

weiter zu "Informationsdarstellung im Rechner"

bisher: Zahlen, Wahrheitswerte
aber man braucht auch: Zeichen, Texte

Zeichen (characters):

Buchstaben, Ziffern, Sonderzeichen

meist pro Zeichen 1 Byte

ASCII-Code ISO-646:

American Standard Code for Information Interchange

z.B.

Hex 30	=	dezimal	48	=	Ziffer 0
Hex 31	=	dez.	49	=	Ziffer 1
...					
Hex 3A	=	dez.	58	=	:
Hex 3B	=	dez.	59	=	;
...					
Hex 41	=	dez.	65	=	A
Hex 42	=	dez.	66	=	B
...					
Hex 61	=	dez.	97	=	a

→ Tabelle

in C: Datentyp `char` (= *character*)

Unicode:

2 Byte (16 Bit) - Code für multilinguale
Textverarbeitung

kann 65 536 Zeichen darstellen

u.a.: 27 786 chinesisch-jap.-koreanische Zeichen

11 172 Hangeul-Zeichen (Korea)

altnordische Runen

Tibetisch

Cherokee

.....

Unicode Escape-Sequenz:

`\u0041` = A (z.B. in Java verwendbar)

Hexadezimal-Code

UTF-8: Universal Transformation Format

Zeichen belegen variable Anzahl von Bytes

→ für Texte mit vielen ASCII-Zeichen kürzer als

Unicode

Zeichenketten (Strings; Wörter):

Ketten aus 0, 1, ..., n Zeichen

"" leeres Wort, Länge 0

"DIES IST EIN STRING" Länge 19

in C:

'x' : einzelnes Zeichen **x** (character)

"abc" : String

"x" : String der Länge 1,
ist nicht dasselbe wie 'x' !

Grund-Datentypen in C:

char

int

float

double

void = "leerer Typ"

Modifizierer für int:

signed

unsigned

short

long

(short int = short, long int = long, signed int = int)