MATLAB Programming Language

Fall 2017

About This Course

Instructor / TA contact information:

- Office: EC709
- Office hour: 2DX, or by appointment
- Phone: 56689
- E-mail: wangts@cs.nctu.edu.tw
- Course web site: Use eCampus (e3.nctu.edu.tw)
- TAs: all my first-year graduate students
- TA office: EC031
- TA phone: 56681

Requirements / Grading

- [40%] Individual programming assignments (x5)
  - To be graded on both correctness AND performance.
- No written exams
- [50%] On-machine exams x2 (on Mondays)
- [10%] Mondays (starting in week 2) will be lab sessions.
  - If you miss or cannot complete some lab session for valid reasons, you can arrange with the TAs to do make-up demo within one week.

Textbook: None

Q&A

- What does "MATLAB" mean?
- Why learning MATLAB?
- Isn't MATLAB easy to learn? (Why taking a course?)
- What are MATLAB good for and what are not?

Backgrounds Needed

- Programming: You should know how to program in a high-level language. We will not spend much time on the basic concepts (such as Boolean logic, loops, functions, etc).
  - The ability of programming in C is needed for the last topic.
  - If you don't know C because you're not a CS major, I will provide alternative tasks for the respective programming assignment and exam problem.
- Some knowledge on linear algebra.

To Be Covered

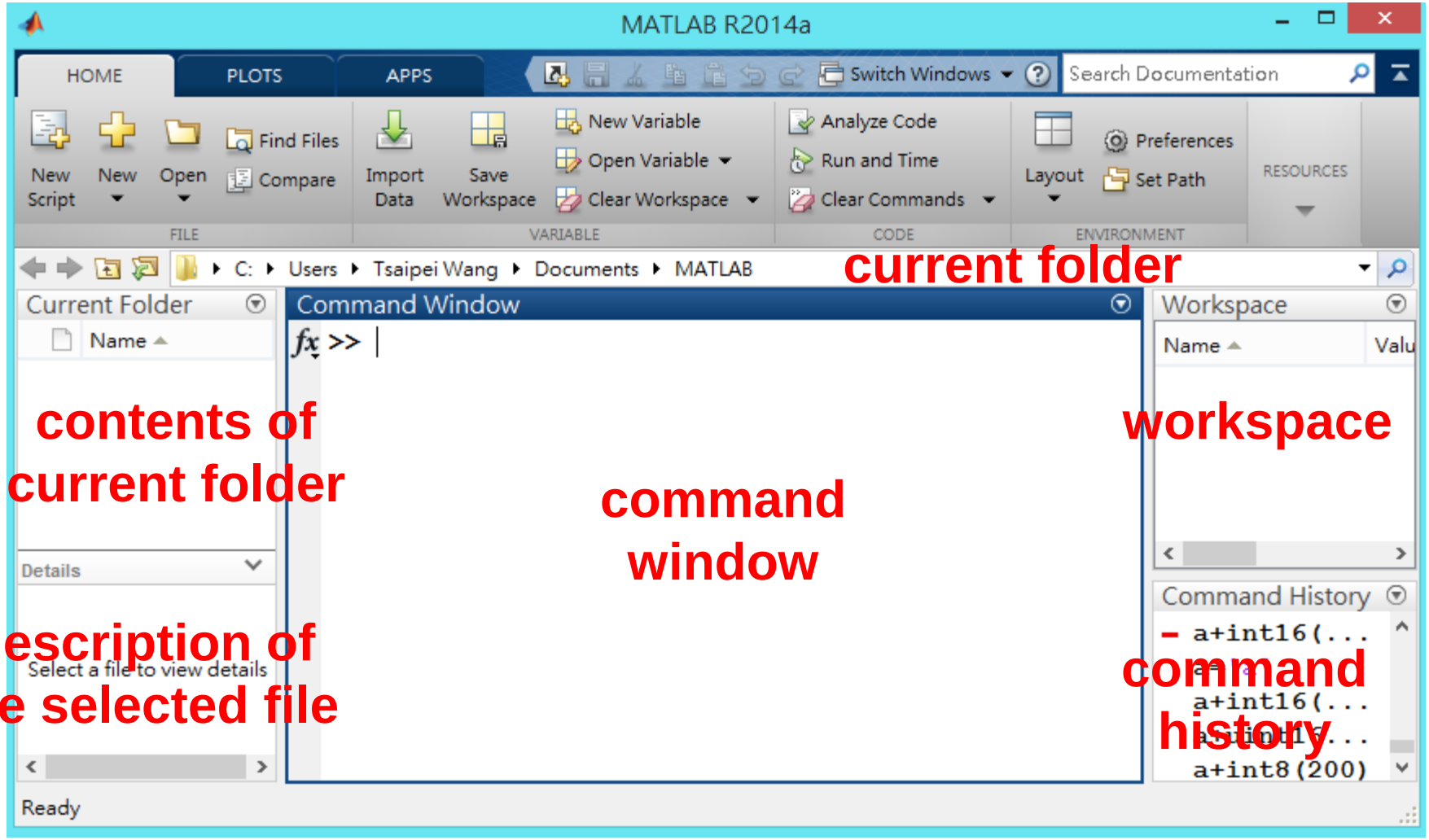
Beyond the basic aspects of programming, we will cover:

- Object-oriented programming in MATLAB
- User interaction and GUI programming (event-driven programming)
- Combining C and MATLAB both ways. (You need to be able to use Visual Studio 2012 or higher for C.)

Additional Notes

- The slides are not intended to be used as manuals. Many functions are listed so that you know they exist. To know how to use them, look into the MATLAB documentation.
- We will not cover the many MATLAB toolboxes, as most of them require specific domain knowledge to use.
- The lab sessions on Mondays may include some lectures related to the lab tasks.
- Class examples done directly on the MATLAB programming environment will be posted after the classes. However, no result will be given – only the command history. You need to try them yourself to see the results.
- Even with something as convenient as MATLAB, try to keep your programs organized, modularized, and documented.

Using MATLAB – A Quick Look



Using MATLAB – A Quick Look

- **Command Window**: Where you can type interactive commands and where the text outputs are displayed.
  - You can use this place to test codes very easily.
- **Current Folder**: This is the default path for loading/saving data or programs.
  - Don't use the default. Create one for each of your projects.
- **Workspace**: This holds all the variables used in the interactive (command-window) mode. (In other words, this is the "interactive" scope.)
- **Command History**: Previously typed commands. You can also use the UP/DOWN arrow keys to access the history items in the command window.

MATLAB File Extensions

- ***.m*** files: MATLAB source codes.
- ***.mat*** files: MATLAB data files.
  - In addition, MATLAB can also read/write regular text or binary files.
- ***.mex**** files: Binaries (similar to DLLs) of programs written in other languages and compiled with appropriate options so that they can be called within MATLAB.
- ***.fig*** files: Files that hold figures (including the associated data) so that they can be displayed directly. Also used for storing GUI layouts.

To-Do Now

If you have no previous MATLAB experience:

- Get a copy from the Computing Center. It is free for you now.
- Follow the online tutorial and get yourself familiar with both the interface and the language.