Matlab introduction

Matthias Hoffmann, Room 10.136 Matthias.Hoffmann @ cs.fau.de

Exercise

Read through the following tutorial. Use it to fill in the missing lines in *matlabintro_exercise.m*.

Tutorial

Start matlab: /local/matlab/bin/matlab

General Information

- Matlab is NOT a compiler! It executes the source code line by line.
- No compilation is necessary!

Command window

Command line to input matlab commands or for output of matlab programm.

Workspace

By default all variables are of double precision. No variable type declaration is necessary!
E.g. type in the command window: ≫ m=1 (and press return)
The variable m of type double is now defined in the current workspace.
Examples:
≫ t=1.0;
≫ txt='Hallo'; % a text string

dbstop error

Help

Start Help: Help → Matlab Help Take a look at the frame in the help window:

- Contents (good command overview)
- Search (search for commands)

- Demo (useful matlab demo programs)

How can I create a new matlab program? Create a new .m file (e.g. *Test.m* file)

How can I create a new .m file?

1. File → New → m File
2. Write your matlab code. (see Let's start programming)
3. Start the program: Debug → Run . . . or press F5

Let's start programming

```
% This is a comment
≫ t=1 % defines t with value 1 and print it in the command window
≫ r=10.5; % defines r value 10.5, but don't print it (because of the semicolon!)

% How can I define a row vector?
Vrow = [1 2 3];

% How can I define a column vector?
Vcol = [1; 2; 3]; % print the vectors in the command window
Vrow
Vcol

% Multiplication
Vmults = Vrow * Vcol; % results a scalar
Vmultm = Vcol * Vrow; % results a 3×3 matrix

% Element by element multiplication (matrix dimensions must agree!)
Vr = Vrow.*Vrow;
Vc = Vcol.*Vcol;

% How can I define a 3×4 matrix (row×column)?
Ma = [1 2 3 4; 5 6 7 8; 9 10 11 12]

% Access the matrix elements: Ma(row, column)
Ma(2,3) % returns 7

% Define a N×M zero matrix
N = 5;
M = 8;
Maz = zeros(N,M)

% Define a N×M one matrix
Mao = ones(N,M)

% Create a N×M identity matrix
```

```
E = eye(N,M)

% Identity square matrix
Es = eye(N)

% Create a matrix with random values (uniformly distributed)
Mar = rand(N,M)

% Draw a matrix as a image
imagesc(Mar)

% Create a vector with values from 1 to 100
v1 = [1:100]

% Transpose a matrix/vector
v1'
Ma'

% How can I create a test image with a circle?
% 1. Define the image matrix
imageSizeX = 256;
imageSizeY = 256;
img = zeros(imageSizeX,imageSizeY);

% 2. Define the image grid positions
[Y, X] = ndgrid(1:imageSizeY,1:imageSizeX);
% Compare to: [X,Y] = meshgrid(1:imageSizeX,1:imageSizeY);
X
Y

% 3. Evaluate the circle equation
R = 50; % radius
% Set all pixel inside the circle to 100
img(((X-(imageSizeX/2)).^2+(Y-(imageSizeY/2)).^2)<R) = 100;

figure
imagesc(img)

% Set the image border to 100. <:> means all elements of this dimension.
img(:,1) = 100;
img(:,imageSizeY) = 100;
img(1,:) = 100;
img(imageSizeX,:) = 100;

% Alternative using <end>
img(:,1) = 100;
img(:,end) = 100;
```

```
img(1,:) = 100;
img(end,:) = 100;

% How can I plot a 1D function?
% Define evaluation positions
ss = 0.5; % step size
x = [-100:ss:-1 1:ss:100]; % exclude the zero!
% equal to x = [−100 − 99.5 − 99 . . . − 1  1  1.5  2  2.5 . . . 100];

% Compute the signal
A = 10.5;
% Use the element by element division operator ./
S = a.*sin(x)./x; % sinc function

% Plot the function
figure(3); % new figure is generated which can be adressed by 3
plot(S);

% Plot subfigures
figure(5);
subplot(4,2,3); % 4×2 plottings inside a window, 4 rows, 2 columns, third position
imagesc(img);
title('something'); % title of the plot
xlabel('What axes is this?'); % label of the 'normal' x axes

% Round to int
ai = floor(a)

% How can I perform a 2D convolution?
% E.g. mean value
mask = (1/9).*[1 1 1; 1 1 1; 1 1 1];
% 3×3 convolution mask: which filter?
imgCon = conv2(img,mask);

% Create a new plot with two subplots
subplot(1,2,1) % one row, two columns and plot the next image to position one
imagesc(imgCon);
colormap('gray'); % the image is shown with gray values; what are other colour maps?
subplot(1,2,2) % plot position two
imagesc(img);

% Where is the origin in the Matlab figures?
% Upper left corner, horizontal: y axes, vertical: x axes

% Check the size of the original image img and imgCon!
size(img)
size(imgCon)
```

```
% Because of the convolution the image imgCon is extended by one in each dimension!
% This is because of the 3×3 convolution mask.
% A convolution mask 5×5 would result in an extension by 2 in each dimension.

% How can I subtract the two image matrices of different size?
imgDiff = imgCon([2:end-1],[2:end-1])-img;
subplot(2,2,3)
imagesc(imgDiff)
title('Difference image');

% How can I read an image from disc?
filename = 'heartDefect.img';

% Convert (force) the image to double
imgHD = double(imread(filename));

subplot(2,2,4)
imagesc(imgHD)
title('HeartDefect image');

% A simple loop (iteration)
Mit = zeros(100,1);
for = i=1:100
      Mit(i,1) = i;
end
% with stepsize
stepsize = 2;
for i=1:stepsize:100
      Mit(i,1) = i;
end

% If statement
i1 = 1;
i2 = 2;
if((i1<i2) & (~i1|i2))
      % do something
end

if(i1)
      % do something
else
      % do something else
end

% Quiver plot of the first image derivative
imgX = img([2:end,end],:)-img(:,:); % x-direction
```

```
imgY = img(:,[2:end,end])-img(:,:); % y-direction
subplot(2,2,4) imagesc(img)
title('image');
hold
quiver(X,Y,imgX,imgY);

% Other useful commands
≫ help fft
≫ help fft2 ≫ help conv
≫ help eigs % find eigenvalues and -vectors
≫ help svd
≫ help mesh, meshc, meshz
≫ help meshgrid
≫ help elmat % elementary functions for matrices
≫ help elfun % elementary functions
≫ help specfun % special functions
≫ help ops % logical functions
```