

# Fragen zu „Programmieren mit MATLAB“ – Ulrich Stein

## 2. Programmstrukturen

### 2.1 Funktionen

*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, ...*

*Was steht im Rumpf einer Funktion?*

Im Rumpf einer Funktion stehen die Anweisungen, die die Funktion ausführen soll. Formalparameter, die im Kopf der Funktion deklariert sind, können im Rumpf der Funktion verwendet werden. Speziell sollten im Rumpf die Werte für die Rückgabeparameter gesetzt werden.

*Womit werden Kommentare gekennzeichnet?*

Kommentare werden in MATLAB mit dem %-Zeichen eingeleitet und gelten bis zum Ende der jeweiligen Zeile.

*Wie lautet der Dateiname einer Funktion?*

Der Dateiname der Funktion ist identisch mit dem Namen der Funktion und hat die Endung „.m“. Besitzt die Datei noch weitere, lokale Funktionen, dann kann von außen nur die Hauptfunktion aufgerufen werden.

*Wie ruft man eine Funktion im Command Window auf?*

Im Command Window und auch in jeder weiteren Funktion wird eine Funktion durch die Angabe ihres Namens aufgerufen. Beim Aufruf müssen die notwendigen Übergabeparameter gesetzt werden.

*Wie werden Daten an eine Funktion übergeben?*

Daten werden einer Funktion bei Aufruf als Werte (Aktualparameter) übergeben und nicht als Variablen im Rechner-Speicher (Pass-by-value). Zur Übergabe wird der Stack verwendet.

*Was ist das Arbeitsverzeichnis?*

Sie arbeiten mit MATLAB immer in einem bestimmten Verzeichnis, dem Arbeitsverzeichnis. Standardmäßig können Sie eigene Funktionen nur dann aufrufen, wenn ihre M-Files in diesem Verzeichnis liegen.

### 2.2 Ein- und Ausgabe

*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.

*Mit welcher Funktionen kann man Daten von der Tastatur einlesen?*

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

*Wie erzwingt man, dass die eingegebenen Daten als Text interpretiert werden?*

Die Funktion `input` erlaubt den optionalen Parameter `'s'`. Ist dieser gesetzt, interpretiert MATLAB jede Eingabe als Text.

*Welche Funktionen wandeln Text in Zahlen bzw. Zahlen in Text um?*

Die Funktion `str2double` wandelt Text in Zahlen, die Funktion `num2str` wandelt Zahlen in Text um.

*Womit überprüft man, ob eine Variable einen Text bzw. eine Zahl enthält?*

Die Funktion `ischar` testet, ob eine Variable ein Text ist, die Funktion `isnumeric` testet, ob eine Variable eine Zahl ist.

## 2.4 Verzweigungen

*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 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.

*Wie formuliert man Bedingungen?*

Bedingungen werden über mathematische Vergleiche formuliert.

*Welche Vergleichsoperatoren kennen Sie?*

MATLAB besitzt folgende Vergleichsoperatoren:

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

*Wie genau rechnet MATLAB?*

Die Rechengenauigkeit für reelle Zahlen liegt in MATLAB bei  $2.2 \cdot 10^{-16}$ . Über die Variable `eps` können Sie diesen Wert abfragen.

*Welche logischen Operatoren kennen Sie?*

MATLAB kennt die drei logischen Operatoren: `&&` (UND), `||` (ODER) und `~` (NICHT).

*Wie fragt man Bereiche ab?*

Zahlenbereiche formuliert man mit den logischen Operatoren `&&` und `||`.

*Wie formuliert man in MATLAB eine einfache Alternative?*

Die Alternative erfolgt über eine *if-else*-Abfrage mit der Syntax:

```
if( logischer_ausdruck )
    Anweisungen 1
else
    Anweisungen 2
end
```

*Was macht man, wenn es mehr als zwei Möglichkeiten gibt?*

Bei mehr als zwei Möglichkeiten verwendet man verschachtelte Schleifen oder eine *switch*-Anweisung.

*Was bedeuten die Fortsetzungszeichen "..." in MATLAB?*

Passt eine Anweisung nicht in eine Zeile des Editors, kann man die Anweisung auf mehrere Zeilen verteilen. Am Ende jeder Zeile müssen dann die Fortsetzungszeichen „...“ stehen. Texte kann man jedoch durch Fortsetzungszeichen nicht trennen.

*Wie formuliert man eine Fallunterscheidung?*

Für eine Fallunterscheidung verwendet man die *switch*-Anweisung

```
switch( ausdruck )
    case
        Anweisungen
    case . . .
    otherwise
        Anweisungen
end
```

## 2.5 Schleifen

*Wie lautet die Syntax einer Zählschleife?*

Die Syntax der Zählschleife lautet:

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

*Welche Funktion hat die Laufvariable (index) bei einer Zählschleife?*

Eine Zählschleife durchläuft eine Anzahl von Zahlen. Welche Zahl beim aktuellen Durchlauf gerade dran ist, vermerkt die Laufvariable.

*Wie berechnet man Summen bzw. Produkte?*

Eine Summe bzw. ein Produkt berechnet man normalerweise mit einer Schleife. Man verwendet hierzu einen Zwischenspeicher (Summentopf), der bei jedem Schleifendurchlauf mit weiteren Werten befüllt wird, z.B. für einen Zwischenspeicher mit dem Namen *sum*, zu dem ein Wert addiert wird:

```
sum = sum + wert;
```

*Wie unterscheiden sich Iteration und Rekursion?*

Iteration und Rekursion dienen dazu, Operationen mehrfach nacheinander auszuführen. Eine Iteration verwendet hierfür eine Schleife, während eine Rekursion sich selbst aufruft.

*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.

*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.

*Was ist eine Taylorreihe?*

Man kann reguläre Funktionen in der Umgebung eines Punktes *x* über eine unendliche Potenzreihe definieren, ihre Taylorreihe. Einige mathematische Funktionen sind auch nur über eine Taylorreihe definiert.

*Wozu benötigt man eine Abbruchbedingung?*

Wenn Sie (formal) den Wert einer unendlichen Reihe berechnen wollen, müssten Sie theoretisch eine Schleife unendlichmal durchlaufen. In der Praxis definiert man deshalb eine Abbruchbedingung (z.B. über eine Schranke), die festlegt, wann das Ergebnis genau genug ist.

*Durch welche Funktionen kann eine Schleife vorzeitig verlassen?*

Mit der Funktion *break* können Sie eine Schleife vorzeitig verlassen. Wenn Sie außer der Schleife auch die gesamte Funktion verlassen wollen, können Sie auch die Funktion *return* verwenden.

## 2.6 Felder

*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  $A(z,s)$ .

*Wie baut man mit Hilfe einer Schleife einen Vektor auf?*

Man belegt die Komponenten des Vektors in der Schleife nacheinander mit Werten, z.B. die  $n$ -te Komponente des Vektors  $v$  mit einem Wert  $wert$ :

```
v(n) = wert;
```

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

Die MATLAB-Funktion `plot` erzeugt 2D-Grafiken aus einem Datensatz, siehe Abschnitt 2.7.

*Welche Matrizen kann man mit speziellen MATLAB-Befehlen erzeugen?*

MATLAB kennt u.a. folgende Funktionen zur Belegung von Matrizen: `zeros`, `ones`, `eye`, `diag`, `rand` und `randn`.

*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
```

*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.

*Wozu dient die Funktion meshgrid?*

Die Funktion `meshgrid` erzeugt ein Gitternetz für 3D-Grafiken.

*Welche Matrix-Operationen kennen Sie?*

MATLAB kennt die Rechenoperationen  $+$ ,  $-$ ,  $*$  und  $^$  auch für Matrizen. Zusätzlich gibt es noch den Punktoperator für  $*$ ,  $/$  und  $^$ . Der Operator `'` transponiert eine Matrix. Ansonsten gibt es u.a. noch die Operationen der Linearen Algebra `norm`, `inv`, `det`, ...

*Wozu dient die Links-Division in MATLAB?*

Die Links-Division dient u.a. zum Lösen von linearen Gleichungssystemen.

*Wie kann man Matrizen verknüpfen?*

Mit Hilfe der Verknüpfungs-Operation `[ ]` können Sie in MATLAB kleinere Matrizen zu größeren zusammenfügen.

*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  $A\{z,s\}$ .

## 2.7 Grafik

*Wie wird der Befehl `fplot` angesteuert?*

Die Funktion `fplot` benötigt beim Aufruf den Function-Handle der zu plottenden Funktion und den Definitionsbereich, über den geplottet werden soll.

*Was sind Function-Handles, was sind anonyme Funktionen?*

Function-Handles verweisen auf eine Funktion in einem m-File. Ein Function-Handle erhält man, indem man das @-Zeichen vor den Funktionsnamen setzt. Sie können einfache Funktionen auch „anonym“, d.h. ohne einen m-File, definieren, indem Sie eine einzeilige Funktionsdefinition direkt hinter dem @-Zeichen angeben.

*Wie plottet man einen Datensatz?*

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.

*Wie ändert man für einen Plot die Farbe, den Linientyp oder das Symbol?*

In einem `plot`-Aufruf können Sie hinter den Daten auch noch Farbe, den Linientyp oder ein Symbol definieren, indem Sie diese Werte als Text angeben.

*Wie verhindert man, dass ein weiterer `plot`-Aufruf die vorherige Grafik löscht?*

Der Aufruf „hold on“ verhindert, dass ein weiterer `plot`-Aufruf die bisherige Grafik löscht.

*Mit welchen Befehlen kann man eine Zeichnung beschriften?*

Sie können an eine Grafik weitere Beschriftungen anbringen, z.B. mit `title` eine Überschrift, mit `legend` eine Legende, mit `xlabel` eine Beschriftung der x-Achse, usw.

*Welche Befehle zum Erzeugen von 3D-Grafiken kennen Sie?*

Für eine schattierte Flächendarstellung gibt es den Befehl `surf`, für eine Drahtmodell Darstellung den Befehl `mesh`. Weitere Darstellungen erhält man z.B. mittels `contour3` und `contourf`.

*Wie kann man die Blickrichtung festlegen?*

Mit dem Befehl `view` kann man die Blickrichtung festlegen.

*Wie kann man mehrere Plots auf einem einzigen Blatt unterbringen?*

Mit dem Befehl *subplot* kann man ein Zeichnungsblatt in mehrere *plot*-Bereiche aufteilen.

*Wie kann man den Plot-Ausschnitt festlegen?*

Der Befehl *axis* legt den Plot-Ausschnitt und weitere Eigenschaften der Grafik fest.

*Wie kann man 3D-Kurven erzeugen?*

3D-Kurven erzeugt man mit dem Befehl *plot3*.

*Wie erzeugt man Text in einer Grafik?*

Der Befehl *text* erstellt Text an einer beliebigen Stelle der Grafik.

*Was ist ein Grafik-Handle?*

Grafik-Handles sind Referenzen auf ein Grafik-Objekt. Sie erlauben den Zugriff auf das Grafik-Objekt, um z.B. dessen Eigenschaften zu verändern.

## 2.8 Strukturen und Klassen

*Wie definiert man in MATLAB einen struct?*

Sie können einen *struct* durch die Angabe des Datentyps *struct* definieren. Noch einfacher ist es, wenn Sie zu einer Variablen über den Punktoperator eine weitere Komponente definieren.

*Wie greift man auf die Komponenten eines structs zu?*

Auf Komponenten eines *structs* greift man über den Punktoperator zu.

*Was beschreibt die Sektion properties einer Klasse?*

In der Sektion *properties* werden die Eigenschaften (Variablen) einer Klasse deklariert.

*Was beschreibt die Sektion methods einer Klasse?*

In der Sektion *methods* wird die Funktionalität (Methoden/Funktionen) einer Klasse definiert.

*Was ist ein Konstruktor?*

Der Konstruktor ist die wichtigste Methode der Klasse und dient dazu, Objekte der Klasse zu erzeugen.

*Wie sieht das Gerüst einer Klasse aus (Klassendefinition)?*

```
classdef Classname
    properties
        var1 ...
    end
    methods
        function obj = Classname( ... )
            ...
        end
        ...
    end
end
```

*Wie lautet der Name der Datei, die eine Klassendefinition enthält?*

Der Name des m-Files mit der Klassendefinition hat denselben Namen wie die Klasse, mit der Endung „.m“.

*Wie erzeugt man ein Objekt einer Klasse?*

Ein Objekt einer Klasse wird durch den Aufruf des Konstruktors erzeugt.

*Wie greift man auf die Eigenschaften eines Objekts zu?*

Auf die Eigenschaften eines Objekts greift man über den Punktoperator zu.

*Warum haben Methoden, die die Eigenschaften ihres Objekts verändern, als Rückgabewert die Objektvariable?*

Den Methodenaufrufen wird eine Kopie des Objekts übergeben (Pass-by-value). Verändert die Methode die Eigenschaften des Objekts, dann erfolgt die Änderung nur in der Kopie. Deshalb muss das veränderte Objekt zurückgegeben und in zum Objekt gehörenden Variablen gespeichert werden.

*Was bedeutet Datenkapselung?*

Mit Attributen kann der Zugriff auf Eigenschaften und Methoden eines Objekts eingeschränkt werden. Die Daten können so vor unbefugten Veränderungen geschützt werden, sie sind gekapselt.

*Was sind abgeleitete Klassen?*

Abgeleitete Klassen sind „Kinder“ einer Basisklasse. Damit lassen sich Klassengruppen realisieren, für die man gewisse Basis-Funktionalitäten an die gemeinsame Vaterklasse auslagert.

*Wie erzeugt man ein Objekt-Array?*

Man erzeugt in MATLAB ein Objekt-Array ähnlich wie ein Zahlen-Array, indem man mit Hilfe der Konstruktoren indizierte Variable erzeugt, z.B. `p(1) = ClassName();`, `p(2) = Classname();`, etc.

*Wie durchsucht man ein Objekt-Array?*

Man durchsucht ein Objekt-Array am besten mit einer Schleife, wobei man bei jedem Schleifendurchlauf gewisse Eigenschaften des aktuellen Objekts testet, z.B. `if( p(n).var == v ) ...`

## 2.9 Dateien

*Was ist ein File-Identifizierer?*

Beim Öffnen einer Datei wird ein File-Identifizierer vergeben, eine Zahl größer oder gleich 3. Erhält man beim Öffnen den Wert -1, dann ist der Aufruf fehlgeschlagen.

*Was bedeutet die Zugriffart beim Öffnen einer Datei?*

Beim Öffnen einer Datei muss man neben dem Dateinamen noch die Zugriffsart festlegen, z.B. ‚r‘ für einen Lesezugriff oder ‚w‘ für einen Schreibzugriff.

*Mit welchen Funktionen greift man auf den Inhalt einer Text-Datei zu?*

Am einfachsten ist es, wenn man mit der Funktion `fgetl` die Daten der Datei zeilenweise einliest. Jeder `fgetl`-Aufruf schiebt den File-Pointer für den Zugriff um eine Zeile weiter.



*Wie merkt man, dass das Ende einer Text-Datei erreicht ist?*

Die Funktion *fgetl* gibt die Zahl -1 zurück, wenn das Ende der Datei erreicht ist.

*Welche Funktion schreibt Text in eine Text-Datei?*

Sie können mit der Funktion *fprintf* Texte in eine Datei schreiben.

*Welche Funktionen gibt es zum Zugriff auf eine Excel-Datei?*

Die Funktion *writematrix* schreibt Daten in eine Excel-Datei, *readmatrix* liest Daten aus einer Excel-Datei nach MATLAB ein.

*Welche Funktionen gibt es zum Zugriff auf MAT-Files?*

Mit der Funktion *save* schreiben Sie Daten in einen MAT-File, mit *load* lesen Sie Daten ein.

## 2.10 Strings

*Was versteht man unter Character-Arrays?*

Texte werden in MATLAB als Felder vom Datentyp *char* geführt.

*Was ist ein string-Objekt?*

Texte können auch als *string*-Objekte in MATLAB geführt werden. Zur Erzeugung verwendet man dann doppelte Anführungszeichen.

*Welche String-Funktionen kennen Sie?*

Die wichtigsten String-Funktionen sind: *ischar*, *length*, *strcmp*, *strncmp*, *sscanf*, *str2num*.

*Welche Datums-Funktionen kennen Sie?*

MATLAB hat die Funktion *datetime* zum Auslesen des am Rechner eingestellten Datums.

*Was versteht man unter einer String-Evaluation?*

Die Funktion *eval* dient dazu, einen übergebenen String als MATLAB-Kommando auszuführen.