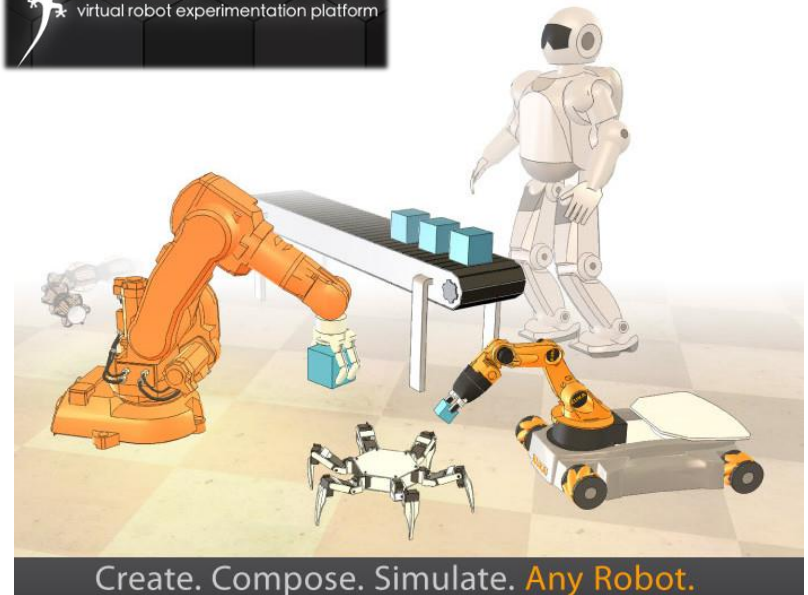
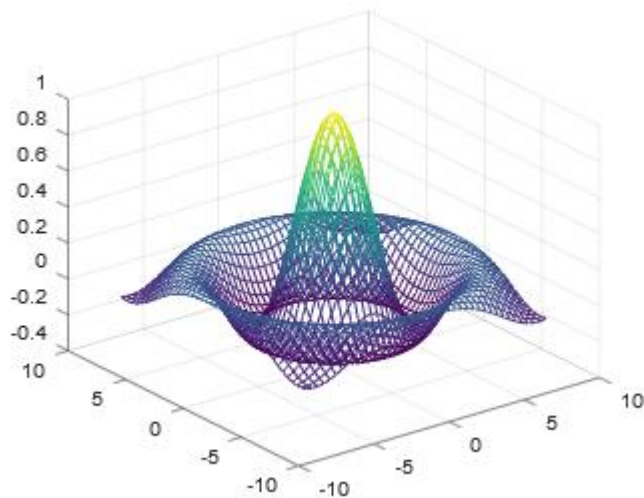


Einführung in Octave und V-REP

Installationsanleitung



TECHNISCHE
UNIVERSITÄT
DARMSTADT



Octave Installation

- Webseite: <https://www.gnu.org/software/octave/>
- Installation: <https://www.gnu.org/software/octave/#install>
- Für Windows, Linux und Mac verfügbar (mind. Version 5.1.0 verwenden)
 - Windows Installation: <https://ftp.gnu.org/gnu/octave/windows/>
 - Warnung bei Installation unter Windows 10 ignorieren
 - Linux Installation: Über Paketmanager aktuelle Version noch nicht verfügbar, daher alternative Installation über Flatpak
 - Flatpak installieren

```
$ sudo apt install flatpack
```

```
$ sudo apt install gnome-software-plugin-flatpak
```
 - Flathub repository hinzufügen

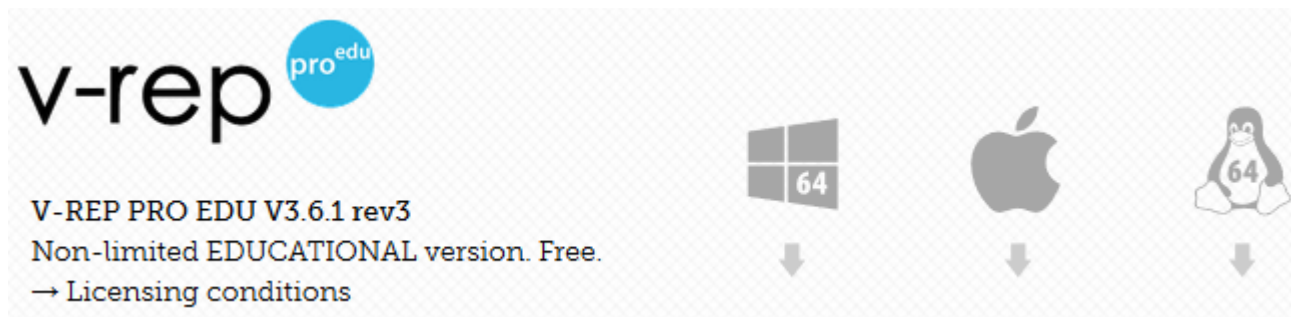
```
$ flatpak remote-add --if-not-exists flathub https://flathub.org/repo/flathub.flatpakrepo
```

 - System neu starten
 - Octave installieren:

```
$ flatpak install flathub org.octave.Octave
```
 - Mac Installation (nicht getestet): [http://wiki.octave.org/Octave for MacOS X](http://wiki.octave.org/Octave_for_MacOS_X)
 - **Unbedingt die 64bit Version verwenden!**

V-REP Installation

- Webseite: <http://www.coppeliarobotics.com>
- Installation: <http://www.coppeliarobotics.com/downloads.html>
- Installationsdateien verfügbar für Windows, Linux und Mac
- „Educational Version“ frei verfügbar
 - Empfohlen: **V-REP PRO EDU**



- Windows: Installationsdatei herunterladen und ausführen
- Linux: Archiv-Datei herunterladen und an gewünschten Ort entpacken
- Minimal (ohne Gewähr): **V-REP PLAYER**

Einrichtung von V-REP in Octave

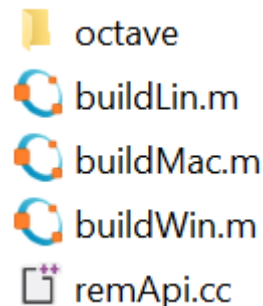
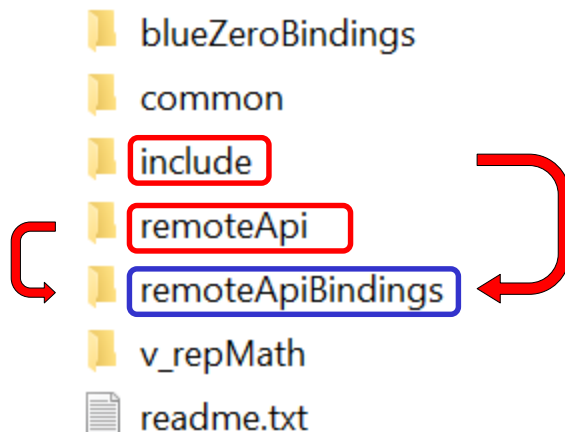
- Offizielle Anleitungen unter:
 - <http://www.coppeliarobotics.com/helpFiles/en/remoteApiClientSide.htm>
 - Abschnitt „Octave client“
- Anleitung hier wird exemplarisch für Windows durchgeführt, verläuft aber auf anderen Plattformen analog
 - Plattformspezifische Abweichungen möglich (z.B. benötigen manche Schritte unter Linux sudo-Rechte)

Einrichtung von V-REP in Octave

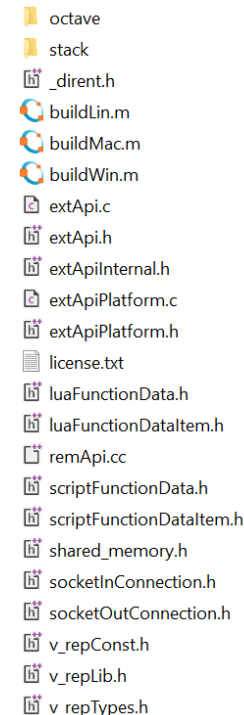
Erzeugung der Bindings

- Ins programming Verzeichnis von V-REP wechseln
 - z.B.: C:\Program Files\V-REP3\V-REP_PRO_EDU\programming
- Dort die **Inhalte** der Ordner `include` sowie `remoteApi` in das Verzeichnis `remoteApiBindings/octave` kopieren

Inhalt octave Ordner:



Vorher



Nachher

Einrichtung von V-REP in Octave

Erzeugung der Bindings

- Octave starten
- In das V-REP `programming` Verzeichnis wechseln
- Dort in den Unterordner `remoteApiBindings/octave` gehen
- Im Kommandofenster von Octave direkt möglich, z.B.:

```
cd "C:/Program Files/V-REP3/V-REP_PRO_EDU/programming/remoteApiBindings/octave"
```
- Danach einfach das Build Script in Octave starten
 - Windows: `buildWin`
 - Linux: `buildLin` (ggf. werden zusätzliche Pakete benötigt)
 - Mac: `buildMac` (nicht getestet!)
- Inhalt von `buildWin` bei installiertem Visual Studio Community 2017 (inkl. der Komponente „Desktopentwicklung mit C++“)

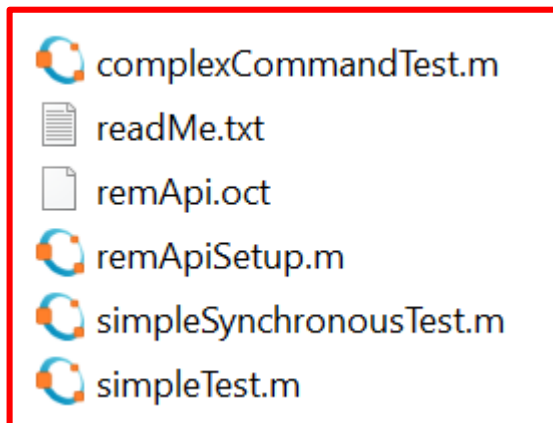
```
system('"C:\\Program Files (x86)\\Microsoft Visual Studio\\2017\\Community\\VC\\Auxiliary\\Build\\vcvarsall.bat" x64')
mkoctfile -DMAX_EXT_API_CONNECTIONS=255 -DNON_MATLAB_PARSING -DDO_NOT_USE_SHARED_MEMORY -lwinmm -lWs2_32 remApi.cc
extApi.c extApiPlatform.c
```

- Warnungen sind normal und können ignoriert werden

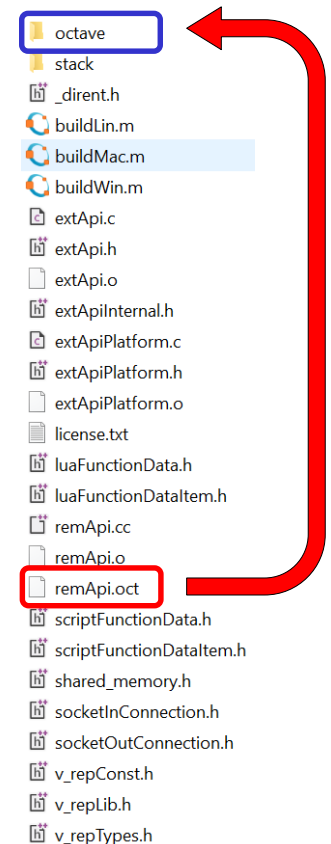
Einrichtung von V-REP in Octave

Erzeugung der Bindings

- Die erzeuge `remApi.oct` ins `octave` Verzeichnis kopieren
- Der **Inhalt** des `octave` Verzeichnis sind nun unsere **Bindings!**
- Diese Bindings werden für **Windows 10 64 Bit** auf dem Moodle Portal zur Verfügung gestellt
 - Sollte auch für Windows 7 funktionieren (keine Garantie)
 - Bindings für andere Systeme können mit der Zeit ergänzt werden



Bindings



Einrichtung von V-REP in Octave

Einbindung der Bindings

- Das Binding sollte in einen Ordner `v-rep` (muss manuell erstellt werden) in das Octave-Paketverzeichnis entpackt/kopiert werden

z.B.: `C:\Octave\Octave-5.1.0.0\mingw64\share\octave\5.1.0\m`

- In der `remApiSetup.m` aus dem Binding sollte die Zeile `octfile` direkt („hardcoded“) auf eurer erzeugte Bibliothek zeigen, z.B:

```
octfile = "C:/Octave/Octave-5.1.0.0/mingw64/share/octave/5.1.0/m/v-rep/remApi.oct";
```

- Andere Verzeichnisse sind möglich, aber dann muss bei jedem Octave Start dieser Ordner der Umgebung hinzugefügt werden

```
addpath("<my_custom_path>/v-rep")
```

- Achtung! Hier muss Slash (/) nicht Backslash (\) im Pfad verwendet werden!

- Alternativ zum Aufruf von `addpath`, kann der Pfad permanent hinzugefügt werden, indem der Befehl in die `octaverc` hinzugefügt wird

z.B.: `C:\Octave\Octave-5.1.0.0\mingw64\share\octave\5.1.0\m\startup`

- **In den Übungen wird vorausgesetzt, dass diese Bindings in der Umgebung eingerichtet sind!**

Einrichtung von V-REP in Octave

Einbindung der Bindings

- Binding Verzeichnisse unter Ubuntu 18.04
 - Neuen Ordner in Octave-Verzeichnis anlegen

```
$ sudo mkdir  
/var/lib/flatpak/app/org.octave.Octave/current/active/files/share/octave/5.1.0/m/v-rep
```

- Bindings kopieren

```
$ sudo cp /<my_custom_path>/V-  
REP_PRO_EDU_V3_6_1_Ubuntu18_04/programming/remoteApiBindings/octave/octave/*  
/var/lib/flatpak/app/org.octave.Octave/current/active/files/share/octave/5.1.0/m/v-rep/
```

- In `remApiSetup.m` Pfad angeben:

```
octfile = "/app/share/octave/5.1.0/m/v-rep/remApi.oct";
```

Einrichtung von V-REP in Octave

Testen der Bindings

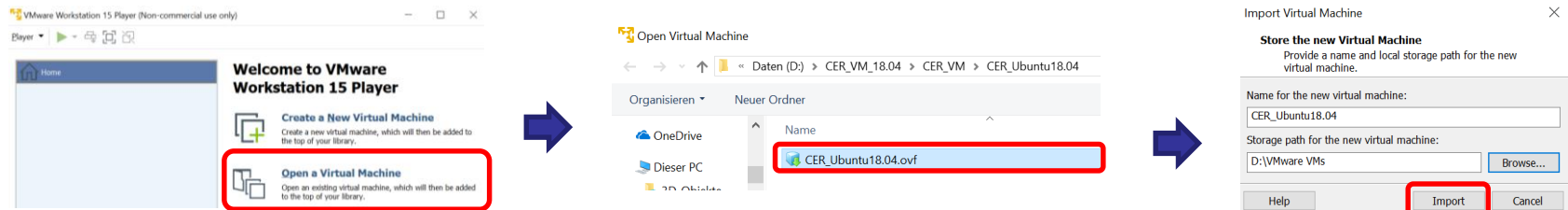
- Octave sowie V-REP starten
 - Bei Octave müssen die Bindings in der Umgebung hinzugefügt sein (vorherige Folien)!
- V-Rep erstellt beim Starten einen Server zur Kommunikation mit den Clients. Der Port 19997 hierfür ist in der Datei `remoteApiConnections.txt` (z. B. in : „C:\Program Files\V-REP3\V-REP_PRO_EDU“) definiert
- Im Verzeichnis der Bindings ist eine `simpleTest.m` beigelegt
 - Die ggf. editiert, so dass der Port 19997 (anstelle 19999) verwendet wird

```
12 function simpleTest()  
13     disp('Program started');  
14     vrep=remApiSetup();  
15     simxFinish(-1); % just in case, close all opened connections  
16     clientID=simxStart('127.0.0.1',19997,true,true,5000,5);  
17     if (clientID>-1)  
18         disp('Connected to remote API server');  
19     end
```

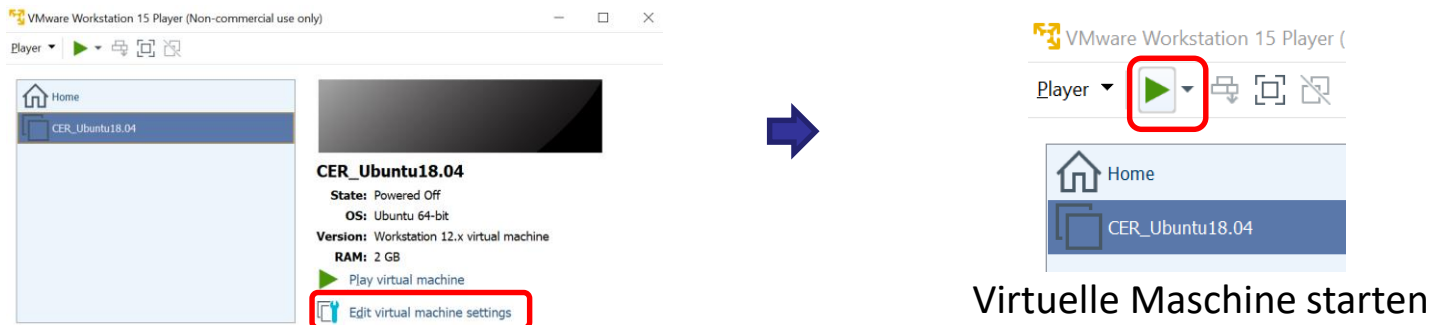
- Nun `simpleTest.m` in Octave ausführen:
 - Octave: Auflistung von Maus-Positionen
 - V-REP: In der Konsolenausgabe sollte „Hallo V-REP!“ erscheinen

Virtuelle Maschine

- Wir empfehlen die Installation von Octave und V-Rep wie gezeigt (Win10, Ubuntu 18.04)!
- Als Alternativlösung bei nicht auflösbaren Installationsproblemen stellen wir eine virtuelle Maschine mit **Ubuntu 18.04** und bereits installiertem Octave und V-Rep im *Open Virtualization Format* zur Verfügung (PW des Benutzers ist Moodle Einschreibepasswort).
 - Benötigt circa 18.4GB Plattenplatz. Virtualisierungsoption des CPUs im Bios muss aktiviert sein
- Einrichten z. B. über VMware Workstation Player 15



- Empfohlene Einstellungen der VM: zwei CPUs, 2 GB Arbeitsspeicher, 768 MB Grafikspeicher



Virtuelle Maschine

- VMware tools sind bereits installiert

```
$ sudo apt install open-vm-tools
```

- Bei Verwendung nur eines CPUs kommt es vor, dass die Display-Auflösung nicht mehr automatisch an die Fenstergröße angepasst wird.

- Lösungsansatz: anschließend Auflösung neu festlegen.
- Alternative Lösungsansätze gerne im Forum kommunizieren

- Octave starten:

- Windows-Taste drücken
- Im erscheinenden Suchfenster "Octave" eintippen
- Das Suchergebnis "GNU Octave" auswählen

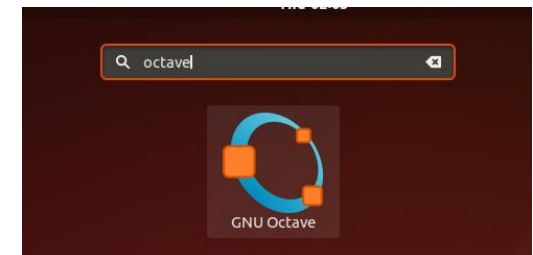
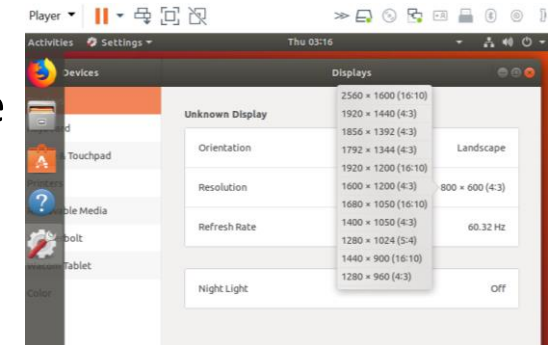
- V-Rep starten:

- Konsole öffnen (Tastenkombination Strg+Alt+T)
- In das Verzeichnis von V-Rep wechseln

```
$ cd /home/cer/Applications/V-REP_PRO_EDU_V3_6_1_Ubuntu18_04/
```

- V-Rep ausführen:

```
$ ./vrep.sh
```

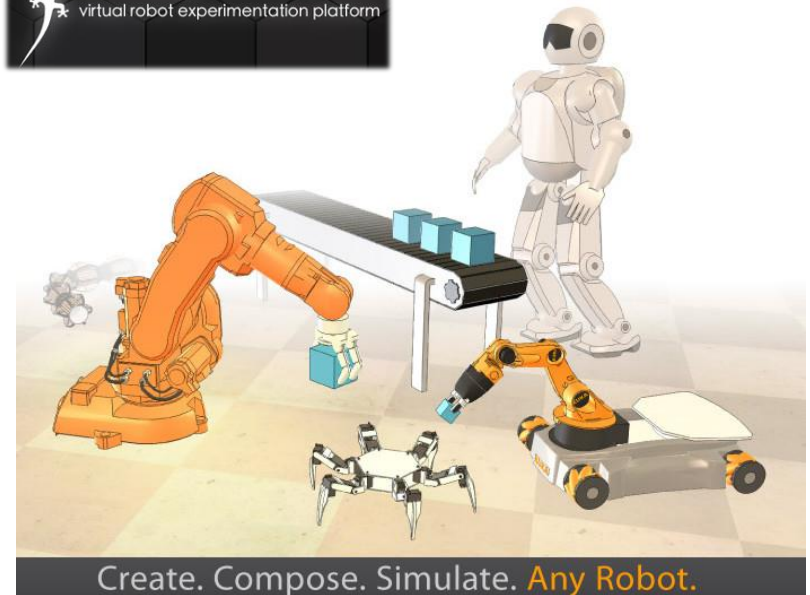
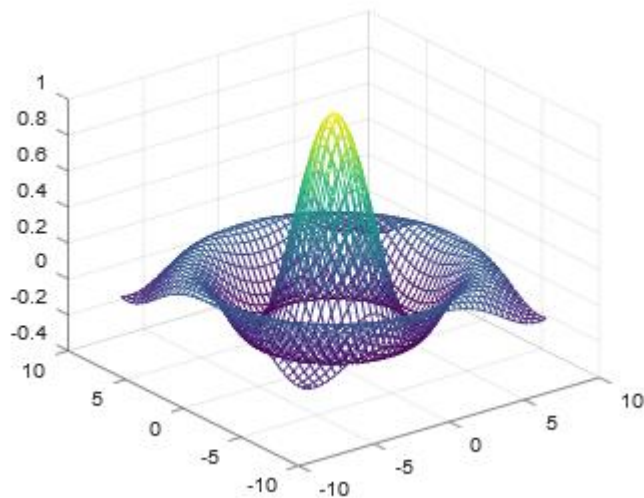


Einführung in Octave und V-REP

Tutorial



TECHNISCHE
UNIVERSITÄT
DARMSTADT

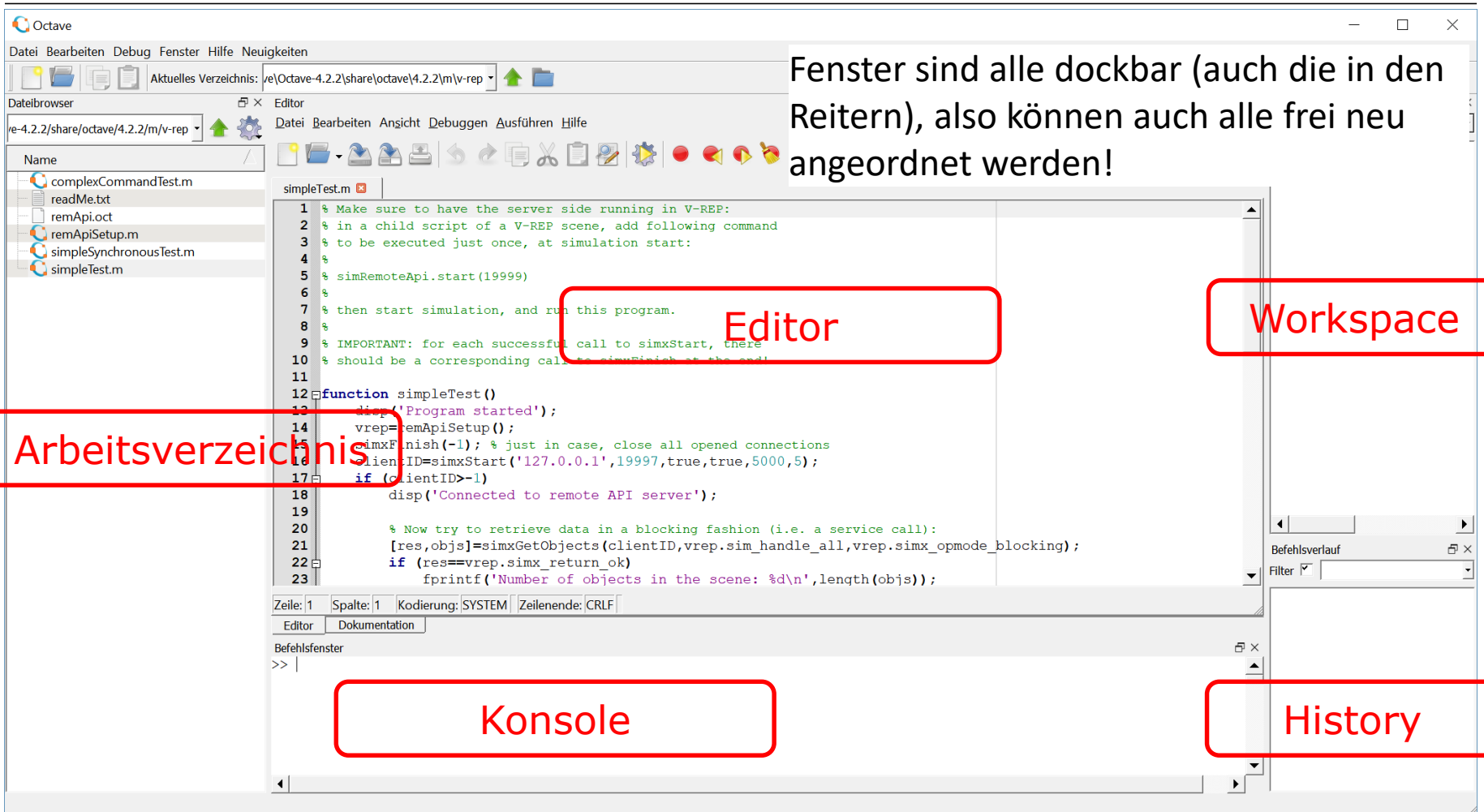


Octave vs. Matlab

- Octave unterstützt fast alle Funktionalitäten aus Matlab
- Syntax ist identisch zu Matlab, aber auch stark erweitert mit sogenannten „Syntactic Sugar“
 - Abhängig vom verwendeten Code-Stil ist Octave-Code häufig nicht lauffähig in Matlab

Octave

Benutzeroberfläche



Fenster sind alle dockbar (auch die in den Reitern), also können auch alle frei neu angeordnet werden!

Editor

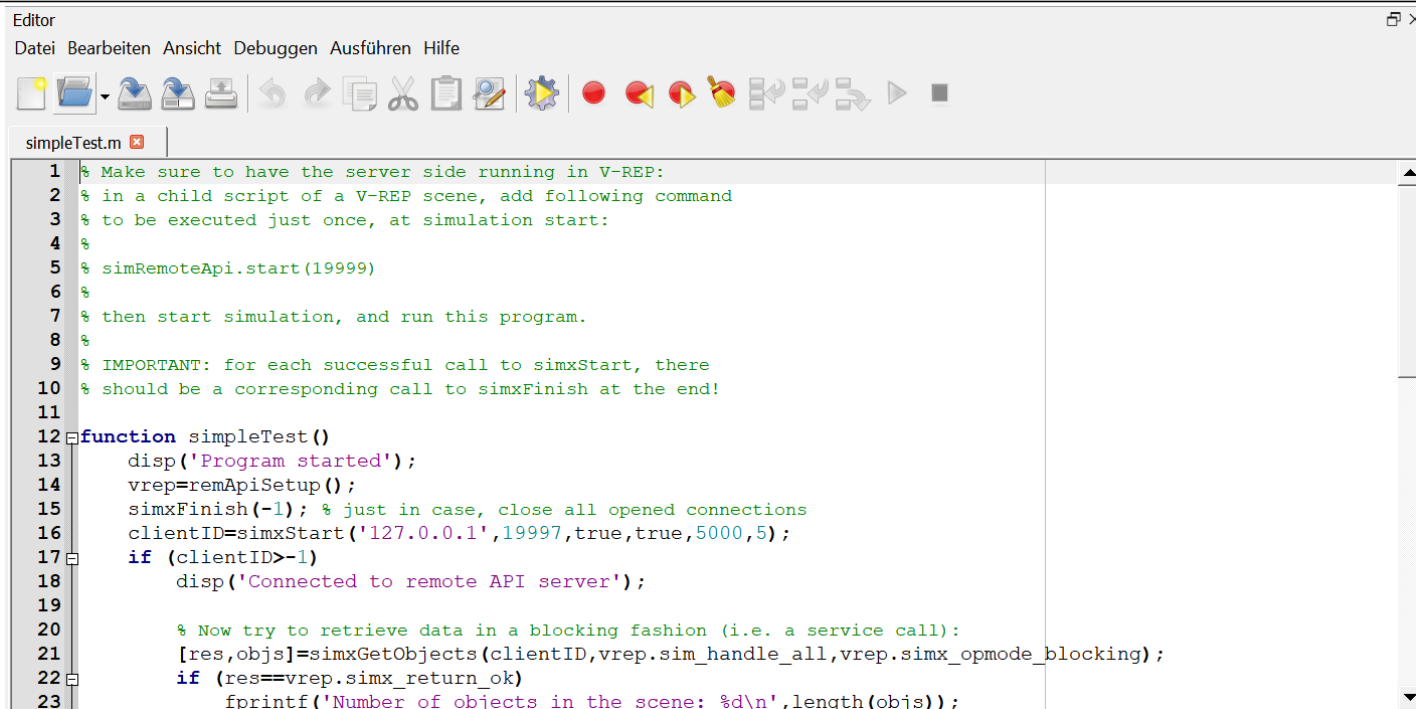
Workspace

Arbeitsverzeichnis

Konsole

History

```
1 % Make sure to have the server side running in V-REP:
2 % in a child script of a V-REP scene, add following command
3 % to be executed just once, at simulation start:
4 %
5 % simRemoteApi.start(19999)
6 %
7 % then start simulation, and run this program.
8 %
9 % IMPORTANT: for each successful call to simxStart, there
10 % should be a corresponding call to simxFinish at the end!
11
12 function simpleTest()
13     disp('Program started');
14     vrep=remApiSetup();
15     simxFinish(-1); % just in case, close all opened connections
16     clientID=simxStart('127.0.0.1',19997,true,true,5000,5);
17     if (clientID>-1)
18         disp('Connected to remote API server');
19
20         % Now try to retrieve data in a blocking fashion (i.e. a service call):
21         [res,objs]=simxGetObjects(clientID,vrep.sim_handle_all,vrep.simx_opmode_blocking);
22         if (res==vrep.simx_return_ok)
23             fprintf('Number of objects in the scene: %d\n',length(objs));
```



The screenshot shows the Octave Editor window with a menu bar (Datei, Bearbeiten, Ansicht, Debuggen, Ausführen, Hilfe) and a toolbar. The script 'simpleTest.m' is open, containing the following code:

```
1 % Make sure to have the server side running in V-REP:
2 % in a child script of a V-REP scene, add following command
3 % to be executed just once, at simulation start:
4 %
5 % simRemoteApi.start(19999)
6 %
7 % then start simulation, and run this program.
8 %
9 % IMPORTANT: for each successful call to simxStart, there
10 % should be a corresponding call to simxFinish at the end!
11
12 function simpleTest()
13     disp('Program started');
14     vrep=remApiSetup();
15     simxFinish(-1); % just in case, close all opened connections
16     clientID=simxStart('127.0.0.1',19997,true,true,5000,5);
17     if (clientID>-1)
18         disp('Connected to remote API server');
19
20         % Now try to retrieve data in a blocking fashion (i.e. a service call):
21         [res,objs]=simxGetObjects(clientID,vrep.sim_handle_all,vrep.simx_opmode_blocking);
22         if (res==vrep.simx_return_ok)
23             fprintf('Number of objects in the scene: %d\n',length(objs));
```

- Erstellung von Octave-Funktionen und Octave-Skripten
- Octave-Dateien werden analog zu Matlab mit der Endung `.m` gespeichert
- Datei und Funktion müssen den gleichen Namen haben und es kann normalerweise nur eine Funktion pro Datei angelegt werden

Konstanten

<code>pi</code>	% 3,14159...
<code>inf</code>	% unendlich
<code>nan</code>	% keine Zahl
<code>eps</code>	% kleinste Zahl

Operatoren

<code>+</code>	% Addition
<code>-</code>	% Subtraktion
<code>*</code>	% Multiplikation
<code>/</code>	% Division
<code>sqrt(x)</code>	% Quadratwurzel
<code>x^y</code>	% Potenz
<code>s:e</code>	% Folge von s bis
<code>s:i:e</code>	% e mit der Schritt-
	% weite 1 oder i
<code>.*</code> od. <code>./</code>	% komponentenweise
<code>~</code>	% Nicht Operator

Variablen

Es gibt zwar Datentypen, Variablen müssen jedoch nicht definiert, sondern können dynamisch initialisiert werden.

Datentypen

<code>4.5678</code>	% reelle Zahl
<code>4.3 + 5.6i</code>	% komplexe Zahl
<code>'Text'</code>	% Zeichenkette
<code>1</code>	% logische Eins
<code>0</code>	% logische Null

Skalare

```
>> a = 1;           % Unterscheidung zwischen Groß- und Kleinschreibung
>> A = 1;
>> b = a + A % Eingabe ohne ; am Ende gibt das Ergebnis direkt aus
b =
    2
```

Vektoren

```
>> C = [1, 2]        % Zeilenvektor
C =
    1    2
>> C = [1; 2]         % Spaltenvektor
C =
    1
    2
>> C = [1; 2] '       % Transponierte
C =
    1    2
```

Matrizen

```
>> M = [1, 2, 3; 4, 5, 6] % Matrix
```

```
M =  
    1  2  3  
    4  5  6
```

```
>> M = [1:3; 4:6] % Matrix
```

```
M =  
    1  2  3  
    4  5  6
```

Indizes

```
>> % Indizes sind 1-basiert (mathematisch Zählweise)
```

<pre>>> M(1)</pre>	<pre>>> M(1, 2)</pre>	<pre>>> M(2, :)</pre>	<pre>>> M(:, 1)</pre>	<pre>>> M(1, 1:2)</pre>
<pre>ans =</pre>	<pre>ans =</pre>	<pre>ans =</pre>	<pre>ans =</pre>	<pre>ans =</pre>
1	2	4 5 6	1	1 2
			4	

- **Erstellen von initialisierten Matrix- / Vektor-Blöcken:**

```
>> M1 = zeros(2,3)
```

```
M1 =  
    0    0    0  
    0    0    0
```

```
>> M2 = rand(2)
```

```
M2 =  
    0.52640    0.67637  
    0.57948    0.54348
```

```
>> M3 = ones(1,5)
```

```
M3 =  
    1    1    1    1    1
```

- **Konkatenation von Matrizen und Vektoren:**

```
>> M = eye(2)
```

```
M =  
Diagonal Matrix
```

```
    1    0  
    0    1
```

```
>> M = [M, ones(2,1); rand(1,3)]
```

```
M =  
    1.00000    0.00000    1.00000  
    0.00000    1.00000    1.00000  
    0.87866    0.30739    0.28320
```

Matrixfunktionen

```
>> eig(M);      % Eigenwerte der Matrix
>> det(M);      % Determinante der Matrix
>> inv(M);      % Inverse der Matrix
>> M^(-1);      % Inverse der Matrix
>> rank(M);     % Rang der Matrix
>> trace(M);    % Spur der Matrix
```

Gleichungssysteme

```
>> % Möglichkeiten zur Lösung eines linearen Gleichungssystems
>> %  $A * x = b$  mit der Matrix A und den Vektoren b und x
>> x = A \ b;
>> x = inv(A) * b;
>> x = A^(-1) * b;
```

```
>> % Matrix A initialisieren
>> A = [1, -1, 1; 2, 3, 0; 0, 3, 4];
>> % Vektor b initialisieren
>> b = [0; 2; 4];
>> % Gleichungssystem lösen (1. Möglichkeit)
>> x = A \ b
x =
    0.0769231
    0.6153846
    0.5384615
>> % Gleichungssystem lösen (2. Möglichkeit)
>> x = inv(A) * b
x =
    0.0769231
    0.6153846
    0.5384615
```

Octave

Beispiel: Darstellung von Funktionen

% Vektor x initialisieren

```
>> x = [0:0.1:2 * pi]'
```

```
x =
```

```
0
```

```
0.1
```

```
...
```

```
6.1
```

```
6.2
```

% Darstellung der Sinus-

% und Cosinus-Funktionen

```
>> plot(x, sin(x), 'ro-');
```

```
>> grid on;
```

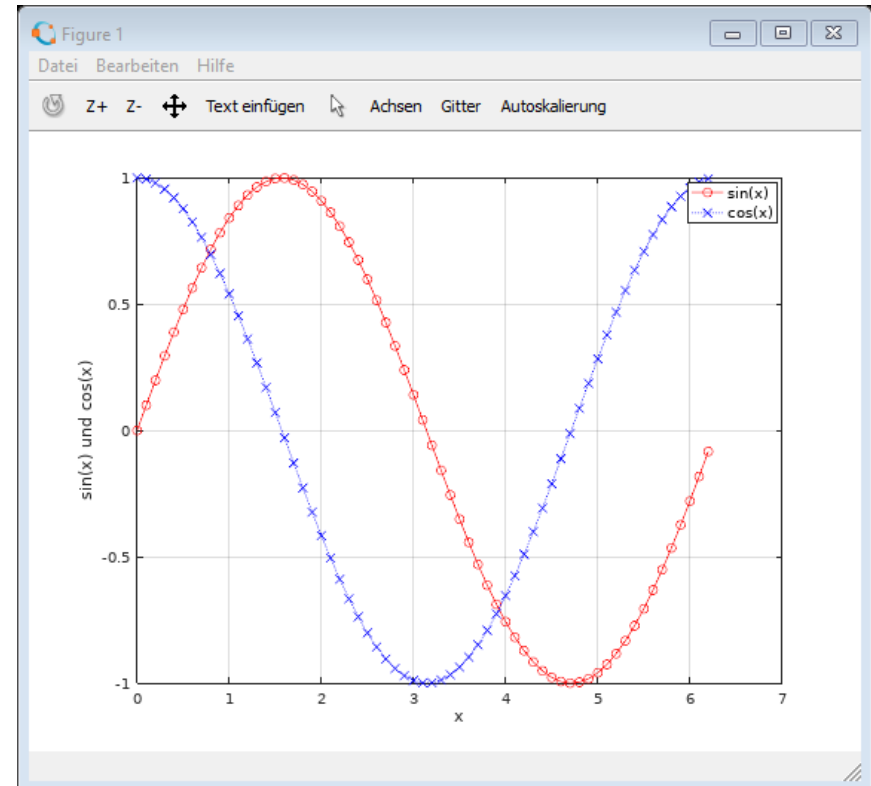
```
>> hold on;
```

```
>> plot(x, cos(x), 'bx:');
```

```
>> legend('sin(x)', 'cos(x)');
```

```
>> xlabel('x');
```

```
>> ylabel('sin(x) und cos(x)');
```



Form

```
>> for k = s:i:e      % k ist die Zählervariable
>>   Instruktion1;    % s ist der Startwert
>>   Instruktion2;    % i ist das Inkrement
>> end                % e ist der Endwert
```

Beispiel

```
>> n = 3; m = 6; a = zeros(n, m);
>> for i = 1:n
>>   for j = 1:m
>>     a(i, j) = j + i - 1;
>>   end
>> end
>> display(a)
a =
    1  2  3  4  5  6
    2  3  4  5  6  7
    3  4  5  6  7  8
```


Form

```
>> while Bedingung  
>>     Instruktion1;  
>>     Instruktion2;  
>> end
```

Beispiel

```
>> d = 1; f = 4;  
>> while d < f  
>>     display('d ist kleiner als f')  
>>     d = d + 1;  
>> end
```

Form

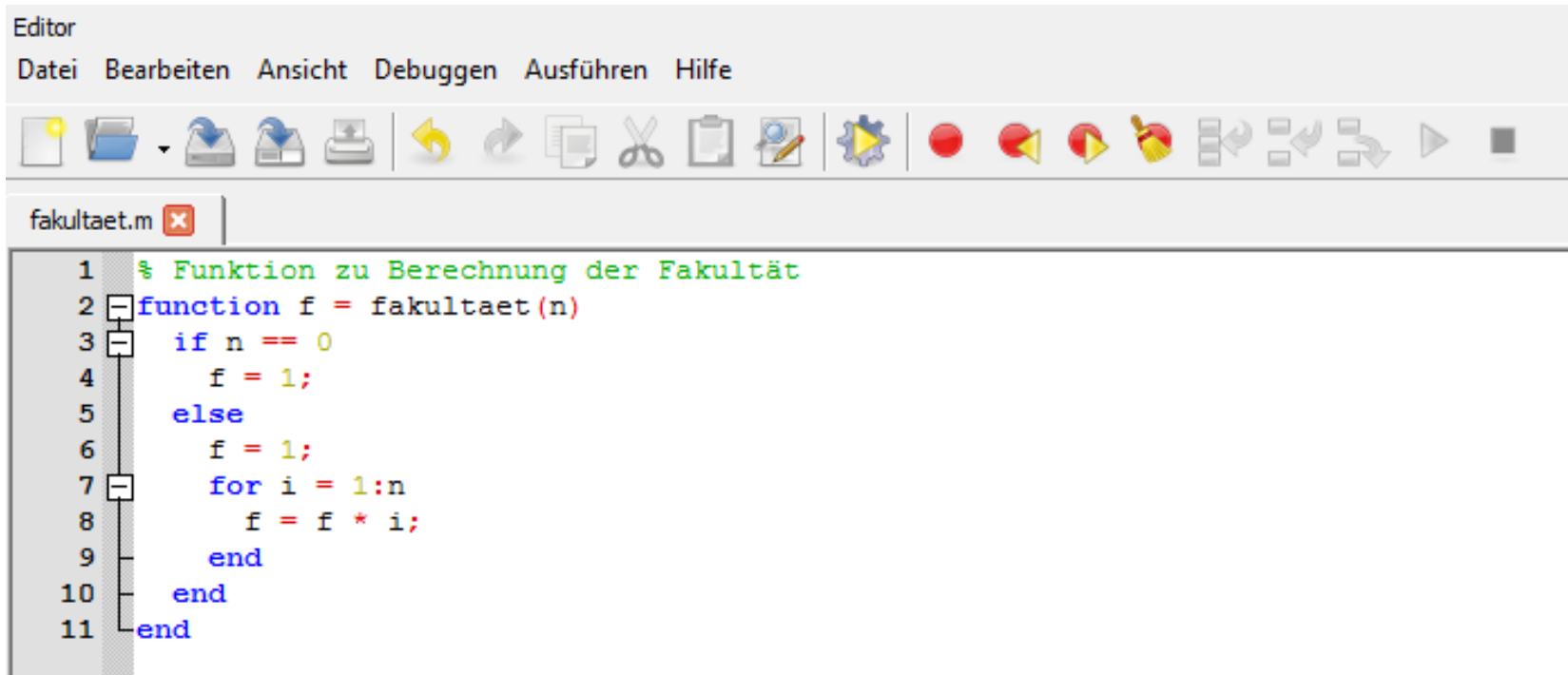
```
>> if Bedingung1
>>   Instruktion1;
>> elseif Bedingung2
>>   Instruktion2;
>> else
>>   Instruktion3;
>> end
```

Beispiel

```
>> m = 2;
>> if m == 1
>>   display('m ist 1')
>> else
>>   display('m ist nicht 1');
>> end
```

Octave

Beispiel: Anlegen von Funktionen



The screenshot shows the Octave Editor interface. The menu bar includes 'Datei', 'Bearbeiten', 'Ansicht', 'Debuggen', 'Ausführen', and 'Hilfe'. The toolbar contains various icons for file operations, editing, and execution. The editor window is titled 'fakultaet.m' and contains the following MATLAB code:

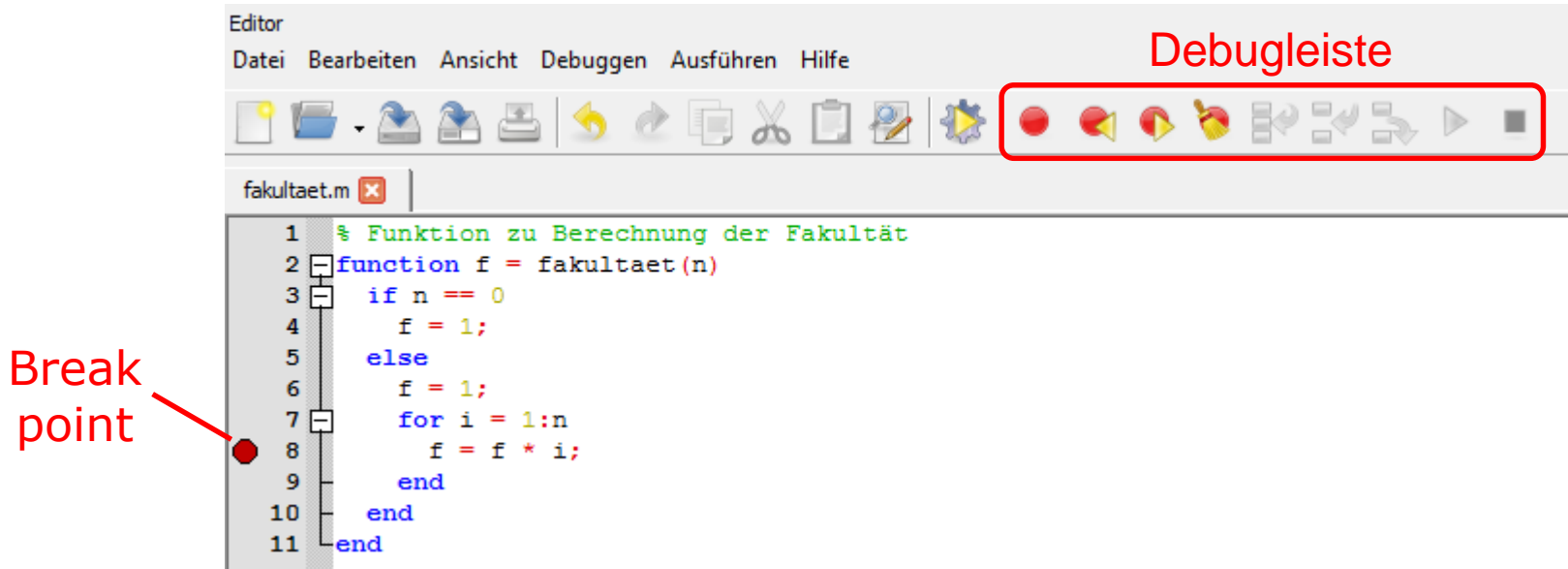
```
1 % Funktion zu Berechnung der Fakultät
2 function f = fakultaet(n)
3     if n == 0
4         f = 1;
5     else
6         f = 1;
7         for i = 1:n
8             f = f * i;
9         end
10    end
11 end
```

```
>> x = fakultaet(5)
```

% Aufruf der Funktion fakultaet

```
x =
```

```
120
```



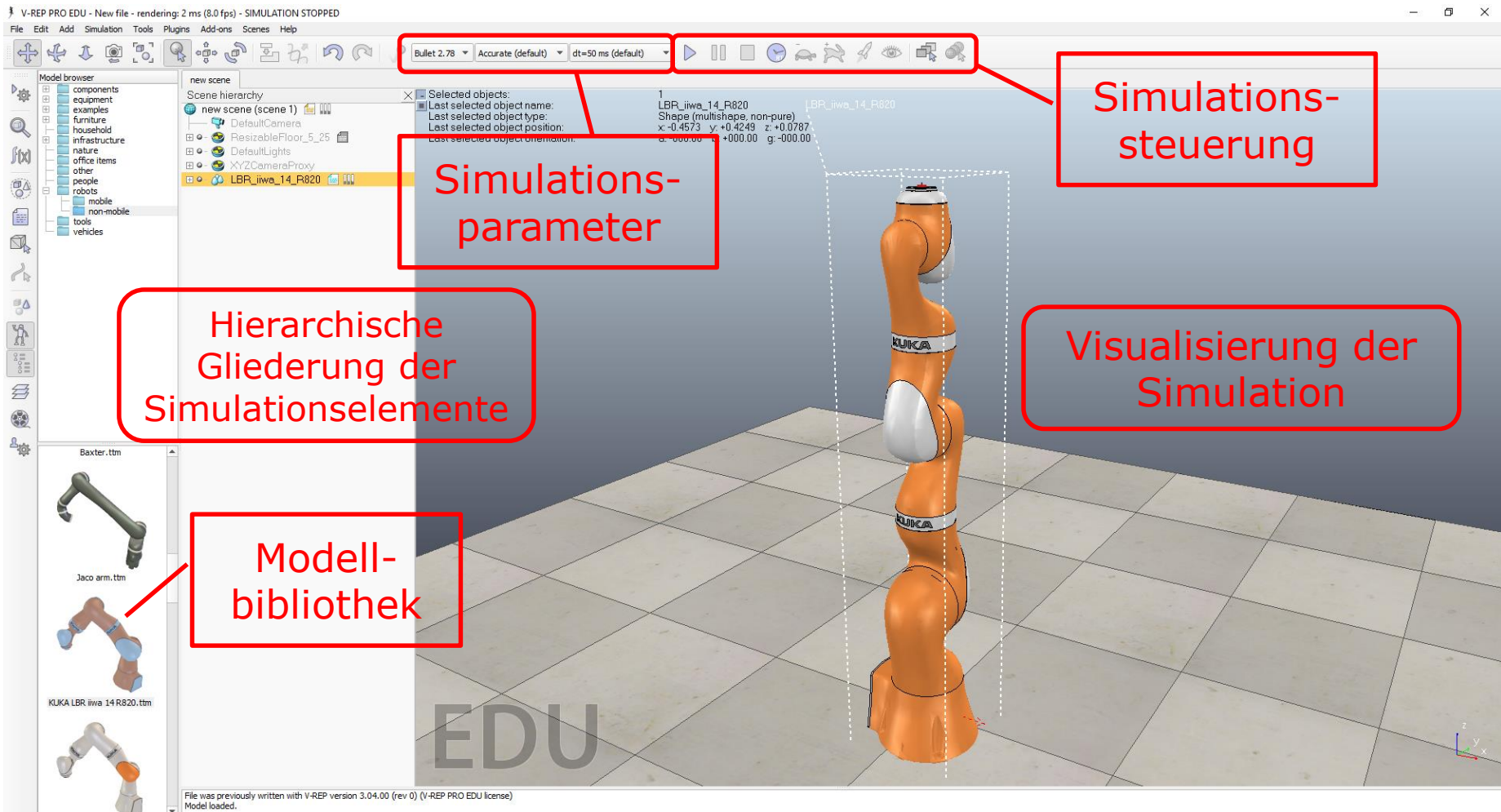
- Klick auf „roten Punkt“ erzeugt Breakpoint an Cursorposition
- Octave hält nach Ausführung der Funktion bei jedem Durchlauf am Breakpoint an
- Wert der Variablen kann dann überprüft und modifiziert werden
 - Workspace
 - Command Window

Einführung V-REP

- V-REP wird im Rahmen der Übungen nur als **Simulationswerkzeug** eingesetzt
- Umgebungen/Szenarien werden zur Verfügung gestellt
- **Arbeit findet hauptsächlich in Octave statt!**



Einführung V-REP



- Probieren geht [hier](#) über Studieren
- Nutzen Sie die Zeit vor der ersten Programmierübung und machen Sie sich mit **Octave** und **V-REP** vertraut
- Stellen Sie rechtzeitig sicher, dass Sie Zugang zu **Octave** und **V-REP** haben, um an den Programmierübungen teilnehmen zu können
 - Frühzeitige Einrichtung der Arbeitsumgebung empfohlen, um bei Problemen keine Zeit zu verlieren