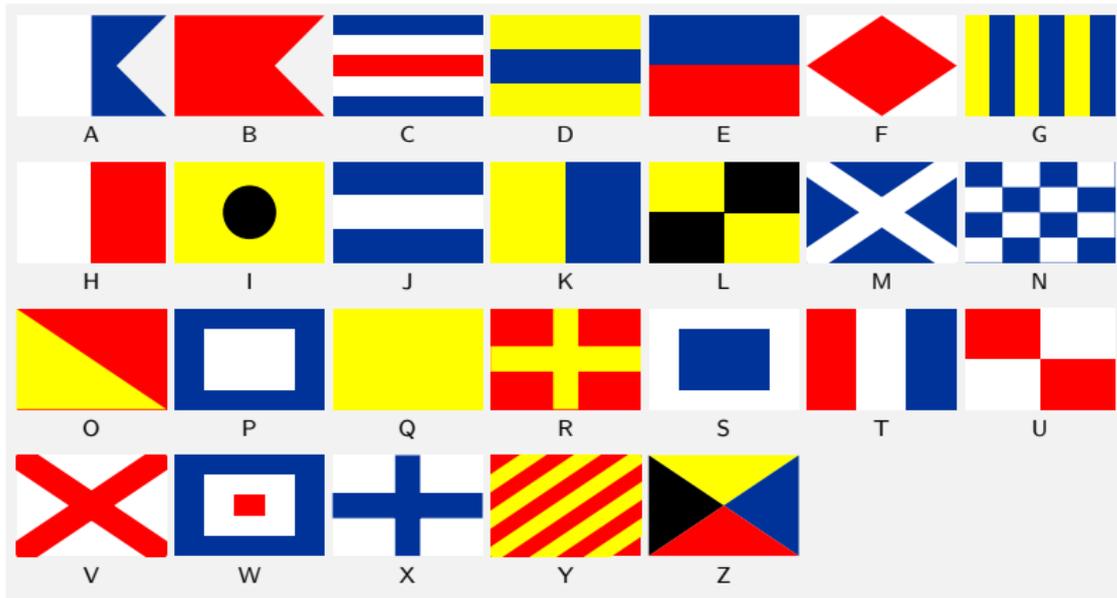
- ▶ **Alphabet:** engl. *alphabet*
Ein in vereinbarter Reihenfolge geordneter Zeichenvorrat
- ▶ **Zeichenkette:** engl. *string*
Eine Folge von Zeichen
- ▶ **Wort:** engl. *word*
Eine Folge von Zeichen, die in einem gegebenen Zusammenhang als Einheit bezeichnet wird
- ▶ Worte mit 8 bit werden als **Byte** bezeichnet
- ▶ **Stelle:** engl. *position*
Die Lage/Position eines Zeichens innerhalb einer Zeichenkette

Flaggen-Signale

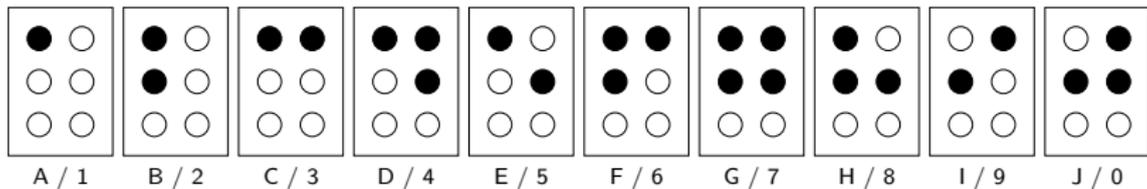


wikipedia.org/wiki/Winkeralphabet

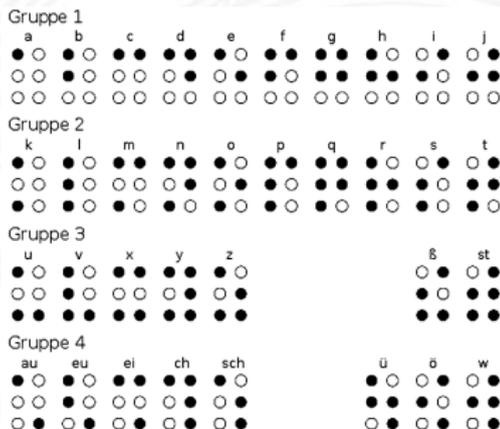
Flaggen-Alphabet



de.wikipedia.org/wiki/Flaggenalphabet



- ▶ Symbole als 2x3 Matrix (geprägte Punkte)
- ▶ Erweiterung auf 2x4 Matrix (für Computer)
- ▶ bis zu 64 (256) mögliche Symbole
- ▶ diverse Varianten
 - ▶ ein Symbol pro Buchstabe
 - ▶ ein Symbol pro Silbe
 - ▶ Kurzschrift/Steno



Codetabelle

		• kurzer Ton	– langer Ton
A	• –	S	• • •
B	– • • •	T	–
C	– • – •	U	• • –
D	– • •	V	• • • –
E	•	W	• – –
F	• • – •	X	– • • –
G	– – •	Y	– • – –
H	• • • •	Z	– – • •
I	• •	0	– – – – –
J	• – – –	1	• – – – –
K	– • –	2	• • – – –
L	• – • •	3	• • • – –
M	– –	4	• • • • –
N	– •	5	• • • • •
O	– – –	6	– • • • •
P	• – – •	7	– – • • •
Q	– – • –	8	– – – • •
R	• – •	9	– – – – •
.	• – • – • –	,	– – • • – –
?	• • – – • •	'	• – – – – •
!	– • – • – –	/	– • • – •
(– • – – •	&	• – • • •
)	– • – – • –	:	– – – • • •
&	• – • • •	;	– • – • • •
=	– • • • –	=	– • • • –
+	• – • – •	+	• – • – •
-	– • • • • –	-	– • • • • –
–	• • – – • –	–	• • – – • –
"	• – • • – •	"	• – • • – •
\$	• • • – • • –	\$	• • • – • • –
@	• – – • – •	@	• – – • – •
S-Start	– • – • –	S-Start	– • – • –
Verst.	• • • – •	Verst.	• • • – •
S-Ende	• – • – •	S-Ende	• – • – •
V-Ende	• • • – • –	V-Ende	• • • – • –
Error	• • • • • • • •	Error	• • • • • • • •
Ä	• – • –	Ä	• – • –
À	• – – • –	À	• – – • –
É	• • – • •	É	• • – • •
È	• – • • –	È	• – • • –
Ö	– – – •	Ö	– – – •
Ü	• • – –	Ü	• • – –
ß	• • • – – • •	ß	• • • – – • •
CH	– – – –	CH	– – – –
Ñ	– – • – –	Ñ	– – • – –
...		...	
SOS	• • • – – – • • •	SOS	• • • – – – • • •

▶ Eindeutigkeit Codewort: ● ● ● ● ● — ●

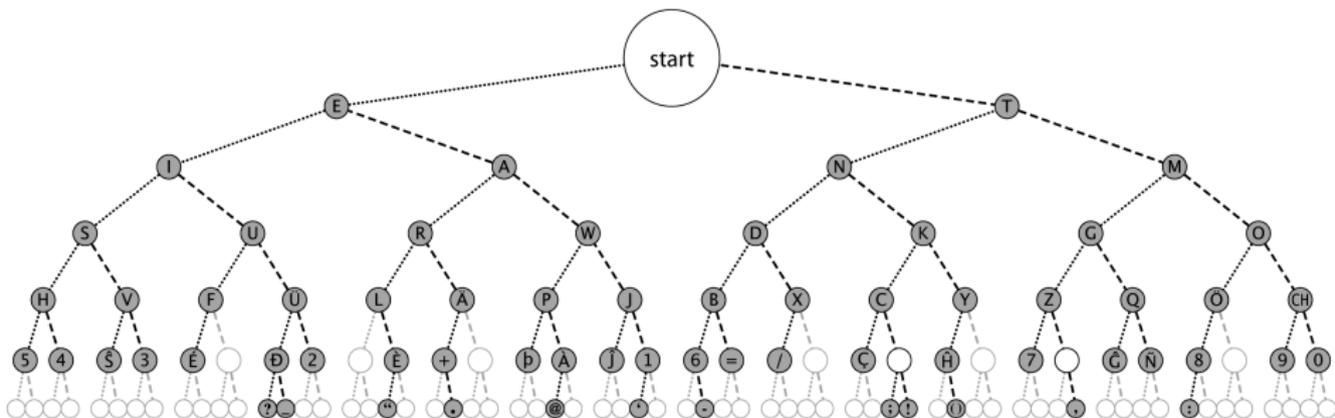
E	●
I	● ●
N	— ●
R	● — ●
S	● ● ●

- ▶ bestimmte Morse-Sequenzen sind mehrdeutig
- ▶ Pause zwischen den Symbolen notwendig

▶ Codierung

- ▶ Häufigkeit der Buchstaben = $1 / \text{Länge des Codewortes}$
- ▶ Effizienz: kürzere Codeworte
- ▶ Darstellung als Codebaum

Morse-Code: Baumdarstellung (Ausschnitt)



- ▶ Anordnung der Symbole entsprechend ihrer Codierung



ASCII

American Standard Code for Information Interchange

- ▶ eingeführt 1967, aktualisiert 1986: ANSI X3.4-1986
- ▶ viele Jahre der dominierende Code für Textdateien
- ▶ alle Zeichen einer typischen Schreibmaschine
- ▶ Erweiterung des früheren 5-bit Fernschreiber-Codes (Murray-Code)

- ▶ 7-bit pro Zeichen, 128 Zeichen insgesamt
- ▶ 95 druckbare Zeichen: Buchstaben, Ziffern, Sonderzeichen (Codierung im Bereich 21 ... 7E)
- ▶ 33 Steuerzeichen (engl: *control characters*) (0 ... 1F, 7F)

ASCII: Codetabelle

Code	...0	...1	...2	...3	...4	...5	...6	...7	...8	...9	...A	...B	...C	...D	...E	...F
0...	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1...	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2...	SP	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3...	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4...	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5...	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6...	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7...	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL

- ▶ SP = Leerzeichen, CR = carriage-return, LF = line-feed
- ▶ ESC = escape, DEL = delete, BEL = bell usw.

<https://de.wikipedia.org/wiki/ASCII>



- ▶ Erweiterung von ASCII um Sonderzeichen und Umlaute
- ▶ 8-bit Codierung: bis max. 256 Zeichen darstellbar

- ▶ Latin-1: Westeuropäisch
- ▶ Latin-2: Mitteleuropäisch
- ▶ Latin-3: Südeuropäisch
- ▶ Latin-4: Baltisch
- ▶ Latin-5: Kyrillisch
- ▶ Latin-6: Arabisch
- ▶ Latin-7: Griechisch
- ▶ usw.

- ▶ immer noch nicht für mehrsprachige Dokumente geeignet



ISO-8859-1: Codetabelle (1)

Erweiterung von ASCII für westeuropäische Sprachen

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

ISO-8859-1: Codetabelle (2)

Sonderzeichen gemeinsam für alle 8859 Varianten

Code	...0	...1	...2	...3	...4	...5	...6	...7	...8	...9	...A	...B	...C	...D	...E	...F	
0...	<i>NUL</i>	<i>SOH</i>	<i>STX</i>	<i>ETX</i>	<i>EOT</i>	<i>ENQ</i>	<i>ACK</i>	<i>BEL</i>	<i>BS</i>	<i>HT</i>	<i>LF</i>	<i>VT</i>	<i>FF</i>	<i>CR</i>	<i>SO</i>	<i>SI</i>	
1...	<i>DLE</i>	<i>DC1</i>	<i>DC2</i>	<i>DC3</i>	<i>DC4</i>	<i>NAK</i>	<i>SYN</i>	<i>ETB</i>	<i>CAN</i>	<i>EM</i>	<i>SUB</i>	<i>ESC</i>	<i>FS</i>	<i>GS</i>	<i>RS</i>	<i>US</i>	
2...	wie ISO/IEC 8859, Windows-125X und US-ASCII																
3...																	
4...																	
5...																	
6...																	
7...																	<i>DEL</i>
8...																	<i>PAD</i>
9...	<i>DCS</i>	<i>PU1</i>	<i>PU2</i>	<i>STS</i>	<i>CCH</i>	<i>MW</i>	<i>SPA</i>	<i>EPA</i>	<i>SOS</i>	<i>SGCI</i>	<i>SCI</i>	<i>CSI</i>	<i>ST</i>	<i>OSC</i>	<i>PM</i>	<i>APC</i>	
A...	wie ISO/IEC 8859-1 und Windows-1252																
B...																	
C...																	
D...																	
E...																	
F...																	

ISO-8859-2

Erweiterung von ASCII für slawische Sprachen

Code	...0	...1	...2	...3	...4	...5	...6	...7	...8	...9	...A	...B	...C	...D	...E	...F
0...	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1...	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2...	SP	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3...	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4...	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5...	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6...	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7...	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL
8...	PAD	HOP	BPH	NBH	IND	NEL	SSA	ESA	HTS	HTJ	VTS	PLD	PLU	RI	SS2	SS3
9...	DCS	PU1	PU2	STS	CCH	MW	SPA	EPA	SOS	SGCI	SCI	CSI	ST	OSC	PM	APC
A...	NBSP	Ą	ˆ	Ł	ł	Ł	Ś	ś	ˆ	Š	š	Ť	ž	SHY	Ž	ž
B...	°	ą	ˆ	ł	ł	Ł	ś	ˆ	š	š	ť	ž	ˆ	ž	ž	
C...	Ř	Á	Â	Ă	Ä	Á	Ç	Č	É	È	Ë	Ë	Í	Î	Ď	
D...	Đ	Ñ	Ň	Ó	Ô	Õ	Ö	×	Ř	Ú	Ú	Û	Ü	Ý	Ť	ß
E...	đ	á	â	ă	ä	á	ç	č	é	è	ë	ë	í	î	ď	
F...	đ	ñ	ň	ó	ô	õ	ö	÷	ř	ú	ú	û	ü	ý	ť	·

ISO-8859-15

Modifizierte ISO-8859-1 mit € (0xA4)

Code	...0	...1	...2	...3	...4	...5	...6	...7	...8	...9	...A	...B	...C	...D	...E	...F
0...	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1...	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2...	SP	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3...	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4...	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5...	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6...	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7...	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL
8...	PAD	HOP	BPH	NBH	IND	NEL	SSA	ESA	HTS	HTJ	VTS	PLD	PLU	RI	SS2	SS3
9...	DCS	PU1	PU2	STS	CCH	MW	SPA	EPA	SOS	SGCI	SCI	CSI	ST	OSC	PM	APC
A...	NBSP	ı	ç	£	€	¥	Š	§	š	©	ª	«	¬	SHY	®	¯
B...	°	±	²	³	Ž	µ	¶	·	ž	¹	º	»	Œ	œ	ÿ	ı
C...	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D...	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E...	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F...	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

- ▶ Zeichensatz des IBM-PC ab 1981
- ▶ Erweiterung von ASCII auf einen 8-bit Code
- ▶ einige Umlaute (westeuropäisch)
- ▶ Grafiksymbole

- ▶ https://de.wikipedia.org/wiki/Codepage_437
- ▶ verbesserte Version: Codepage 850, 858 (€-Symbol an 0xD5)
- ▶ Codepage 1252 entspricht (weitgehend) ISO-8859-1
- ▶ Sonderzeichen liegen an anderen Positionen als bei ISO-8859

Microsoft: Codepage 850

Code	...0	...1	...2	...3	...4	...5	...6	...7	...8	...9	...A	...B	...C	...D	...E	...F
0...		☺	☹	♥	♦	♠	♣	•	◻	◊	◼	♂	♀	♪	♫	☼
1...	▶	◀	↕	!!	¶	§	—	↕	↑	↓	→	←	↔	▲	▼	
2...		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3...	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4...	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5...	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6...	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7...	p	q	r	s	t	u	v	w	x	y	z	{		}	~	△
8...	Ç	ü	é	â	ä	à	å	ç	ê	ë	è	ï	î	í	Ä	Å
9...	É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	ø	£	Ø	×	f
A...	á	í	ó	ú	ñ	Ñ	ª	º	¿	®	¬	½	¼	¡	«	»
B...	⌘	⌘	⌘		†	Á	Â	Ã	©	¶		¶	¶	¢	¥	¶
C...	L	⊥	⊥	†	—	†	ã	Ã	ℒ	℞	⊥	⊥	⊥	=	⊥	◻
D...	ð	Ð	Ê	Ë	È	ı	í	î	ï	Ĵ	ŀ	■	■	ı	ı	■
E...	Ó	ß	Ô	Ò	õ	Õ	µ	þ	Ɔ	Ú	Û	Ù	ý	Ý	—	´
F...		±	=	¾	¶	§	÷	,	°	¨	.	¹	³	²	■	

- ▶ die meisten gängigen Codes (abwärts-) kompatibel mit ASCII
- ▶ unterschiedliche Codierung für Umlaute (soweit vorhanden)
- ▶ unterschiedliche Codierung der Sonderzeichen

- ▶ Systemspezifische Konventionen für Zeilenende
 - ▶ abhängig von Rechner- und Betriebssystem
 - ▶ Konverter-Tools: `dos2unix`, `unix2dos`, `iconv`

Betriebssystem	Zeichensatz	Abkürzung	Hex-Code	Escape
Unix, Linux, Mac OS X, AmigaOS, BSD	ASCII	<i>LF</i>	0A	<code>\n</code>
Windows, DOS, OS/2, CP/M, TOS (Atari)	ASCII	<i>CR LF</i>	0D 0A	<code>\r\n</code>
Mac OS bis Version 9, Apple II	ASCII	<i>CR</i>	0D	<code>\r</code>
AIX OS, OS 390	EBCDIC	<i>NEL</i>	15	



- ▶ zunehmende Vernetzung und Globalisierung
 - ▶ internationaler Datenaustausch?
 - ▶ Erstellung mehrsprachiger Dokumente?
 - ▶ Unterstützung orientalischer oder asiatischer Sprachen?

 - ▶ ASCII oder ISO-8859-1 reicht nicht aus
 - ▶ temporäre Lösungen konnten sich nicht durchsetzen, z.B.:
ISO-2022: Umschaltung zwischen mehreren Zeichensätzen durch Spezialbefehle (*Escapesequenzen*).
- ⇒ **Unicode** als System zur Codierung aller Zeichen aller bekannten (lebenden oder toten) Schriftsysteme

- ▶ auch abgekürzt als UCS: **Universal Character Set**
- ▶ zunehmende Verbreitung (Betriebssysteme, Applikationen)
- ▶ Darstellung erfordert auch entsprechende Schriftarten
- ▶ <http://www.unicode.org>
<http://www.unicode.org/charts>
- ▶ 1991 1.0.0: europäisch, nahöstlich, indisch
- ▶ 1992 1.0.1: ostasiatisch (Han)
- ▶ 1993 akzeptiert als ISO/IEC-10646 Standard
- ▶ ...
- ▶ 2019 12.1.0: inzwischen 137 929 Zeichen
 - ▶ Sprachzeichen, Hieroglyphen etc.
 - ▶ Symbole: Satzzeichen, Währungen (\$... ₪), Pfeile, mathematisch, technisch, Braille, Noten etc.
 - ▶ Emojis (3 187 aktuell) / Kombinationen

- ▶ ursprüngliche Version nutzt 16-bit pro Zeichen
- ▶ die sogenannte „*Basic Multilingual Plane*“
- ▶ Schreibweise hexadezimal als U+xxxx
- ▶ Bereich von U+0000 . . . U+FFFF
- ▶ Schreibweise in Java-Strings: \uxxxx
z.B. \u03A9 für Ω , \u20AC für das €-Symbol

- ▶ mittlerweile mehr als 2^{16} Zeichen
- ▶ Erweiterung um „*Extended Planes*“
- ▶ U+10000 . . . U+10FFFF

Unicode: in Webseiten (HTML)

- ▶ HTML-Header informiert über verwendeten Zeichensatz
- ▶ Unterstützung und Darstellung abhängig vom Browser
- ▶ Demo <http://kermitproject.org/utf8.html>

```
<html>
<head>
<META http-equiv="Content-Type" content="text/html;
  charset=utf-8">
<title>UTF-8 Sampler</title>

<META ...
</head>
...
```

Unicode: Demo

<http://kermitproject.org/utf8.html>

1. **English:** The quick brown fox jumps over the lazy dog.
2. **Jamaican:** Chruu, a kwik di kwik brong fox a jomp huova di liezi daag de, yu no siit?
3. **Irish:** "An bfuil do croí ag bualadh ó faitíos an grá a m'eall lena póg éada ó síl do leasa tú?" "D'fuascaill íosa Úrmac na hÓige Beannaite pór Éava agus Ádairm."
4. **Dutch:** Pa's wijze lynx bezag vroom het fikse aquaduct.
5. **German:** Falsches Üben von Xylophonmusik quält jeden größeren Zwerg. (1)
6. **German:** Im finföteren Jagdchloß am offenen Felsquellwaller patzte der affig-flatterhafte kauzig-höfliche Bäcker über feinem verfliffen kniffligen C-Xylophon. (2)
7. **Norwegian:** Blåbærsyltetøy ("blueberry jam", includes every extra letter used in Norwegian).
8. **Swedish:** Flygande bäckasiner söka strax hwila på mjuka tuvor.
9. **Icelandic:** Sævör grét áðan því úlpan var ónýt.
10. **Finnish:** (5) Törkylempijävongahdus (This is a perfect pangram, every letter appears only once. Translating it is an art on its own, but I'll say "rude lover's yelp". :-D)
11. **Finnish:** (5) Albert osti fagotin ja töräytti puhkuvan melodian. (Albert bought a bassoon and hooted an impressive melody.)
12. **Finnish:** (5) On sangen hauskaa, että polkupyörä on maanteiden jokapäiväinen ilmiö. (It's pleasantly amusing, that the bicycle is an everyday sight on the roads.)
13. **Polish:** Pchnąc w tę łódź jeża lub osiem skrzyń fig.
14. **Czech:** Přilíš žluťoučký kůň úpěl ďábelské ódy.
15. **Slovak:** Starý kôň na hrbe knih žuje tiško povädnuté ruže, na stípe sa datel učí kvákať novú ódu o živote.
16. **Slovenian:** Šerif bo za domačo vajo spet kahal žgance.
17. **Greek (monotonic):** ξεσκεπάζω την ψυχοφθόρα βδελυγμία
18. **Greek (polytonic):** ξεσκεπάζω τήν ψυχοφθόρα βδελυγμία
19. **Russian:** Съешь же ещё этих мягких французских булок да выпей чаю.
20. **Russian:** В чашках юга жил-был цитрус? Да, но фальшивый экземпляр! ёь.
21. **Bulgarian:** Жълтата дюля беше щастлива, че пухът, който цъфна, замръзна като гън.
22. **Sami (Northern):** Vuol Ruota geđgeđiid leat máhga luosa ja čuovžža.
23. **Hungarian:** Árvízűrő tükörfűrógép.
24. **Spanish:** El pingüino Wenceslao hizo kilómetros bajo exhaustiva lluvia y frío, añoraba a su querido cachorro.
25. **Spanish:** Volé cigüeña que jamás cruzó París, exhibe flor de kiwi y atún.
26. **Portuguese:** O próximo voo à noite sobre o Atlântico, pde frequentemente o único médico. (3)
27. **French:** Les naïfs ægithales hâtifs pondant à Noël où il gèle sont sûrs d'être déçus en voyant leurs drôles d'œufs abimés.
28. **Esperanto:** Eĥoŝanĝo ŝuĵaŭde.
29. **Hebrew:** זה כריך סתם לשמוע איך תנצח קרפר ען טוב בגן.
30. **Japanese (Hiragana):**

いろはにほへど ちりぬるを
わがよたれぞ つねならむ
うみのおくやま けふこえて
あさきゆめみじ 桑ひもせず (4)

Unicode: Demo (cont.)

<http://kermitproject.org/utf8.html>

[Šota Rustaveli](#)'s Vep̄xis T̄qaosani, Ⴈ, *The Knight in the Tiger's Skin* (Georgian):

ვეპ̄ხის ტყაოსანი შოთა რუსთაველი

ღმერთის შემდეგდრე, ნუთუ კვლა დამხსნას სოფლისა შრომისა, ცეცხლს, წყალსა და მიწასა,
ჰაერთა თანა მრომისა; მომცნეს ფრთენი და აღფურინდე, მივჰხუდე მას ჩემსა ნდომისა, დღისით
და ღამით ვჰხუდედი მზისა ელვათა კრთომისა.

Tamil poetry of Subramaniya Bharathiyar: சுப்ரமணிய பாரதியார் (1882-1921):

யாமறிந்த மொழிகளிலே தமிழ்மொழி போல் இனிதாவது எங்கும் காணோம்,
பாமரராய் விலங்குகளாய், உலகனைத்தும் இகழ்ச்சிசொலப் பான்மை கெட்டு,
நாமமது தமிழ்ரெனக் கொண்டு இங்கு வாழ்ந்திடுதல் நன்றோ? சொல்லீர்!
தேமதுரத் தமிழோசை உலகமெலாம் பரவும்வகை செய்தல் வேண்டும்.

- ▶ Zeichen im Bereich U+0000 bis U+007F wie ASCII
www.unicode.org/charts/PDF/U0000.pdf
- ▶ Bereich von U+0100 bis U+017F für Latin-A
Europäische Umlaute und Sonderzeichen
www.unicode.org/charts/PDF/U0100.pdf
- ▶ viele weitere Sonderzeichen ab U+0180
Latin-B, Latin-C usw.



Vielfältige Auswahl von Symbolen und Operatoren

- ▶ griechisch www.unicode.org/charts/PDF/U0370.pdf
- ▶ letterlike Symbols www.unicode.org/charts/PDF/U2100.pdf

- ▶ Pfeile www.unicode.org/charts/PDF/U2190.pdf
- ▶ Operatoren www.unicode.org/charts/PDF/U2A00.pdf
- ▶ ...

- ▶ Dingbats www.unicode.org/charts/PDF/U2700.pdf

Chinesisch (traditional/simplified), Japanisch, Koreanisch

- ▶ U+3400 bis U+4DBF
www.unicode.org/charts/PDF/U3400.pdf
- ▶ U+4E00 bis U+9FCF
www.unicode.org/charts/PDF/U4E00.pdf

Unicode: Java2D Fontviewer

The screenshot shows the Font2DTest application window. The title bar reads "Font2DTest". The menu bar includes "File" and "Option". The main interface contains several settings:

- Font: Arial
- Size: 20
- Font Transform: None
- Range: Arabic
- Style: Plain
- Graphics Transform: None
- Method: drawString
- Text to use: Unicode Range
- LCD contrast: 100 120 140 160 180 200 220 240 (slider at 140)
- Antialiasing: DEFAULT
- Fractional metrics: DEFAULT

Below the settings is a grid of 16 columns and 10 rows of Arabic characters. The characters are arranged in a grid, showing various forms of letters and symbols. The text "Pointing to Unicode 0619" is visible at the bottom left of the grid area.

Oracle [JavaD]: JDK demos and samples .../demo/jfc/Font2DTest

- ▶ 16-bit für jedes Zeichen, bis zu 65 536 Zeichen
 - ▶ schneller Zugriff auf einzelne Zeichen über Arrayzugriffe (Index)
 - ▶ aber: doppelter Speicherbedarf gegenüber ASCII/ISO-8859-1
 - ▶ Verwendung u.a. in Java: Datentyp `char`

 - ▶ ab Unicode 3.0 mehrere *Planes* zu je 65 536 Zeichen
 - ▶ direkte Repräsentation aller Zeichen erfordert 32-bit/Zeichen
 - ▶ vierfacher Speicherbedarf gegenüber ISO-8859-1

 - ▶ bei Dateien ist möglichst kleine Dateigröße wichtig
- ⇒ Codierung als UTF-8 oder UTF-16

Zeichen	Unicode	Unicode binär	UTF-8 binär	UTF-8 hexadezimal
Buchstabe y	U+0079	00000000 01111001	01111001	79
Buchstabe ä	U+00E4	00000000 11001000	11000011 10100100	C3 A4
Zeichen für eingetragene Marke ®	U+00AE	00000000 10101110	11000010 10101110	C2 AE
Eurozeichen €	U+20AC	00100000 10101100	11100010 10000010 10101100	E2 82 AC
Viollinschlüssel 	U+1D11E	00000001 11010001 00011110	11110000 10011101 10000100 10011110	F0 9D 84 9E

<https://de.wikipedia.org/wiki/UTF-8>

- ▶ effiziente Codierung von „westlichen“ Unicode-Texten
- ▶ Zeichen werden mit variabler Länge codiert, 1...4-Bytes
- ▶ volle Kompatibilität mit ASCII

Unicode-Bereich (hexadezimal)	UTF-Codierung (binär)	Anzahl (benutzt)
0000 0000 - 0000 007F	0*** ****	128
0000 0080 - 0000 07FF	110* **** 10** ****	1 920
0000 0800 - 0000 FFFF	1110 **** 10** **** 10** ****	63 488
0001 0000 - 0010 FFFF	1111 0*** 10** **** 10** **** 10** ****	bis 2^{21}

- ▶ untere 128 Zeichen kompatibel mit ASCII
- ▶ Sonderzeichen westlicher Sprachen je zwei Bytes
- ▶ führende Eins markiert Multi-Byte Zeichen
- ▶ Anzahl der führenden Einsen gibt Anz. Bytegruppen an
- ▶ Zeichen ergibt sich als Bitstring aus den ***...*
- ▶ theoretisch bis zu sieben Folgebytes a 6-bit: max. 2^{42} Zeichen



Locale: die Sprach-Einstellungen und Parameter

- ▶ auch: `i18n` („internationalization“)
 - ▶ Sprache der Benutzeroberfläche
 - ▶ Tastaturlayout/-belegung
 - ▶ Zahlen-, Währungs-, Datums-, Zeitformate
 - ▶ Linux/POSIX: Einstellung über die Locale-Funktionen der Standard C-Library (Befehl `locale`)
- Java: `java.util.Locale`
- Windows: Einstellung über System/Registry-Schlüssel

- ▶ Umwandeln von ASCII-Texten (z.B. Programm-Quelltexte) zwischen DOS/Windows und Unix/Linux Maschinen

- ▶ Umwandeln von a.txt in Ausgabedatei b.txt:

```
dos2unix -c ascii -n a.txt b.txt
```

```
dos2unix -c iso -n a.txt b.txt
```

```
dos2unix -c mac -n a.txt b.txt
```

- ▶ Umwandeln von Unix nach DOS/Windows, Codepage 850:

```
unix2dos -850 -n a.txt b.txt
```



Das „Schweizer-Messer“ zur Umwandlung von Textcodierungen

▶ Optionen

- ▶ `-l` Liste der unterstützten Codierungen ausgeben
- ▶ `-f <encoding>` Codierung der Eingabedatei
- ▶ `-t <encoding>` Codierung der Ausgabedatei
- ▶ `-o <filename>` Name der Ausgabedatei

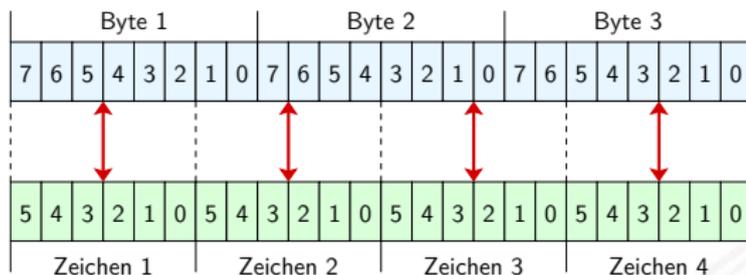
▶ Beispiel

```
iconv -f iso-8859-1 -t utf-8 -o foo.utf8.txt foo.txt
```

Übertragung von (Binär-) Dateien zwischen verschiedenen Rechnern?

- ▶ SMTP (Internet Mail-Protokoll) verwendet 7-bit ASCII
 - ▶ bei Netzwerk-Übertragung müssen alle Rechner/Router den verwendeten Zeichensatz unterstützen
- ⇒ Verfahren zur Umcodierung der Datei in 7-bit ASCII notwendig
- ⇒ etabliert ist das **Base64** Verfahren (RFC 2045)
- ▶ alle E-Mail Dateianhänge und 8-bit Textdateien
 - ▶ Umcodierung benutzt nur Buchstaben, Ziffern und drei Sonderzeichen
 - ▶ Daten werden byteweise in ASCII Symbole umgesetzt

1. Codierung von drei Bytes als vier 6-bit Zeichen



- ▶ $3 \times 8\text{-bit} \Leftrightarrow 4 \times 6\text{-bit}$
- ▶ 6-bit Binärwerte: 0 ... 63
- ▶ nutzen 64 (von 128) 7-bit ASCII Symbolen

2. Zahl ASCII Zuordnung der ASCII-Zeichen

0 ... 25	A ... Z
26 ... 51	a ... z
52 ... 61	0 ... 9
62	+
63	/
=	Füllzeichen, falls Anz. Bytes nicht durch 3 teilbar
CR	Zeilenumbruch (opt.), meistens nach 76 Zeichen

Base64-Codierung: Prinzip (cont.)

Text content	M	a	n	
ASCII	77	97	110	
Bit pattern	0 1 0 0 1 1 0 1	0 1 1 0 0 0 0 1	0 1 1 0 1 1 1 0	
Index	19	22	5	46
Base64-encoded	T	W	F	u

- ▶ drei 8-bit Zeichen, neu gruppiert als vier 6-bit Blöcke
- ▶ Zuordnung des jeweiligen Buchstabens/Ziffer
- ▶ ggf. =, == am Ende zum Auffüllen
- ▶ Übertragung dieser Zeichenfolge ist 7-bit kompatibel
- ▶ resultierende Datei ca. 33% größer als das Original

- ▶ in neueren Java Versionen (> 1.8) im JDK enthalten
Module `java.base`, Package `java.util`:
`Base64Encoder`, bzw. `Base64Decoder`
- ▶ diverse andere Packages
 - ▶ Apache Commons Codec <http://commons.apache.org/proper/commons-codec>
`org.apache.commons.codec.binary.Base64InputStream`
`org.apache.commons.codec.binary.Base64OutputStream`
 - ▶ JAXB (Java Architecture for XML Binding)
in `javax.xml.bind.DatatypeConverter`
`parseBase64Binary`, `printBase64Binary`
Beispiel in *Java ist auch eine Insel* [Ull18]
openbook.rheinwerk-verlag.de/javainsel/04_008.html#u4.7.4
 - ▶ ...

[Uni] The Unicode Consortium; Mountain View, CA.
home.unicode.org, unicode.org/main.html

[JavaI] Oracle Corporation; Redwood Shores, CA.
The Java Tutorials – Trail: Internationalization.
docs.oracle.com/javase/tutorial/i18n

[JavaD] Oracle Corporation: *Java SE Downloads.*
www.oracle.com/technetwork/java/javase/downloads

[Ull18] C. Ullenboom: *Java ist auch eine Insel – Einführung, Ausbildung, Praxis.* 14. Auflage, Rheinwerk Verlag GmbH, 2018. ISBN 978-3-8362-6721-2

12. Auflage (Java 8) unter
openbook.rheinwerk-verlag.de/javainsel, bzw.
www.tutego.de/javabuch