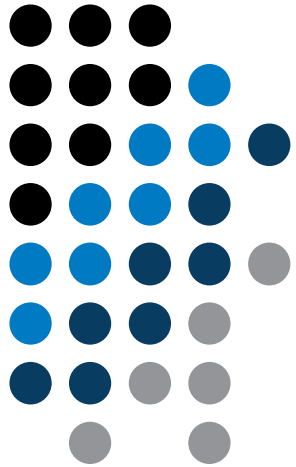
A0B17MTB – Matlab

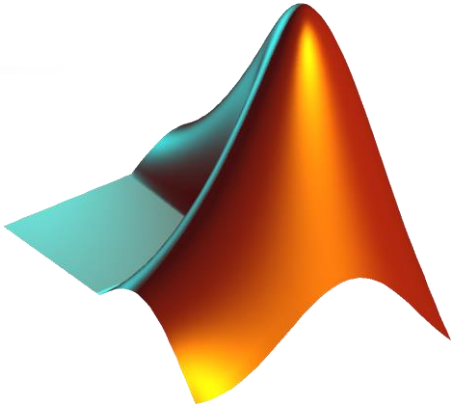
Introduction



Miloslav Čapek
miloslav.capek@fel.cvut.cz
Filip Kozák, Viktor Adler, Pavel Valtr

Department of Electromagnetic Field
B2-626, Prague

You will learn …



What is MATLAB?

Why to learn MATLAB?

Details of the A0B17MTB course

Recommended literature, further resources

First steps in MATLAB

What is MATLAB?

MATLAB is…

- high-level programming language (*4th gener. language*)
- interpreted language (not compiled, but... JIT)
  - intended mainly for numerical computing (nevertheless includes MuPAD symbolic kernel)
- philosophy: kernel + tool boxes + user-defined functions → wide application
  - wide possibilities of linking with other tools (Java, C++, Fortran, Python, .NET, Excel, physical- / multi-physical softwares)
- speed (of well written) algorithm comes near to that of C++
- excellent for „fast prototyping“
  - Matlab does not require variables declaration (not always the advantage)
- multi-license for CTU
  - Available for students as well!
  - `download.cvut.cz` + main access password
  - `fel.cvut.cz` → computer network → Multi-license software at CTU

Why to learn MATLAB?

- Matlab is a worldwide standard
- used by more than 5000+ universities worldwide
- licenses used by thousands of corporations in aviation, biotechnology, electronics, cybernetics, mechanical engineering, finance, ...
- knowledge of Matlab can be used in other courses at the University as well as in professional life

Where to make use of Matlab?

- data processing and visualization during laboratory exercises
- when elaborating diploma works
- seminar exercises (signals, algorithm development, ...)
- theory verification (mathematics and physics classes, electromagnetic field, electronic circuits, ...)

- studying abroad (Erasmus, Sokrates)

$\Rightarrow$ **“everywhere“** :)

Historical development of MATLAB

- 70's
  - Cleve Moler, Matlab used instead of Fortran
  - MATrix LABoratory → matrix is the basic data structure
  - Fortran-based syntax
- 1983
  - Jack Little rewriting Matlab in C
  - new functionality and new mathematical libraries added
- 1984 (Matlab is so far for free!)
  - MathWorks founded in 1984
  - `http://www.mathworks.com/`
- 2004
  - Matlab used by more than 1 million of active users
- now...
  - ... R2015b is the newest version of Matlab
  - local distribution: Humusoft

see: `http://www.mathworks.com/company/aboutus/founders/clevemoler.html`

Alternatives to MATLAB

- Fortran – most of the libraries still in Fortran, used mostly by physisists
- Python – for free, fast and intuitive; Spyder provides MATLAB-like features
- Mathematica – symbolic and numerical calculations, excellent symbolic kernel, extensive applicability, mostly for mathematicians and physicists
- Maple – symbolic and numerical calculations
- MathCad –used for symbolic and numerical calculations, slightly out-of-date
- Octave – for free, syntax and functionality similar to Matlab, not so extensive, smartphone executable
- R – for free, designed particularly for statistical applications
- Scilab – Matlab-like, open documentation
- Derive – small, fast, Casio calculator executable

Alternatives to MATLAB

- Matlab vs. C/C++
  - optimal language strongly depends on the application
  - C/C++ faster in general, Matlab, on the other hand, provides implicit parallelism
  - general principle: Matlab more than suitable for everything except commercial compiled code (especially Matlab 6.5 and above: JIT + Real-Time Type Analysis)

- Matlab vs. Fortran
  - Matlab has wider support, more intuitive syntax
  - speed of a well written code is (usually, at least) comparable
  - utilization of Fortran is on the decline

- Matlab vs. Python
  - Matlab offers significant support thanks to MathWorks, Matlab File Exchange
  - Python entirely for free, it's becoming more and more popular

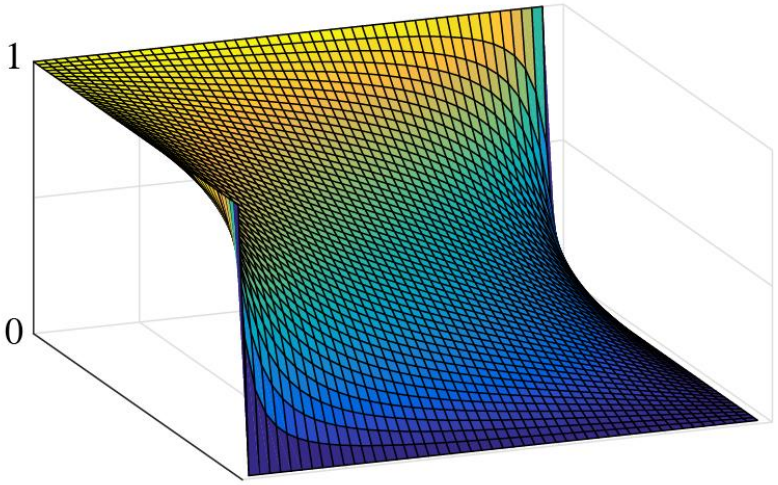
A0B17MTB

- 13 weeks (14th week is a 'reserve')
  - 10 blocks with new theory, 1 block of bonuses, 1 block of examples
- conditions of credit award:
  - to hand in a project (next-to-last week of the semester, 60 points)
    - **competition assignment** (see next slide)
  - to pass a test, 20 points (min. 50%, next-to-last week)
    - on top of that two short tests during semester, 20 points (min. 10 points are needed)
    - 3 bonus examples during the semestr, 5 bonus points
  - max. 2 missed classes (more absences only after prior arrangement)
    - any lecture can be substituted
- could happen that not all of the stuff of the course will be presented, because of time constraint – understanding the basics is a priority
  - bonus stuff (slides) available for advanced students

Data types	Code execution	Visualization	Relation and logical operators
Matrix operations	User scripts and functions	Numerical methods	Symbolic math

Competition assignment

Jacobi method



- see https://cw.fel.cvut.cz/wiki/courses/a0b17mtb/start
  `> projects > seznam_projektu`

- project can be selected by any number of students
- conditions:
  - project is completed according the assignment → credit award
  - project is the best one → winning the competition
    - prizes for the first three winners

A0B17MTB – Course syllabus

1	Introduction, information on the course, MATLAB workspace, basic arithmetic operators, basic functions
2	Complex numbers, complex matrix design, matrix operations, element-by-element operations, introduction to vectorization, matrix dimension
3	Indexing, data type and size, output format, MATLAB Editor, script design
4	Cycles, relation and logical operators, cycles vs. vectorization, control flow #1
5	Control flow #2, visualization in MATLAB #1, debugging #1
6	Set operations, sorting, searching, user-defined functions #1
7	User interface (main functions, subfunctions, nested functions, anonymous functions)
8	Struct, Strings, ‘eval’ and ‘feval’ functions, MATLAB path
9	Visualization in MATLAB #2, GUI #1
10	GUI #2
11	Date and time functions, error handling, cell, I/O, basics of symbolic computations
12	MATLAB profile, p-code, numerical accuracy, publishing MATLAB code, programming style guidelines
13	Exercises , test
14	/reserve/

A0B17MTB – Deadlines

1	call for project proposals
2	
3	*bonus example (1-3 points)*, list of projects, discussion on own topics
4	*short test (approx. 10-15 min) aimed on solving given problem in Matlab, 10 points*
5	project choice
6	
7	*bonus example (1-3 points)*
8	*short test (approx. 10-15 min) aimed on solving given problem in Matlab, 10 points*
9	
10	*bonus example (1-3 points)*
11	
12	
13	*project hand-in (next-to-last week of the semester)*, *test*
14	test evaluation, credit award

Credit award

	Points	Min. points
Bonus example #1	2	10
Short test #1	10	
Bonus example #2	1	
Short test #2	10	
Bonus example #3	2	
Test	20	10
Project	60	30

Grade	Points
A	90–100
B	80–89
C	70–79
D	60–69
E	50–59
F	0–49

A0B17MTB – Schedule

- harmonogram of SS 2015/2016 (also on the web page):

	8. týden			9. týden			10. týden			11. týden			12. týden		
	22. 2.	23. 2.	24. 2.	29. 2.	1. 3.	2. 3.	7. 3.	8. 3.	9. 3.	14. 3.	15. 3.	16. 3.	21. 3.	22. 3.	24. 3.
	PO 16:15	UT 16:15	ST 16:16	PO 16:15	UT 16:15	ST 16:16	PO 16:15	UT 16:15	ST 16:16	PO 16:15	UT 16:15	ST 16:16	PO 16:15	UT 16:15	ST 16:16
poznámka							Filip - hory								
master	M	M	M	M	M	M	P	P	P	F	F	F	F	F	F
slave	F	F	F	F	F	F	V	V	V	V	V	V	M	M	M
náplň	1 (úvod)	1 (úvod)	1 (úvod)	2 (matice)	2 (matice)	2 (matice)	3 (indexace)	3 (indexace)	3 (indexace)	4 (relac. op., cykly)	4 (relac. op., cykly)	4 (relac. op., cykly)	5 (if, vizualizace	5 (if, vizualizace	5 (if, vizualizace
harmonogram							bonusový příklad	bonusový příklad	bonusový příklad	1. písemka	1. písemka	1. písemka	zadání projektů	zadání projektů	zadání projektů

	13. týden			14. týden			15. týden			16. týden			17. týden		
	28. 3.	29. 3.	30. 3.	4. 4.	5. 4.	6. 4.	11. 4.	12. 4.	13. 4.	18. 4.	19.4.	20. 4.	25. 4.	26. 4.	27. 4.
	PO 16:15	UT 16:15	ST 16:16	PO 16:15	UT 16:15	ST 16:16	PO 16:15	UT 16:15	ST 16:16	PO 16:15	UT 16:15	ST 16:16	PO 16:15	UT 16:15	ST 16:16
poznámka	Velikonoční pondělí						Mila, Pavel - EuCAP								
master		V	V	V	V	V	V	V	V	V	F	F	F	F	F
slave		M	M	M	F	F	F	F	F	F	M	M	M	M	M
náplň		6 (mnoz. op.)	6 (mnoz. op.)	6 (mnoz. op.)	7 (funkce 2)	7 (funkce 2)	7 (funkce 2)	8 (textové řetězce)	8 (textové řetězce)	8 (textové řetězce)	9 (gui1)	9 (gui1)	9 (gui1)	10 (gui2)	10 (gui2)
harmonogram					bonusový příklad	bonusový příklad	bonusový příklad	2. písemka	2. písemka	2. písemka				bonusový příklad	bonusový příklad

	18. týden			19. týden			20. týden			21. týden						soutěž
	2. 5.	3. 5.	4. 5.	9. 5.	10. 5.	11. 5.	16. 5.	17. 5.	18. 5.	23. 5.	24. 5.	25. 5.				
	PO 16:15	UT 16:15	ST 16:16	PO 16:15	UT 16:15	ST 16:16	PO 16:15	UT 16:15	ST 16:16	PO 16:15	UT 16:15	ST 16:16				
poznámka						rektorský den										bude doplněno (viz web)
master	F	V	V	V	M		M	M	M	M,V,F	M,V,F	M,V,F				
slave	M	F	F	F												
náplň	10 (gui2)	11 (bonusy)	11 (bonusy)	11 (bonusy)	12 (velké příklady)		12 (velké příklady)	13 (zápočet)	12 (velké příklady)	13 (zápočet)	14 (rezerva)	13 (zápočet)				
harmonogram	bonusový příklad						test	test	test	zápočet	zápočet	zápočet				soutěž

Náplň předmětu:

1 (úvod)	5 (if, vizualizace	9 (gui1)	13 (zápočet)
2 (matice)	6 (mnoz. op.)	10 (gui2)	14 (rezerva)
3 (indexace)	7 (funkce 2)	11 (bonusy)	
4 (relac. op., cykly)	8 (textové řetězce)	12 (velké příklady)	

zadání projektů	1. písemka	2. písemka	test	zápočet	soutěž	bonusový příklad

Pozn.: bonusový příklad je za 1-3b a vybrán ze šedých příkladů (případně celá mimo sl

F - Filip

M - Míla

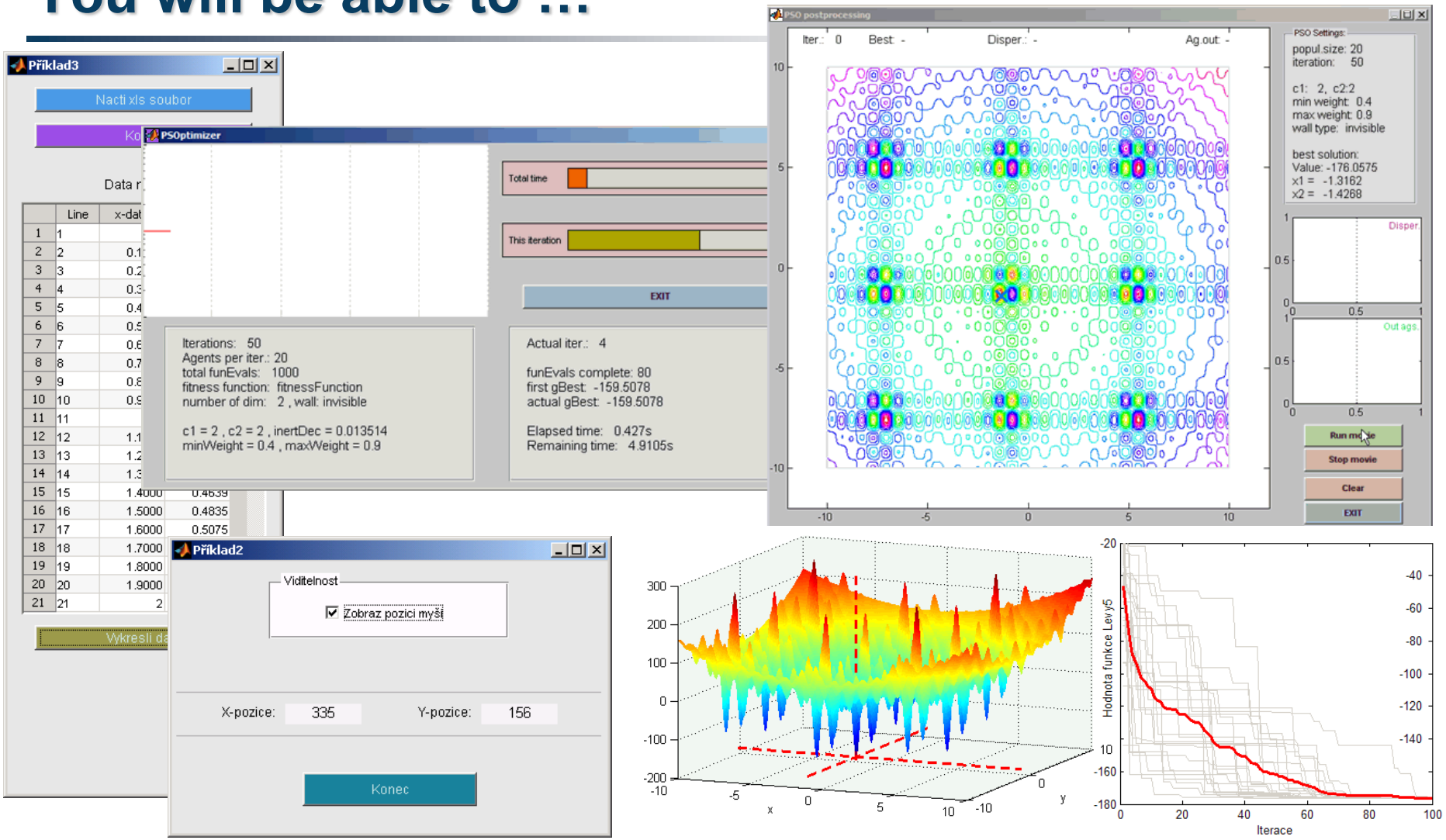
V - Viktor

- this is how the bonus slides look like (see the background color...)

A0B17MTB – Principles

- the aim of the course is to teach you something – if the presentation is to fast, be heard
- if you have an idea / proposal how to solve a problem in a more efficient way, put it forward
- can happen that the lecturer is not able to answer your question immediately, in that case the answer will be provided during the next lecture

You will be able to …



- see http://elmag.org/cs/Matlab/projekty for the previous students' projects

Recommended literature, resources

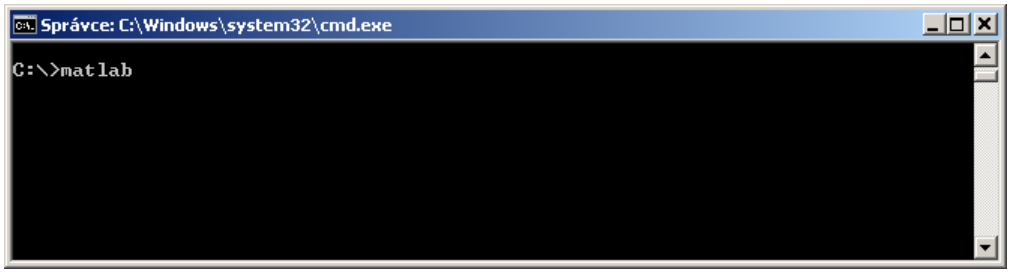
- Matlab documentation

```
>> doc % opens the help browser
```

- Basic web-based textbooks on Matlab (so called primers)
  - www.mathworks.com/help/pdf_doc/matlab/getstart.pdf
  - http://artax.karlin.mff.cuni.cz/~beda/cz/matlab/primercz/matlab-primer.html
- Attaway, S.: Matlab – A Practical Introduction to Programming and Problem Solving, 3rd ed.
  - available at Department's library
- Hahn, B. H., Valentine, D. T.: Essential Matlab, 5th Ed.
  - available at Department's library
- other literature and sources will be mentioned during the semester...

Launching Matlab

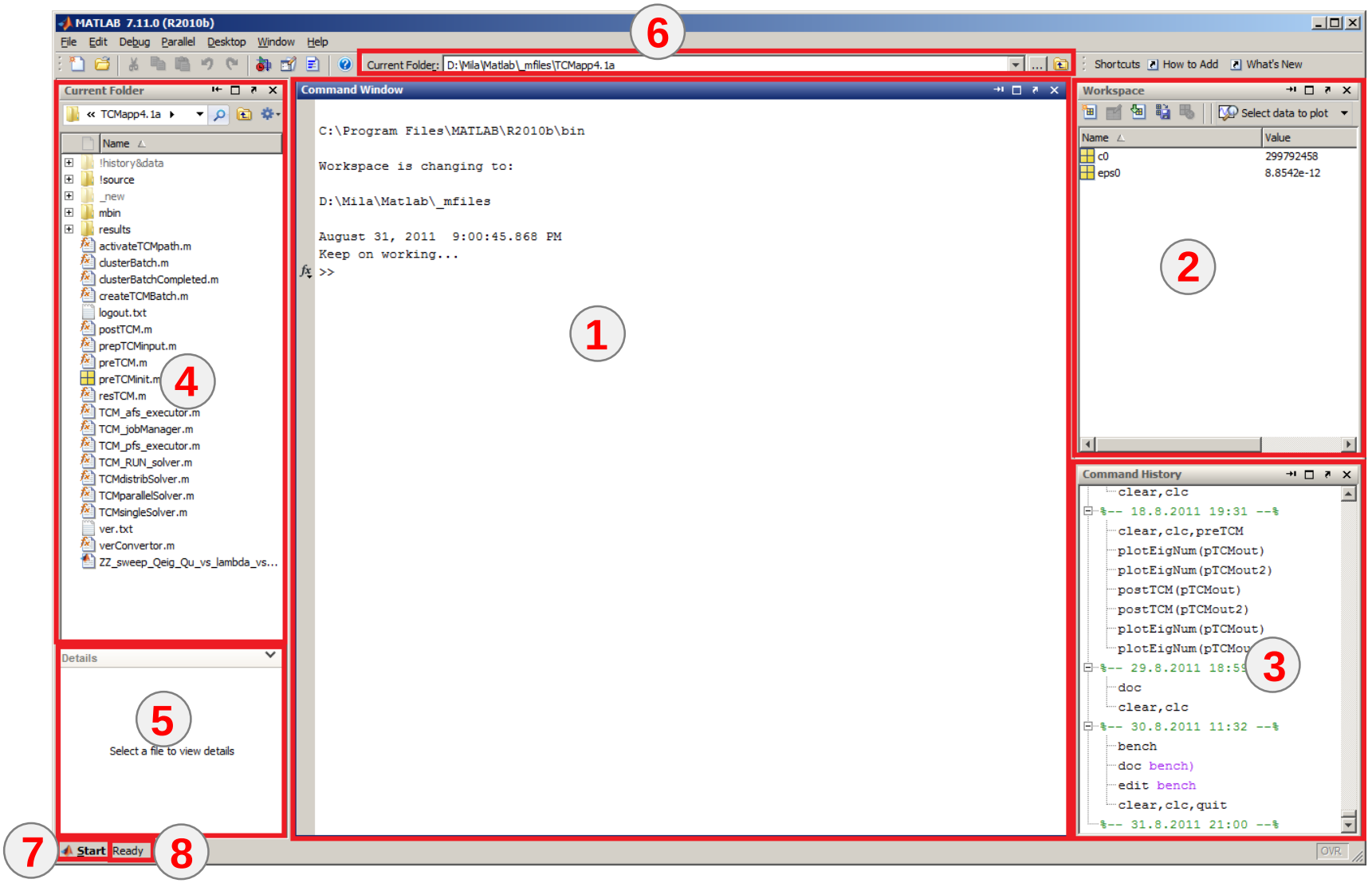
- command line
  - `matlab`



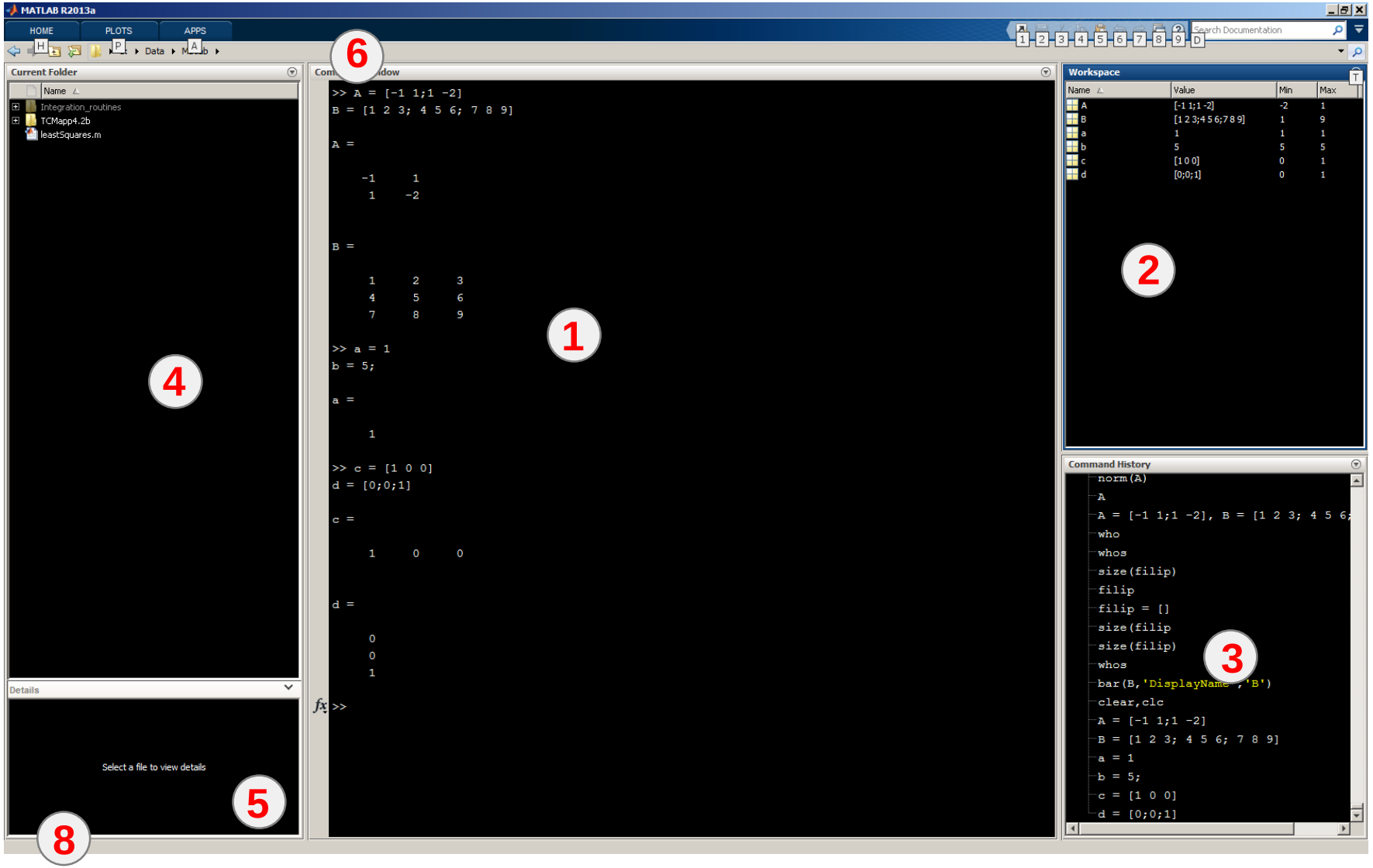
- Matlab can be launched with a set of optional parameters (see later)
  - `matlab -r "test(10)"`

- version dependent, up to 500MB RAM (win7) per matlab thread

The Matlab Environment (≤ R2011b)

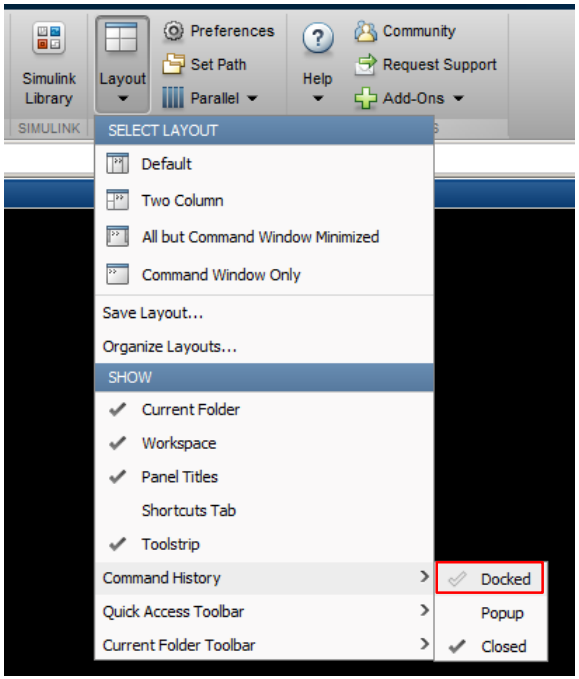


The Matlab Environment (≥ R2011b)

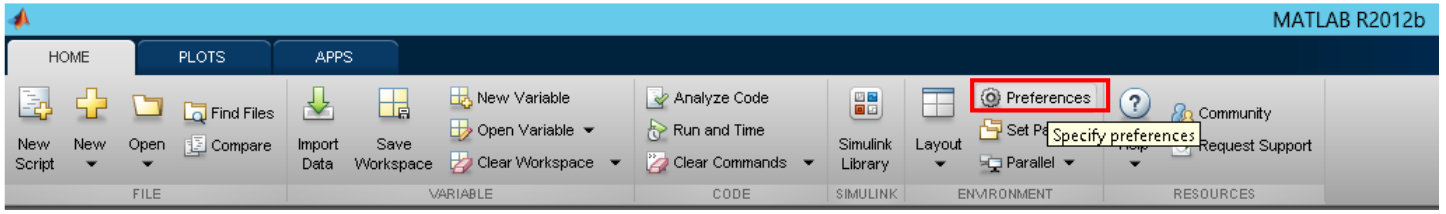


The Matlab Environment – panels

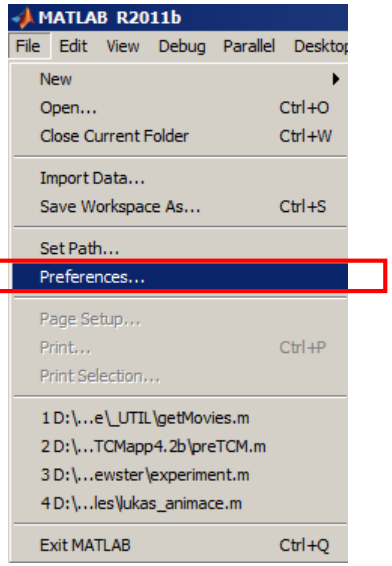
1. Command Window (CTRL+0)
2. Workspace (CTRL+3)
3. Command History (CTRL+1) – not activated in case of ≥ R2015a; to activate...
4. Current Folder (CTRL+2)
5. Current Folder – Details
6. Current Folder (with history)
7. Start (Windows like), only for ≤ Matlab R2011b
8. status



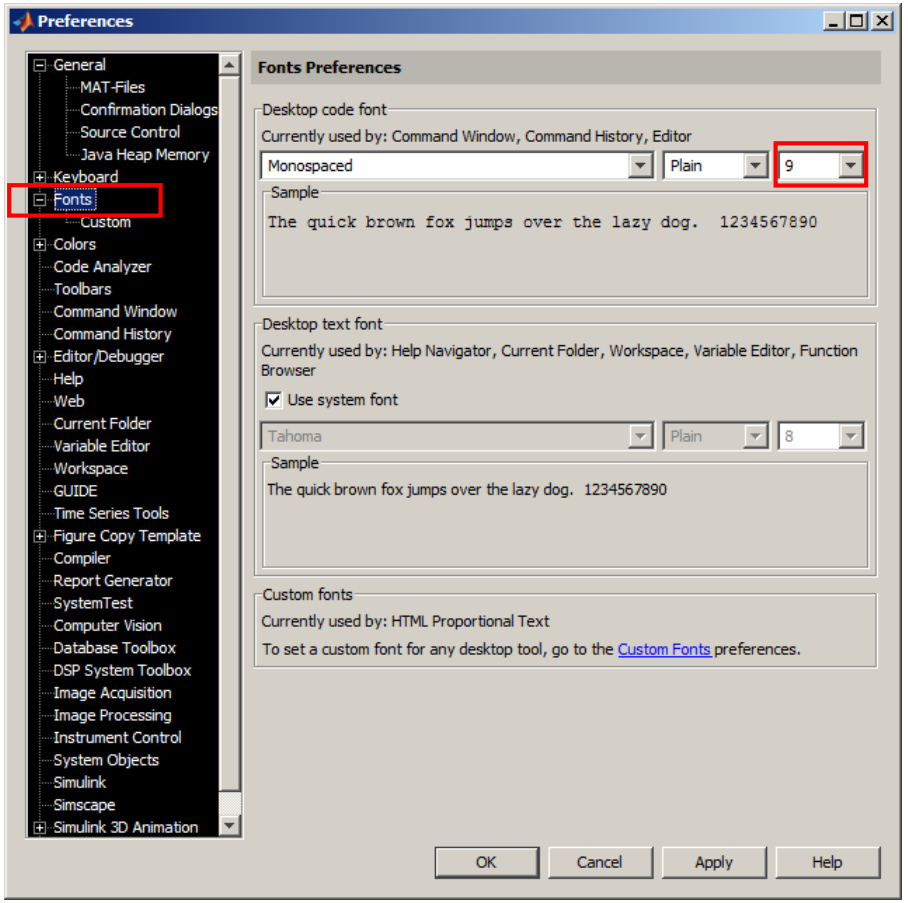
Environment setting – basics



- Matlab R2012a and later
  - ribbon menu
- Matlab R2011b and older




```
>> preferences
```



- Font size

Matlab termination

- always terminate Matlab in the command window

```
>> quit % terminates Matlab (and all windows)
>> exit %     -//-
```

- more advanced options (see documentation)

```
>> quit cancel
>> exit force
```

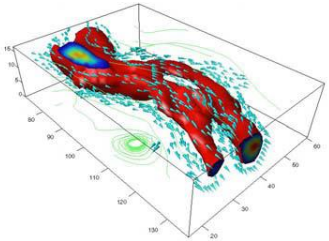
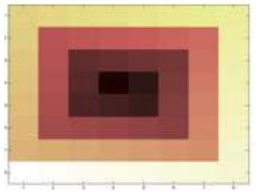
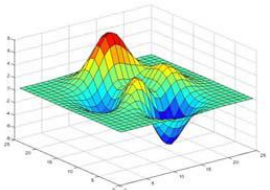
Command line, documentation

```
>> doc % opens documentation window
```

```
>> help % Matlab help
```

```
>> demo % tutorials
```

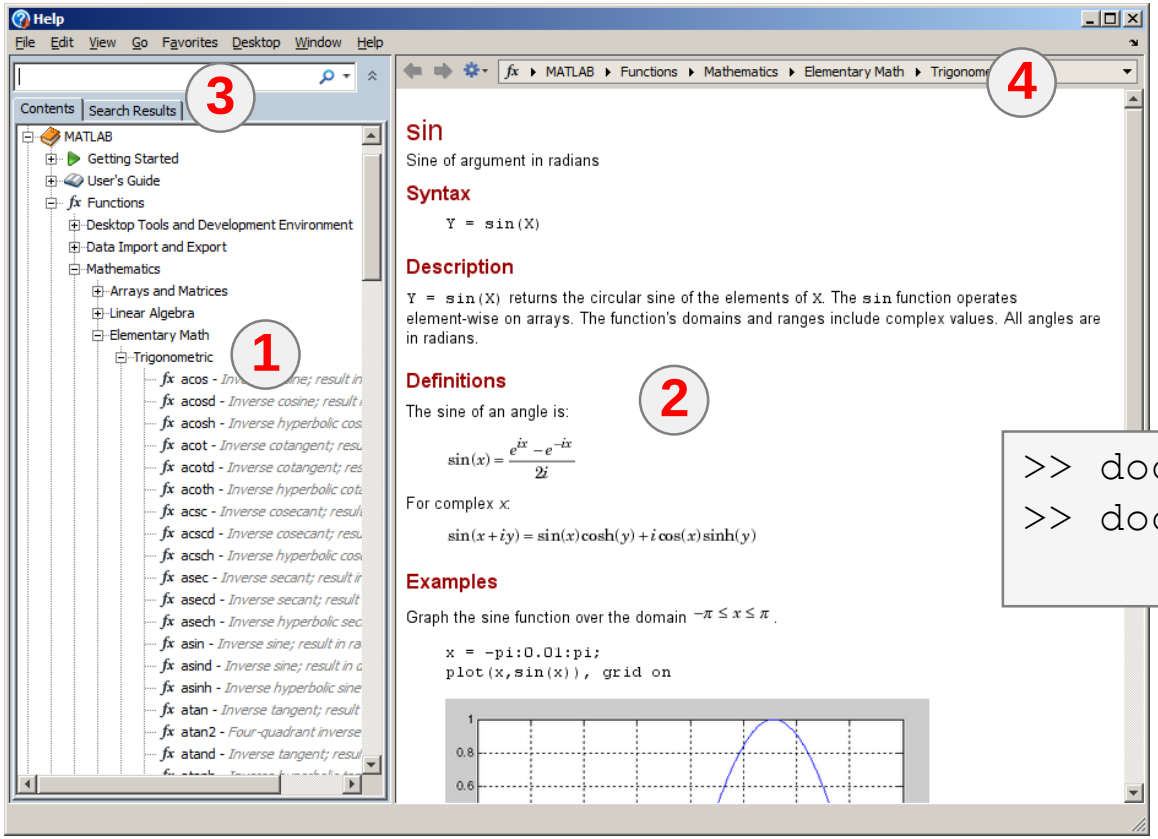


The Help structure

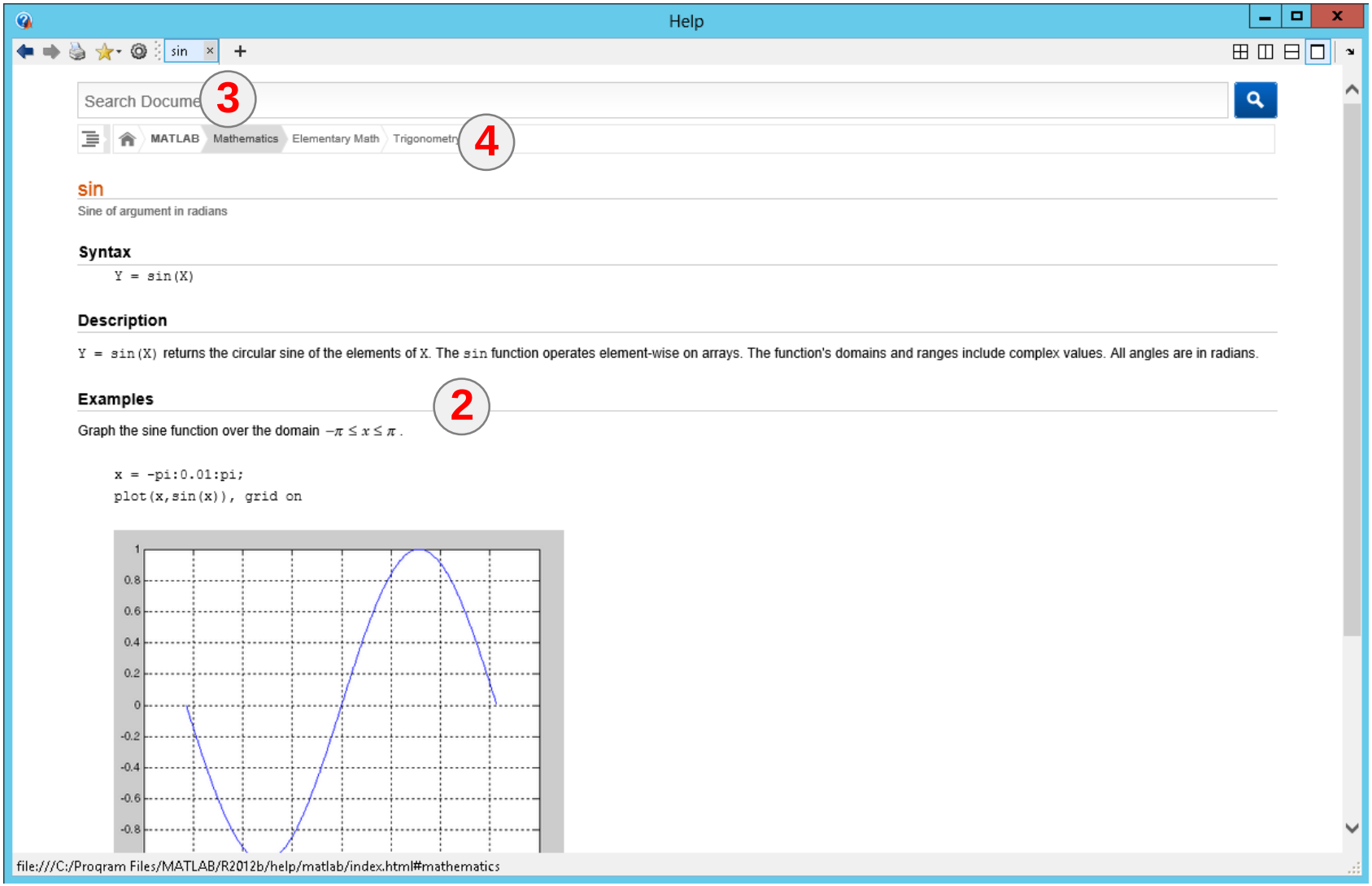
```
>> help      % displays basic help contents
>> help sin % displays help related to sine function
```

```
>> help sin
 SIN    Sine of argument in radians.
    SIN(X) is the sine of the elements of X.

    See also asin, sind.

    Overloaded methods:
       codistributed/sin

    Reference page in Help browser
       doc sin

>>
```




```
>> doc % launches help window
>> doc sin % sine function
           % related help
```

The Help structure, Matlab >R2011b



Matlab Help

240 s ↑

- start and terminate Matlab
- set the Matlab environment to your taste

- try to launch the help
- find the documentation of the following functions: `sin`, `cos`, `abs`
- browse through individual help chapters
  - pay attention to the part *Getting Started*

Shortcuts Command Window

key	meaning
ENTER	sends line for processing
ESC	deletes whole line
DEL	deletes one character (right to the cursor)
BACKSPACE	deletes one character (left to the cursor)
HOME	moves cursor to the beginning of line
END	moves cursor to the end of line
CTRL + ↑	moves cursor to the beginning of next word
CTRL + ↓	moves cursor to the beginning of previous word
SHIFT + ENTER	sends cursor to the next line
CTRL + K	deletes all to the right of cursor
CTRL + C	forces interruption of Matlab (e.g. long / erroneous calculation)
CTRL + TAB	switching between windows of Matlabu Environment
↓ a ↑	command history listing (searching is available CTRL+F)
F1	context help related to the word where the cursor is placed (Command Window, Editor)
TAB	function or variable name hint

+ usual Windows shortcuts for text processing

Searching the Help

key / command	meaning
SHIFT + F1	when pressed in command line, opens searchable function library
F9	evaluation of selected part of the code in Editor
NOT, OR, AND	it is possible to use logical operators in documentation search
*	it is possible to use wildcards in documentation search
""	to search exact phrase in documentation

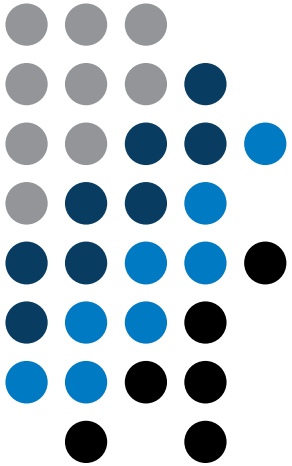
```
>> docsearch "plot tools"
```

```
>> docsearch plot* tools
```

Discussed functions

`quit, exit`	terminates Matlab	•
`preferences`	opens Matlab preferences	
`doc, help, demo`	commands related to documentation and help	•
`sin, cos`	sample goniometric functions	
`abs`	absolute value	

Thank you!



ver. 5.1 (19/02/2016)
Miloslav Čapek (C), Pavel Valtr (E)
miloslav.capek@fel.cvut.cz
Pavel.Valtr@fel.cvut.cz