FF505

Computational Science

Introduction to the course

Marco Chiarandini (marco@imada.sdu.dk),
Claudio Pica (pica@cp3-origins.net)
Paolo Sibani (paolo.sibani@sdu.dk)

a joint course between IMADA and FKF
University of Southern Denmark

Outline

Outline

Aims of the course

- To learn to use the computer as a tool for scientific reasoning and discovery
- To learn how to solve numerically problems that can be solved analytically
- To equip you with tools you may need when analytical solutions are not known
- To learn to reason on the meaning of the results found

Organization of the Course

1. Introduction to a numerical computing environment, MATLAB
   (Marco Chiarandini) weeks 3-6
2. Introduction to mathematical tools, Linear Algebra
   (Claudio Pica) weeks 3-6
3. Applications in physics
   (Paolo Sibani) weeks 7-14
4. Training sessions
   (Jarno Markku Olavi Rantaharju and Philip Sørensen) weeks 3-14

Week	3	4	5	6	7	8	9	10	11	12	13	14
Tir, 09-12	I+TL [Ma] (Fælles) (IMADA terminalrum)		I+TL [Ma] (Fælles) (IMADA terminalrum)	I+TL [Ma] (Fælles) (IMADA terminalrum)	I [Pa] (Fælles) (IMADA terminalrum)						I [Pa] (Fælles) (IMADA terminalrum)	
Ons, 09-12						I [Pa] (Fælles) (IMADA terminalrum)		I [Pa] (Fælles) (IMADA terminalrum)				
Tor, 09-12	TL [Ja] (Fælles) (IMADA terminalrum)	I+TL [Cl] (Fælles) (IMADA terminalrum)										I [Pa] (Fælles) (IMADA terminalrum)
Tor, 14-17			I+TL [Ja] (Fælles) (IMADA terminalrum)	I+TL [Ja] (Fælles) (IMADA terminalrum)								
Fre, 10-13												TL [Ph] (Fælles) (IMADA terminalrum)
Fre, 13-16			TL [Cl] (Fælles) (IMADA terminalrum)	TL [Cl] (Fælles) (IMADA terminalrum)	TL [Ph] (Fælles) (IMADA terminalrum)	TL [Ph] (Fælles) (IMADA terminalrum)		TL [Ph] (Fælles) (IMADA terminalrum)			TL [Ph] (Fælles) (IMADA terminalrum)	

Contents of the course

Mathematical Tools:

- Matrices and vectors: matrix calculs
- Matrix inversion and determinants
- Eigenvalues and Eigenvectors
- Ordinary differential equations
- Coupled differential equations
- Lattice Laplacian
- Fourier analysis

MATLAB Section

The MATLAB Section will cover

- interactive environment
- vectorized operations
- programming: control structures, script, functions
- data input/output
- graphics

More specifically, it should prepare you to carry out the calculations needed in the other parts of the course and in the project.

Communication tools

- BlackBoard (BB)
  (link to MATLAB Section http://www.imada.sdu.dk/~marco/FF505)
- Announcements in BlackBoard
- Discussion Board in (BB) - allowed anonymous posting and rating
- Write to instructors and to Marco or Paolo
- Ask peers
- You are welcome to visit me in my office in working hours (8-16)

$\rightsquigarrow$ It is good to ask questions!!

$\rightsquigarrow$ Please, let us know **immediately** if you think we should do things differently!

Hands on Experience

The course assumes active participation to classes:

Introductory classes: you will be asked to perform small tasks at the computer.

Training sessions: weekly exercises; preparation + active participation

Study phase: you work with your study group

For the MATLAB part, slides and exercises are available at
http://www.imada.sdu.dk/~marco/FF505

You are also recommended to document your progress in a personal log-book, which contains your thoughts, calculations, figures etc.

Assessment

- Project in the last part of the course
  - similar to those you will do with Paolo
  - to be carried out in groups of two persons
  - hand in a written report, you should attempt to represent your results in a few plots and explain the meaning of what you see.
- Individual oral exam based on the written report
  - Graded with external censor according to 7-grade scale

Some Practical Stuff

Getting Matlab:

- machines in IMADA terminal room have Matlab R2014a installed (type `matlab` from command line)
- Install the SDU Site-License version (MATLAB R2015b) in your computer
  see link from MATLAB section page.
- use a Matlab clone, eg, Octave, SciLab
- use other software for similar purposes: eg, sage, python

Who is here?

31 in BlackBoard... how many are here?

Something about you...

- Which programme are you attending
  - First year physics
  - Applied mathematics
  - Guest student
- Previous experience with programming
- Experience with a computing environment?
- Expectations from this course?

The Unix Operating System

In Linux and Darwin (MacOsX). In Windows via CygWin:

- The command shell
- Commands: `ls`, `ls -l`, `cd`, `.`, `..`, `~`, `pwd`
- Manuals: `man ls`
- Commands: `cp`, `mv`, `rm`, `rm -r`
- displaying content: `less`, `more`
- searching content: `grep`
- access rights
- editors: `vi`, `emacs`, `gedit`, others like Sublime Text can be installed.
- plenty of useful command line programs:
  http://www.gnu.org/software/coreutils/manual/

Exercise:
Create a directory called `FF505` and a file named `hello.txt`. Edit the file and write something inside. Then try to access the file of one of your neighbors.

MATLAB

MATLAB (**ma**trix **la**boratory) is a high-level language and interactive environment to perform computationally intensive numerical computations faster than with low-level programming languages such as C, C++, and Fortran.

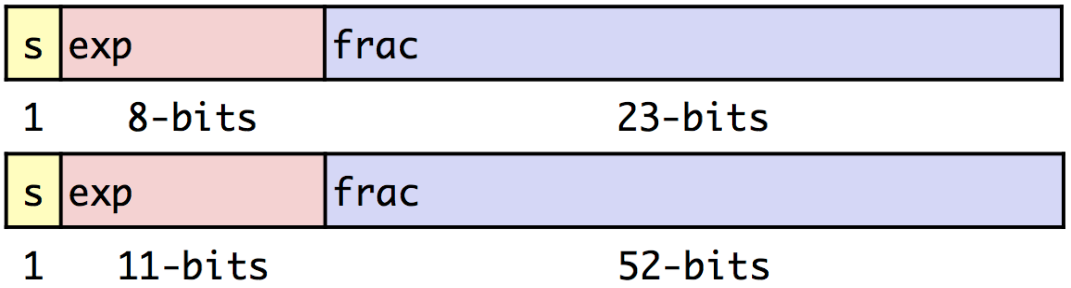
- Developed by a privately held company, MathWorks, 70% located at the company's headquarters in Massachusetts.
- Stable release: 2015b
- Written in C and Java
- License: Proprietary

Other similar numerical computing environments with high-level programming language are:

- Maple www.maplesoft.com (symbolic) – Proprietary
- Mathematica http://www.wolfram.com/mathematica (discrete mathematics) – [Proprietary]
- Octave www.gnu.org/software/octave – [General Public License]
- R www.r-project.org (statistics) – [GPL]
- Sage www.sagemath.org (discrete mathematics) – [GPL]
- SciPy www.scipy.org (based on python) – [GPL]
- ...
- later a comparison

Scientific vs Symbolic Computing

- scientific computing is based on numerical computation with approximate floating point numbers. $(-1)^s M 2^E$, $M \in [1, 2)$



http://www.mathworks.se/help/matlab/matlab_prog/floating-point-numbers.html

- symbolic computation manipulates mathematical expressions and other mathematical objects.
  emphasis on exact computation with expressions containing variables that have not any given value and are thus manipulated as symbols

  $\rightsquigarrow$ Try http://www.wolframalpha.com
  Symbolic computation can be done in MATLAB with the Symbolic Math Toolbox and the MuPAD editor (not installed)