

Kapitel 2

Datentypen und Variablen

Datentypen - Einführung

- In Matlab gibt es verschiedene Datentypen, um Werte zu speichern.
- Die grundlegenden Datentypen sind:
 - double, single (Floating-point Numbers)
 - int8, uint8, etc (Integers)
 - char, string (Characters and Strings)
 - logical (logical operations)

Datentypen - Einführung

- Des Weiteren gibt es noch:
 - function_handle (Function Handles)
 - table (Tables)
 - struct (Structures)
 - cell (Cell Arrays)
- Dies sind aber keine Datentypen im klassischen Sinne. Matlab spricht daher auch von "fundamental Matlab classes".
- Verwenden wir Zahlen und geben keinen Typ vor, so verwendet Matlab standardmäßig immer "double" als Datentyp.

Datentypen - Einführung

- In dieser Tabelle sind die wichtigsten arithmetischen Datentypen aufgelistet:

Datentyp	Länge	Wertebereich	
		von	bis
int8	1 Byte	-128	+127
int64	8 Byte	-9,223,372,036,854,775,808	+9,223,372,036,854,775,807
uint8	1 Byte	0	+255
single	4 Byte	$\pm 1.1755 * 10^{-38}$	$\pm 3.4028 * 10^{+38}$
double	8 Byte	$\pm 2.2251 * 10^{-308}$	$\pm 1.7977 * 10^{+308}$

Datentypen - Einführung

- Durch den Befehl *whos* kann man sich Infos zu dem verwendeten Datentyp der Variable zeigen lassen:

```
a = 3.23
```

```
b = "Hallo"
```

```
whos a
```

Name	Size	Bytes	Class	Attributes
a	1x1	8	double	

Unterschiedliche Zahlensysteme

dezimal	hexadezimal	oktal	binär (dual)
0	0	0	0
1	1	1	1
2	2	2	10
3	3	3	11
4	4	4	100
5	5	5	101
6	6	6	110
7	7	7	111
8	8	10	1000
9	9	11	1001
10	A	12	1010
11	B	13	1011
12	C	14	1100
13	D	15	1101
14	E	16	1110
15	F	17	1111
16	10	20	10000
USW.	USW.	USW.	USW.

Ganze Zahlen

- Ganze Zahlen (ganzzahlige Konstanten) werden unter den Datentyp *int* (*int8*, *int16*, *int32*, *int64*, *uint8*, *uint16*, *uint32*, *uint64*) gespeichert.
- Der Computer speichert die Zahlen aber nicht in der Form, wie wir sie benutzen. (Also z.B. die Zahl 6 wird nicht als "6" gespeichert.)
- Die Speicherung erfolgt als Binärzahl. (z.B. 0110 für die Zahl 6)
- Um auch negative Zahlen speichern zu können, wird die Zahl bei den meisten Rechenwerken in der Zweierkomplementdarstellung gespeichert.

Ganze Zahlen

Beispiel: 4 wird im Computer so gespeichert:

1. Vorzeichen ignorieren und ins Binärsystem umrechnen

4 => 0000 0100

Ganze Zahlen

Beispiel: -4 wird im Computer so gespeichert:

1. Vorzeichen ignorieren und ins Binärsystem umrechnen

$$4 \Rightarrow 0000\ 0100$$

2. Invertieren, da negativ

$$1111\ 1011$$

3. Eine 1 addieren, da negativ

$$1111\ 1011 + 0000\ 0001 \Rightarrow 11111100$$

Ganze Zahlen

- Durch die Speicherung der Zahlen in Zweierkomplementdarstellung können wir auf dem vorgesehenen Platz zwar nur halb so viele Zahlen speichern.
- Jedoch ermöglicht uns dies, dass die Verarbeitung von positiven und negativen Zahlen im Rechenwerk gleich bleiben kann. Ansonsten müsste der Computer bei Berechnungen positive und negative Zahlen intern unterschiedlich behandeln.

Ganze Zahlen

- Ganze Zahlen liegen meist im Dezimalzahlensystem vor.
- Sie können aber auch in einem anderen Zahlensystem eingegeben, verarbeitet und ausgegeben werden.
- Gängige Zahlensysteme sind das
 - Binärsystem (Dualsystem),
 - Dezimalsystem,
 - Oktalsystem
 - und das Hexadezimalsystem.

Ganze Zahlen

- Ganze Zahlen können in die verschiedenen Zahlensysteme dezimal, oktal oder hexadezimal umgerechnet werden.
- Es gibt dafür in Matlab diverse Umwandlungs-Befehle:

`dec2hex, dec2base, dec2bin, hex2dec, etc.`

Umwandlung - Zahlensysteme

- Umwandlung einer Binärzahl in eine Dezimalzahl:
 $(010111)_2 = 0 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = (23)_{10}$
=> Matlab: `bin2dec("010111")`
- Umwandlung einer Hexadezimalzahl in eine Dezimalzahl:
 $(C2)_{16} = (12 \cdot 16^1 + 2 \cdot 16^0) = (194)_{10}$
=> Matlab: `hex2dec("c2")`
- Umwandlung einer Zahl zur Basis 3 in eine Dezimalzahl:
 $(12)_3 = (1 \cdot 3^1 + 2 \cdot 3^0) = (5)_{10}$
=> Matlab: `base2dec("12", 3)`

Beispiele

```
# Umwandlung von 15 in eine Hexadezimalzahl  
dec2hex(15)
```

```
# Umwandlung von 15 in eine Binärzahl  
dec2bin(15)
```

```
# Umwandlung von 15 in eine Zahl zur angegebenen Basis  
dec2base(15,3)
```

```
'F'
```

```
'1111'
```

```
'120'
```

Gleitkommazahlen

- Der Standarddatentyp für Gleitkommazahlen (Gleitkomma-Konstanten) ist `double`.
- Die Abspeicherung erfolgt im IEEE-Format (Institute of Electrical and Electronics Engineers) durch Verwendung: eines Vorzeichenbits, 11 Exponentenbits und 52 Mantissenbits.
- Die Genauigkeit der Darstellung ist auf ca. 15 Stellen begrenzt.

Beispiel:

Die Zahl 487 lässt sich als $4,87 \cdot 10^2$ schreiben.

(Mit 4,87 als Mantisse, 10 als Basis und 2 als Exponent.)

Gleitkommazahlen

- Beispiele für die Ein- und Ausgabe von Gleitkommazahlen (Gleitkomma-Konstanten)

1.2345
-0.012
41.0
1.1E+02
-25.0E+15

Das zweite Matlab-Programm

Zeile	Befehl
1	%% Programm zur Berechnung des Kugelvolumens.
2	
3	clc % löscht das Command Window
4	clear % löscht den Workspace
5	
6	% Programmhinweis
7	disp("Dieses Programm berechnet das Kugelvolumen durch einen gegebenen Radius.");
8	
9	% Variablendeklaration (Benötigte Variablen initialisieren)
10	PI = 3.14159;

Das zweite Matlab-Programm

Zeile	Befehl
11	<code>% Benutzerinteraktion</code>
12	<code>radius = input("Bitte geben Sie einen Radius ein: ");</code>
13	
14	<code>volumen = 4 / 3 * PI * radius^3;</code>
15	<code>fprintf("Volumen = %f\n", volumen);</code>

Das zweite Matlab-Programm

- Dieses Programm berechnet das Volumen einer Kugel bei gegebenem Radius.
- In **Zeile 1** wird durch %% eine hervorgehobene Beschreibung des Programms angegeben.
- **Zeile 3** löscht das Command Window
- **Zeile 4** löscht den Workspace
- In **Zeile 7** wird der vorgegebene Text ausgegeben.
- In **Zeile 10** wird eine Variable namens *PI* deklariert und mit dem Wert 3.14159 initialisiert.

Das zweite Matlab-Programm

- In **Zeile 12** wird zu einer Eingabe aufgefordert, bei der der angegebene Text angezeigt wird. Die Eingabe des Benutzers wird sodann in der Variablen *radius* gespeichert.
- In **Zeile 14** findet man nun die Berechnung für *volumen*.
- Und in **Zeile 15** wird das Ergebnis der Volumenberechnung ausgegeben.