Westfälische
Hochschule
Gelsenkirchen Bocholt Recklinghausen

Vorlesung „Grundlagen der Informatik und Programmierung 1“

Datentypen und ihre Darstellung

Zeichen und Zeichenketten

Prof. Dr. Tom Vierjahn

Visual Computing (https://vc.w-hs.de)
Fachbereich Wirtschaft und Informationstechnik – Campus Bocholt

Wintersemester 2020/21

Veröffentlicht unter der Creative-Commons-Lizenz
Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

Zeichen

- Zeichen/Buchstaben werden als (vorzeichenlose) Zahl gespeichert
- Datentyp char oder unsigned char (1 Byte)
- Zeichen 0x00 bis 0x7f entsprechen dem ASCII-Zeichensatz:

	…0	…1	…2	…3	…4	…5	…6	…7	…8	…9	…a	…b	…c	…d	…e	…f
0…	∅	…	…	…	EOT	…	…	…	…	TAB	LF	…	…	CR	…	…
1…	…	…	…	…	…	…	…	…	…	…	…	ESC	…	…	…	…
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

- Zeichen 0x80 bis 0xff sind sprachabhängig

Zeichen

Ein- und Ausgabe

Westfälische Hochschule

- Zeichenkonstanten stehen in einfachen Anführungszeichen (').
- Zeilenwechsel (0x0a): '\n'
- Formatanweisung: "%c"

Example

```
char c = 'a';
char input;
printf("Bitte einen Buchstaben eingeben (z. B. %c): ", c);
scanf("%c", &input);
printf("Die Eingabe lautete: %c \n", input);
```

Hinweis:
Beim Lesen mit "%c" wird nur ein einzelnes Zeichen eingelesen.
Alle weiteren Zeichen, inkl. des unvermeidlichen Zeilenwechsels,
werden bei ggf. folgenden Aufrufen von scanf gelesen.

Zeichenketten

Westfälische Hochschule

- Zeichenketten werden in einem Array gespeichert.
- Basistyp: `char` oder `unsigned char`
- Null-terminiert (ASCII-Zeichen 0x00 / ∅)
- Terminator muss ins Array passen.
- Konstante Zeichenketten stehen in doppelten Anführungszeichen (").

Example: Unsere erste Zeichenkette

Im Programmcode:

```
char text[] = "GIP 1";
```

Im Speicher:

Zeichenketten

Ein- und Ausgabe

Westfälische Hochschule

- Formatanweisung: "%s"
- Bei Eingabe kein & vor Variablenname nötig/erlaubt; klären wir später.
- Bei Eingabe wird die Array-Größe nicht überprüft.
- Zu lange Eingabe führt zu Absturz oder unbemerkt zu Problemen.

```
char input[100];

printf("Bitte Wort eingeben: ");
scanf("%s", input);

printf("Die Eingabe lautete: %s \n", input);
```

kein & vor Variablenname

Zeichenketten

Zugriff, Veränderung, etc.

Westfälische Hochschule

- Zugriff auf/Veränderung von Zeichen in Zeichenketten wie bei anderen Arrays
- Maximale Länge: Arraygröße − 1
- Terminator (`\0`) muss korrekt gesetzt werden.
- Zugriff außerhalb der Arraygrenzen: Absturz oder unbemerkte Probleme

Wichtig: Verantwortung liegt beim Programmierer.

- Zeichenketten können nicht mit = kopiert werden
- Kopie muss zeichenweise erfolgen – inkl. Terminator.
- Zeichenketten können nicht mit == verglichen werden.
- Vergleich muss zeichenweise erfolgen – inkl. Terminator.

Zeichenketten

Was macht dieser Code?

```
char input[100];
printf("Bitte Wort eingeben: ");
scanf("%s", input);

for (int i = 0; i < 99; ++i) {
  if (input[i] >= 0x61) {
    if (input[i] <= 0x7a) {
      input[i] -= 0x20;
    }
  }
}

printf("Verarbeitete Eingabe: %s \n", input);
```

Zusammenfassung

Westfälische Hochschule

- einzelne Zeichen
- Zeichenketten/Strings = Zeichen-Arrays
- Terminator
- Ein- und Ausgabe

Prof. Dr. Tom Vierjahn

- tom.vierjahn@w-hs.de

Visual Computing

- https://vc.w-hs.de
- VisualComputingWH
- Visual Computing WH
- @VisComputingWH

Westfälische Hochschule
Fachbereich Wirtschaft und Informationstechnik
Campus Bocholt

Veröffentlicht unter der Creative-Commons-Lizenz
Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)