

# Numerical Methods for CSE

## Exercise Session Week 40

Giuseppe Accaputo

October 3, 2016

## Today's Plan

1. General information regarding
  - 1.1 Exam
  - 1.2 Exercises
2. Debrief of exercises 1.3, 1.5 and 1.7

## Some Questions Regarding the Exam: IDE Usage

- ▶ CodeBlocks is available
- ▶ **BUT** use it at your own risk!
  - ▶ Lots of crashes during last year's exam
- ▶ **Suggestion:** Use gedit (or other editor of choice that is installed)
  - ▶ Get used to the command line (`cd`, `mkdir`, `cmake`, `make`)

## Some Questions Regarding the Exam: Plotting

- ▶ For plotting purposes you will learn how to work with the `Figure` class during the exercises
  - ▶ See chapter 0.3.4 in the lecture script for some examples
- ▶ Lecture script will be completely available as a searchable PDF during the exam
  - ▶ Try to work as much as possible with the lecture script during the exercises, so you know where to find most of the things, e.g. important equations, useful code snippets, and so on

## Simpler Workflow

- ▶ Working directly in the cloned repository NumCSE is cumbersome if you've never worked with git
  - ▶ A simple `git pull` will not work if there are some uncommitted local changes
- ▶ New workflow without having to learn all the git lingo
- ▶ Before we get to the new workflow do the following:
  - ▶ Install needed dependencies:

```
sudo apt-get install git cmake libpng++-  
↪ dev freeglut3-dev libboost-all-dev
```

- ▶ Backup all your own solution files up to this point, i.e. copy them to another folder outside of the NumCSE git folder
- ▶ Delete the NumCSE git folder
- ▶ Clone the NumCSE git folder anew

## New Workflow

1. `git pull` in the NumCSE folder to always get the current version of the source code
2. Copy the folder containing an assignment's source code (e.g. MatVec) to a folder outside of the NumCSE git folder
  - ▶ This is our new folder where we will work on our solutions
3. `cd` into the `templates_nolabel` folder of the exercise you want to solve
4. Run `mkdir build && cd build`
5. Run `cmake ..`
6. Run `make -j <nrcores>` where `<nrcores>` has to be replaced with the number of available processor cores

## New Exercises

- ▶ Problem 1.8 (**Core problem**)
- ▶ Problem 1.10
- ▶ Problem 1.11
- ▶ Problem 2.2 (**Core problem**)