

Getting Started

Contents

① MATLAB as a Calculator

② Variables

③ Scripting with MATLAB

MATLAB as a Calculator

Interfaces

- Command Window: interact with MATLAB
- Current Folder: directory view, navigation
- Workspace: list of saved variables

Arithmetic Operations

- Addition
- Subtraction
- Multiplication
- Division
- Exponentiation

Elementary Mathematical Functions and Predefined Variables

Math library.

- Exponential and logarithmic
- Trigonometric and inverse trigonometric
- and many more

Predefined variables.

- `pi`, `i`, `j`, `Inf`, etc.

Useful Commands

- Clearing screen: `clc`
- `format short/long`
- `format loose/compact`
- `format rat`
- `help`

Variables

Variables

- Predefined variables: `pi`, `i`, `j`, `eps`, `realmax`, `realmin`, `Inf`, `NaN`
- User-defined variables: use the equal symbol (`=`)

`<variable name> = <definition>`

Rules of naming.

- lowercase, uppercase, numbers, and underscore
- no spaces: `my var` \rightarrow `my_var`, `myVar`
- no number at the beginning: `2x` \rightarrow `twoX`, `x2`

Clear Variables

- `clear <VAR>`
- `clear <VAR1> <VAR2> ...`
- `clear`

Scripting with MATLAB

Displaying Text and Numbers

- Your first “Hello, World!” program:

```
disp('Hello, World!')
```

- To display text and number side by side:

```
disp(['the number is ', num2str(rand())])
```

Script M-File

Write multiple lines of MATLAB statements in a single file, called a script m-file. When asked to run the script, MATLAB executes all statements from top to bottom.

- Running a script: hitting “Run” button or calling a script by its name
- Commenting: % or %% at the beginning of a line
- Suppressing outputs: a semicolon(;) at the end of a statement
- Interactive program using `input` and `disp`:

```
var_name = input(' <PROMPT>' );
```

Example: Quadratic Equation Solver

Our first program as a script m-file:

```
% script m-file: quad eqn solver
a = input('the value of a: ');
b = input('the value of b: ');
c = input('the value of c: ');
D = b^2 - 4*a*c;
x1 = (-b + sqrt(D))/(2*a);
x2 = (-b - sqrt(D))/(2*a);
disp(['The first root: ', num2str(x1)])
disp(['The second root: ', num2str(x2)])
```