

## Fragen zu „Objektorientierte Programmierung mit MATLAB“ (c) Ulrich Stein

### 2. Programmieren mit MATLAB

*Was sind Variablen?*

Variablen sind Objekte im Speicher des Rechners, mit einem Namen zur Identifizierung und einem Datentyp, der ihre Verwendungsmöglichkeit und den nötigen Speicherplatz festlegt.

*Welche Zeichen sind für einen Variablennamen erlaubt?*

Die Variablennamen beginnen mit einem Buchstaben, dann können weitere Buchstaben, Zahlen oder das „\_“-Zeichen folgen.

*Was sind die wichtigsten Standard-Datentypen in MATLAB?*

Die wichtigsten Datentypen in MATLAB sind *double* (reelle Zahlen), *int32* (ganze Zahlen) und *char* (Texte).

*Was ist ein Feld (Array)?*

Ein Array ist eine durchnummerierte Folge von Daten, die in Zeilen und Spalten organisiert sind.

*Wie greift man auf die Elemente eines Arrays zu?*

Ein Element des Arrays *A* spricht man durch Angabe der Zeile *z* und der Spalte *s* an, in der das Element steht, in der Form *A(z,s)*.

*Welche Funktionen informieren über die Größe eines Arrays?*

Die Funktion *size* gibt als Zahlenpaar [*z,s*] die Anzahl der Zeilen *z* und Spalten *s* zurück. Die Funktion *length* liefert die Anzahl der Elemente eines eindimensionalen Arrays.

*Was ist ein Cell-Array?*

Ein Cell-Array ist eine Verallgemeinerung des gewöhnlichen (numerischen) Arrays, das beliebige Arten von Datenfolgen verwalten kann. Der Zugriff auf die Daten im Cell-Array erfolgt mit Hilfe von geschweiften Klammern, also z.B. als *A{z,s}*.

*Wie wird der „:“-Operator angewendet?*

Der „:“-Operator erzeugt ein Feld von Zahlen, beginnend mit dem Startwert, im Abstand eines (optionalen) Inkrements, bis zu einem Endwert, z.B. für Startwert = 10, Inkrement = 2, Endwert = 20:

```
>> x = 10:2:20;  
x = 10 12 14 16 18 20
```

*Wie funktioniert der linspace-Operator?*

Der *linspace*-Operator erzeugt ebenfalls ein Feld von Zahlen, beginnend mit dem Startwert, bis zu einem Endwert, aufgeteilt in *n* Zahlen, z.B. für Startwert = 10, Endwert = 14, Anzahl *n* = 3:

```
>> x = linspace( 10, 14, 3 )  
x = 10 12 14
```

*Wie erzeugt man in MATLAB einen Text?*

Ein Text wird in MATLAB in einfache Anführungszeichen gesetzt, z.B.

```
>> t = 'Hallo'
```

*Wie werden Ausdrücke ausgewertet?*

Innerhalb einer Zeile erfolgt die Auswertung zuerst von innen nach außen, wie Sie es bei Klammern in der Mathematik gewohnt sind, und danach in den Ausdrücken, wie beim Lesen, von links nach rechts.

*Was ist eine Wertzuweisung?*

Mit einer Wertzuweisung werden Daten an eine Variable übergeben. Die Variable steht links vor einem Gleichheitszeichen, z.B. eine Zuweisung an x

```
>> x = 3 * 4
```

*Wie hat man die Anweisung „n = n + 1;“ zu verstehen?*

Hierbei wird der Variablen *n* der Wert zugewiesen, der sich aus den (vor der Zuweisung bestehenden) Daten auf der rechten Seite berechnet. Bei der Berechnung von „n + 1“ wird also der aktuelle Wert von *n* ausgelesen und dazu die Zahl 1 addiert. Dieser Wert wird anschließend der Variablen *n* zugewiesen, wodurch der vorher bestehende Wert überschrieben wird, was letztendlich als Effekt den ursprünglichen Wert von *n* um 1 erhöht.

*Wie ist der Kopf einer MATLAB-Funktion (die Signatur) aufgebaut?*

Der Kopf einer Funktion in MATLAB lautet

```
function r = fname( x, y, ... )
```

Mit dem Kenner *function* startet die Deklaration, dann folgt der Name des Rückgabeparameters, hier *r*. Nach dem „=-Zeichen folgt der Name der Funktion, hier *fname*, und anschließend die Liste der Eingangsparameter, hier *x, y, ...*

*Mit welchen Funktionen kann man Texte und Zahlen auf dem Bildschirm ausgeben?*

MATLAB kennt zwei Funktionen zur Ausgabe von Zeichen auf dem Bildschirm: Die einfache Ausgabefunktion *disp* und die Funktion *fprintf* zur formatierten Ausgabe.

*Welche Funktion liest Daten von der Tastatur?*

Die Funktion *input* liest Zahlen oder Text von der Tastatur. *input* besitzt den optionalen Parameter 's'.

*Welche Vergleichsoperatoren gibt es?*

MATLAB besitzt folgende Vergleichsoperatoren:

```
<      kleiner
<=     kleiner oder gleich
>      größer
>=     größer oder gleich
==     gleich
~=     ungleich
```

*Wie formuliert man eine Auswahl (Alternative)?*

Die Alternativ erfolgt über eine if-elseAbfrage mit der Syntax:

```
if( logischer_ausdruck )
    Anweisungen 1
```

```
else
    Anweisungen 2
end
```

**Wie lautet die Syntax einer Zählschleife?**

Die Syntax der Zählschleife lautet:

```
for( index = Startwert : Endwert )
    Anweisungen
end
```

**Wie ist eine Doppelschleife aufgebaut?**

Doppelschleifen benötigt man bei mehrdimensionalen Problemen. Hierzu werden zwei (oder auch mehr) Zählschleifen verschachtelt, z.B.

```
for( m = 1:10 )           % nacheinander alle Zeilen zu Nummer m
    for( n = 1:10 )       % alle Spalten zu fester Zeile m
        Anweisungen, die m und n verwenden
    end % der inneren Schleife zu n
    evtl. weitere Anweisungen zu m
end % der äußeren Schleife zu m
```

**Wie ist eine Wiederholschleife aufgebaut?**

Die Syntax einer Wiederholschleife lautet:

```
while( ausdruck )
    anweisungen
end
```

Der *ausdruck* im Schleifenkopf definiert eine Bedingung, die vor jedem Schleifendurchlauf getestet wird. Solange diese Bedingung erfüllt ist, gibt es einen weiteren Schleifendurchlauf, bei dem die Anweisungen im Schleifenkörper ausgeführt werden.

**Mit welcher Funktion kann man 2D-Grafiken erzeugen?**

Die MATLAB-Funktion *plot* erzeugt 2D-Grafiken aus einem Datensatz, der durch eine Folge von Zahlen definiert ist, z.B.

```
>> x = [0,10,20,30,40,50,60];
>> y = [0,2,1,-3,4,2,-1];
>> plot( x, y );
```

Die Plot-Funktion verbindet standardmäßig die einzelnen Punkte (x,y) mit geraden Linien. Durch weitere Optionen kann man das Aussehen der Grafik beeinflussen.