

Kapitel 3

Ein- & Ausgabe

Ein- und Ausgabe

- Einen gespeicherten Wert auf die Standardausgabe (üblicherweise der Monitor) kann man auf drei verschiedene Arten ausgeben:
 - schreiben des Namens der Variable ohne abschließendes Semikolon
 - durch den Befehl *disp()*
 - durch den Befehl *fprintf()*
 - Für die Eingabe vom Standardeingabegerät (Tastatur) wird der Befehl *input()* verwendet.

Ausgabe - durch Name

- Will man nur eine Variable ausgeben, dann genügt es, deren Namen zu schreiben ohne ein abschließendes Semikolon.
- Es wird dann die Variable mit deren Wert/Werten ausgegeben.
- Es kann hierbei **kein** zusätzlicher Text ausgegeben werden!

```
> a = 1;
```

```
> a
```

```
a =
```

```
1
```

Ausgabe - disp

- Der Befehl *disp()* gibt nur den Wert/die Werte der Variable aus.
- Der Variablenname wird nicht ausgegeben.

```
> a = 1;
```

```
> disp(a);
```

```
1
```

Ausgabe - disp

- Der Befehl *disp()* erlaubt es auch, mehrere Ausgabetexte zu verketteten:

```
> a = 1;
```

```
> disp("Das Ergebnis lautet: " + a + " cm");
```

```
Das Ergebnis lautet: 1 cm
```

Wichtig: Das klappt nur, wenn " benutzt wird (und nicht: ').

Ausgabe - fprintf

- Die flexibelste Methode, um etwas auszugeben hat man, indem man den Befehl *fprintf()* benutzt.
- *fprintf()* kann mit " oder auch ' benutzt werden.
- Hier ist es möglich, Variablen und Text zusammen auszugeben, ohne versehentlich eine Rechnung auszuführen.

```
> a = 1;
```

```
> fprintf("Die Variable a hat den Wert: %i", a);
```

```
> fprintf('Die Variable a hat den Wert: %i', a);
```

```
Die Variable a hat den Wert: 1
```

```
Die Variable a hat den Wert: 1
```

Ausgabe - fprintf

- Nach dem Text, der in Anführungszeichen steht, wird die Variable aufgelistet, die ausgegeben werden soll. Es können auch mehrere Variablen aufgelistet werden.
- Für jede Variable muss in den Anführungszeichen eine Angabe vorhanden sein, als was diese Variable ausgegeben wird. Also: Ist die Variable z.B. ein *int*, *double* oder *string*.

Ausgabe - fprintf

- Die wichtigsten Ausgabeformate sind:

<code>%c</code>	einzelnes Zeichen
<code>%s</code>	Zeichenkette (string)
<code>%i</code> oder <code>%d</code>	ganze Zahl
<code>%f</code>	Gleitpunktzahl
<code>%o</code>	Ausgabe der Zahl als Oktalzahl
<code>%x</code>	Ausgabe der Zahl als Hexadezimalzahl

Ausgabe - fprintf

- Man kann zusätzlich beim Ausgabeformat auch die Genauigkeit (default: 6 Nachkommastellen) und die maximale Anzahl an verfügbarem Platz festlegen:

```
> a = 3.14152;
```

```
> fprintf("Die Variable a hat den Wert: %.2f", a);
```

```
> fprintf("Die Variable a hat den Wert: %10f", a);
```

```
> fprintf("Die Variable a hat den Wert: %10.2f", a);
```

```
Die Variable a hat den Wert: 3.14
```

```
Die Variable a hat den Wert: 3.141520
```

```
Die Variable a hat den Wert: 3.14
```

Ausgabe - fprintf

- Im Ausgabebetext können Steuerzeichen (Escape-Sequenzen) verwendet werden:

<code>\a</code>	Ton
<code>\b</code>	Backspace
<code>\f</code>	Form feed (Zeilenvorschub)
<code>\n</code>	New line (Zeilenumbruch)
<code>\r</code>	Carriage return (Wagenrücklauf)
<code>\t</code>	Tabulator
<code>\\</code>	Backslash
<code>"</code>	Single quotation mark (Hochkomma)
<code>"""</code>	Double quotation mark (Anführungszeichen)
<code>%%</code>	Percent character (Prozentzeichen)

Ausgabe - fprintf

```
fprintf("\tDies ist \n\tein Beispiel \n\tmit \n\tEscape-Sequenzen.");
```

Dies ist

ein Beispiel

mit

Escape-Sequenzen.

Ausgabe - fprintf

- *fprintf()* mit " bzw. ':

```
a = 3.14152;
```

```
fprintf("Die Variable ""a"" hat den Wert: %f", a);
```

```
fprintf('Die Variable "a" hat den Wert: %f', a);
```

```
Die Variable "a" hat den Wert: 3.14152
```

```
Die Variable 'a' hat den Wert: 3.14152
```

Möchte man " oder ' ausgeben, muss man es evtl. verdoppeln.
Je nachdem, ob es selber wieder in " " oder in ' ' steht.

Ausgabe - sprintf

- Möchte man etwas nicht auf den Bildschirm ausgeben, sondern in einem *string*, dann benutzt man den Befehl *sprintf()*.
- Er ist genauso, wie *fprintf()* zu benutzen, wird aber eben nicht auf dem Bildschirm, sondern in eine Variable ausgegeben.

```
a = 3.14152;  
text = sprintf("Die Variable ""a"" hat den Wert: %f", a);  
disp(text);
```

```
Die Variable "a" hat den Wert: 3.14152
```

Eingabe - input

- Zum Einlesen von Werten über die Tastatur, wird der Befehl *input()* benutzt.
- Es wird unterschieden zwischen numerischer Eingabe und einer Eingabe als string.

```
x = input("Bitte einen Wert eingeben:");
```

```
x = input("Möchten Sie fortfahren ? (ja/nein)", "s");
```

```
> Bitte einen Wert eingeben: 2
```

```
> Möchten Sie fortfahren ? (ja/nein) nein
```

Eingabe - numerisch

- Bei der numerischen Eingabe wertet *input()* den eingegebenen Ausdruck zusätzlich aus.

```
x = input("Bitte einen Wert eingeben: ");  
disp(x);
```

```
> Bitte einen Wert eingeben: magic(2)  
1 3  
4 2
```

Eingabe - string

- Bei der Eingabe als string, übernimmt *input()* den eingegebenen Wert **ohne** ihn auszuwerten.

```
x = input("Möchten Sie fortfahren (ja/nein) ? ", "s");  
disp(x);
```

```
> Möchten Sie fortfahren (ja/nein) ?  nein  
nein
```