

1. Vorüberlegungen

Alle Daten und Programme sind in Computern auf dieselbe Weise gespeichert, nämlich binär. Das bedeutet, sie lassen sich mit zwei Symbolen darstellen. Ob die Speicherung magnetisch (Festplatte), optisch (CD/DVD) oder elektrisch (USB-Stick) ist, spielt dabei keine Rolle.

Neben Zahlen (z.B. im Dualsystem) müssen aber auch Zeichen wie Buchstaben und Ziffern dargestellt und so gespeichert werden, dass sie auch nach einer Übertragung zwischen zwei Geräten noch lesbar sind. Also muss es Festlegungen geben, wie Zeichen einheitlich im Speicher darzustellen sind.

a. Codelänge:

Schätze ab, wie viele Zeichen mindestens kodiert werden müssen:

- moderne Tastatur mit Buchstaben, Zahlen, Zeichen, nicht druckbare Zeichen (ENTER, Cursortasten usw.)
- Sonderzeichen verschiedener Sprachen mit lateinischem Alphabet (Umlaute, Accents, Wellen usw.)

Wie viele Stellen muss ein Bitmuster haben, um diese Anzahl verschiedener Zeichen zu kodieren?

Gesamtzahl:

Speichergröße:

b. Suche aus der Tabelle auf der rechten Seite heraus:

Gruppe von Zeichen				ASCII-Wert(e)	
				Hexadezimal	Dezimal
Großbuchstaben A-Z					
Kleinbuchstaben a-z					
Ziffern 0-9					
Leerzeichen		Backspace		Entfernen	
H:	D:	H:	D:	H:	D:

c. Der bekannteste Standard ist wohl der ASCII mit einer Länge von 7 Bit pro Zeichen.

- Notiere die Anzahl der darstellbaren verschiedenen Zeichen.

- Nenne fehlende Buchstaben unserer Sprache (erstmal ohne Codewerte).

Zeichen

Code_{dez}

Code_{hex}

2. Datentypen

a. In einer Speicherzelle befinde sich das Bitmuster 0110 1101. Was bedeutet es ...

- ... als Dezimalzahl,
- ... als Zeichen?

b. Wie könnte ein Bitmuster noch interpretiert werden?

c. Woran erkennt ein Programm, welche Interpretation gemeint ist?

3. ASCII am Rechner nutzen

a. Öffne Notepad++ und teste die folgende Tastenkombination: Alt+65

- Halte dazu die Alt-Taste gedrückt, bis du die 65 (auf dem Nummerblock) fertig getippt hast. Welches Zeichen erscheint, nachdem du die Alt-Taste wieder loslässt?
- Teste weitere Tastenkombinationen.

b. Teste Codes, die größer als die Zahl bei 1c) sind. Findest du dort genannte Zeichen? Notiere ggf. die Kodierungen als Dezimalzahl.

Alt+

4. Recherche:

Fertige Notizen an zu Jahreszahlen, Anzahl der Bits, der Zeichen usw.

a. ASCII-Code

b. UTF-8 (Unicode)

Material:

Hinweis zur Tabelle: Die Zeichen in der Tabelle werden durchnummeriert, begonnen bei 0. Damit hat das Zeichen @ die Nummer 40 (hexadezimal) bzw. 64 (dezimal). Das Zeichen # steht in Zeile 2 und Spalte 3, wird damit durch die die Hexadezimalzahl 23 bzw. die Dezimalzahl 35 dargestellt. Das erste druckbare Zeichen ist 20 (dezimal 32): SP, die Leertaste.

ASCII-Zeichentabelle, hexadezimale Nummerierung

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

Abbildung 1:

https://de.wikipedia.org/wiki/American_Standard_Code_for_Information_Interchange

Übung zur Wiederholung

- 1) In welchem Land wurde der ASCII festgelegt?

- 2) Nenne zwei Gruppen von Zeichen, die schon im ursprünglichen 7-Bit-ASCII enthalten waren.

- 3) Wie viele Zeichen wurden insgesamt mit diesen 7 Bit kodiert? (mit Begründung)

- 4) Nenne ein Zeichen, das später erst in den ASCII aufgenommen wurde:

Übung zur Wiederholung

- 1) In welchem Land wurde der ASCII festgelegt?

- 2) Nenne zwei Gruppe von Zeichen, die schon im ursprünglichen 7-Bit-ASCII enthalten waren.

- 3) Wie viele Zeichen wurden insgesamt mit diesen 7 Bit kodiert? (mit Begründung)

- 4) Nenne ein Zeichen, das später erst in den ASCII aufgenommen wurde:
