

Berechnungen mit MATLAB

Zu den Aufgaben der drei Bände wurden MATLAB-Programme entwickelt, die von dieser Internet-Seite heruntergeladen werden können. Im Folgenden wird eine kurze Einführung in MATLAB gegeben.

MATLAB-Einführung

1 Reelle Zahlen

In der Dezimaldarstellung einer Zahl wird das Komma durch einen Punkt ersetzt. Das Semikolon am Ende der Zuweisung unterdrückt die Bildschirmausgabe des Zahlenwertes:

```
a = 1.234;
```

Namen von Variablen müssen mit einem Buchstaben beginnen. MATLAB unterscheidet zwischen Groß- und Kleinschreibung; die Variable A hat nicht den Wert, welcher der Variablen a zugewiesen wurde.

Das Malzeichen darf nicht weggelassen werden. MATLAB kennt keine griechischen Buchstaben; der Wert von $\pi = \pi$ ist in MATLAB bekannt:

```
w = 2 * pi * f;
```

Bei Funktionen wird das Argument in Klammern gesetzt. Die trigonometrischen Funktionen sin, cos und tan verlangen Winkelangaben im Bogenmaß (rad); Winkelangaben in Grad erhält man mit der Variablen b2g:

```
b2g = 180 / pi; wg = b2g * wr;
```

Es gibt aber auch die Winkelfunktionen sind, cosd und tand, bei denen der Winkel in Grad steht; der Buchstabe d kommt von degree (Grad)

Beim Arkustangens wird der Hauptwert (in rad) geliefert. Es gibt auch einen Arkustangens für zwei Argumente, bei dem die Umrechnung von Koordinatenangaben intern vorgenommen wird.

Math. Ausdruck	MATLAB-Ausdruck	Ergebnis
$\sqrt{2}$	sqrt(2)	1.4142
$\ln 0,5$	log(0.5)	-0.6931
$\lg 0,0001$	log10(1e-4)	-4.0000
$e^{\ln 0,5}$	exp(log(0.5))	0.5000
$\sin 30^\circ$	sin(30/b2g)	0.5000
$\cos \pi$	cos(pi)	-1.0000
arctan 3	atan(3)	1.2490
Punkt (-1; -3)	atan2(-3,-1)	-1.8925
$\sinh 0,5$	sinh(0.5)	0.5211
artanh 0,8937	atanh(0.8937)	1.4400

2 Ein- und Ausgabe

Die Eingabe einer Variablen über die Tastatur kann einen Hinweis-Text enthalten, der in Hochkommata eingeschlossen wird:

```
f = input('Frequenz f = ');
```

Die Ausgabe kann ebenfalls einen Hinweis-Text enthalten. Bei der Ausgabe einer Zahl wird auch die Ausgabeform mit der zugehörigen Anzahl der Zeichen in Hochkommata eingeschlossen:

```
disp([' w = ', sprintf('%5.4e ',w,' 1/s')]);
```

Für die Ausgabeform gibt es die Möglichkeiten:

- d: ganze Zahl;
- e: Gleitpunktzahl;
- f: Festpunktzahl;
- s: Zeichenkette.

3 Komplexe Zahlen

Eine komplexe Zahl wie z. B. $z = 3 + j4$ kann in drei Formen eingegeben werden: $3+4i$; $3+4*j$; $3+j*4$; die übliche mathematische Schreibweise wird von MATLAB nicht akzeptiert.

Von einer komplexen Zahl können folgende Werte abgerufen werden:

`real(z)` ... Realteil
`imag(z)` ... Imaginärteil
`abs(z)` ... Betrag
`angle(z)` ... Winkel
`conj(z)` ... die konjugiert komplexe Größe z^*

Eine Eingabe der komplexen Größe in der P-Form kennt MATLAB nicht, deshalb muss sie umschrieben werden:

```

zb = input('Betrag = ');
wg = input('Winkel in Grad = ');
wb = wg / b2g; % Winkel im Bogenmaß
z = zb * cos(wb) + j * zb * sin(wb);

```

Ein Kommentar kann nach dem Zeichen % eingefügt werden.

Bei komplexen Rechnungen sollten die Variablen `i` und `j` nicht mit anderen Werten belegt werden, damit keine unerwünschten Effekte oder Fehler auftreten.

4 Matrizen

Eine Matrix wird in eckigen Klammern eingegeben. Am Ende jeder Zeile muss ein Semikolon stehen.

Beispiel

Die Lösung des linearen Gleichungssystems

$$\begin{aligned} 3x + 4y &= 22 \\ 4x + 5y &= 28 \end{aligned}$$

ist mit MATLAB folgendermaßen möglich:

```

M = [3 4; 4 5]; Rs = [22; 28];
L = linsolve(M, Rs);

```

Nach der Lösung ist auch `L` eine Matrix mit den Elementen 1 und 2, auf die folgendermaßen zugegriffen werden kann:

```

x = L(1);
y = L(2);

```

5 Kontrollstrukturen

MATLAB kennt vier Arten von Kontrollstrukturen:

- if-Anweisung
- case-Anweisung
- for-Schleife
- while-Schleife

Wir wollen uns im Rahmen dieser kleinen Einführung lediglich mit der if-Anweisung und der for-Schleife befassen.

Bei der einfachen if-Anweisung folgt dem if eine Bedingung und dann eine Gruppe von Anweisungen; das Ganze wird mit einem `end` abgeschlossen.

Beispiel

Betrag der Zahl `x`

```

x = input('x = ');
if x < 0; x = -x; end;

```

Bei diesem einfachen Beispiel ist also `x < 0` die Bedingung und `x = -x`; die Gruppe der Anweisungen.

Sollen zwei Bedingungen angegeben werden, so können sie durch die Konjunktion (und) mit dem Zeichen `&` oder durch die Alternative (oder) mit dem Zeichen `|` miteinander verknüpft werden. Die Zeichen `<=` (kleiner gleich) bzw. `>=` (größer gleich) werden in MATLAB als `<=` bzw. `>=` nebeneinander geschrieben. Bei der Abfrage auf Gleichheit müssen zwei Gleichheitszeichen gesetzt werden.

Die einfache if-Anweisung kann durch `elseif` und `else` ergänzt werden.

Beispiel

Vorzeichen einer Zahl

```

if x < 0; s = -1,
elseif x == 0; s = 0,
else s = 1, end

```

Bei der for-Anweisung durchläuft eine Variable mehrere Werte von Anfangswert bis zum Endwert in der Form: for k=1:3; ... Anweisungen ... , end

Beispiel

Summe der Zahlen von 1 bis 10

```
s = 0;
for k=1:10;
    s = s + k;
end
```

Die Schrittweite wird nach dem Startwert eingefügt.

Beispiel

Summe der ungeraden Zahlen von 1 bis 19

```
s = 0;
for k=1:2:19;
    s = s + k;
end
```

6 Symbolische Rechnung

Bei der symbolischen Rechnung wird nicht mit Zahlen, sondern mit den Variablenamen gerechnet. Diese Variablen müssen dem Programm unter dem Codewort syms mitgeteilt werden. Die symbolische Rechnung muss in Hochkommata eingeschlossen werden.

Das Ergebnis der symbolischen Rechnung ist ein symbolischer Ausdruck, in den gegebene Zahlenwerte, die dem Programm vorher mitgeteilt wurden, mit der Anweisung subs eingebracht werden können.

Als Beispiel wird im Folgenden die Lösung der quadratischen Gleichung

$$ax^2 + bx + c = 0$$

gezeigt, die im Programm quad.m abgelegt ist.

```
% Lösung einer quadratischen Gleichung
syms a b c x Lv;
clc
disp('Lösung einer quadratischen
Gleichung: a*x^2 + b*x + c = 0');
disp(' ');
a = input('a = ');
if a==0 disp('Wert ist nicht sinnvoll:
keine quadratische Gleichung'), else
    b = input('b = ');
    c = input('c = ');
    disp(' ');
    Lv = solve('a*x^2+b*x+c=0',x);
    x1=subs(Lv(1)); % in der
symbolischen Lösung von solve müssen
die Werte
    x2=subs(Lv(2)); % der Variablen
substituiert werden
    if imag(x1)==0
        disp(['x1 = ',sprintf('%2.4e
',x1)]);
        disp(['x2 = ',sprintf('%2.4e
',x2)]), else
        xr=real(x1);
        xi=abs(imag(x1));
        disp(['konjugiert komplexe
Lösung: ',sprintf('%2.4e ',xr),' +/-
j ',sprintf('%2.4e ',xi)]);
    end
end
```

7 Programmierung

Die MATLAB-Anweisungen können im Command Window direkt eingegeben und gestartet werden. Soll eine Befehlsfolge als Programm gespeichert werden, so muss es in eine m-Datei umgewandelt werden. Die m-Dateien sind ASCII-Dateien, die in einem Verzeichnis gespeichert werden, z.B. als quad.m. Beim Aufruf einer m-Datei im Command Window wird nur der Dateiname ohne die Erweiterung „.m“ eingegeben.