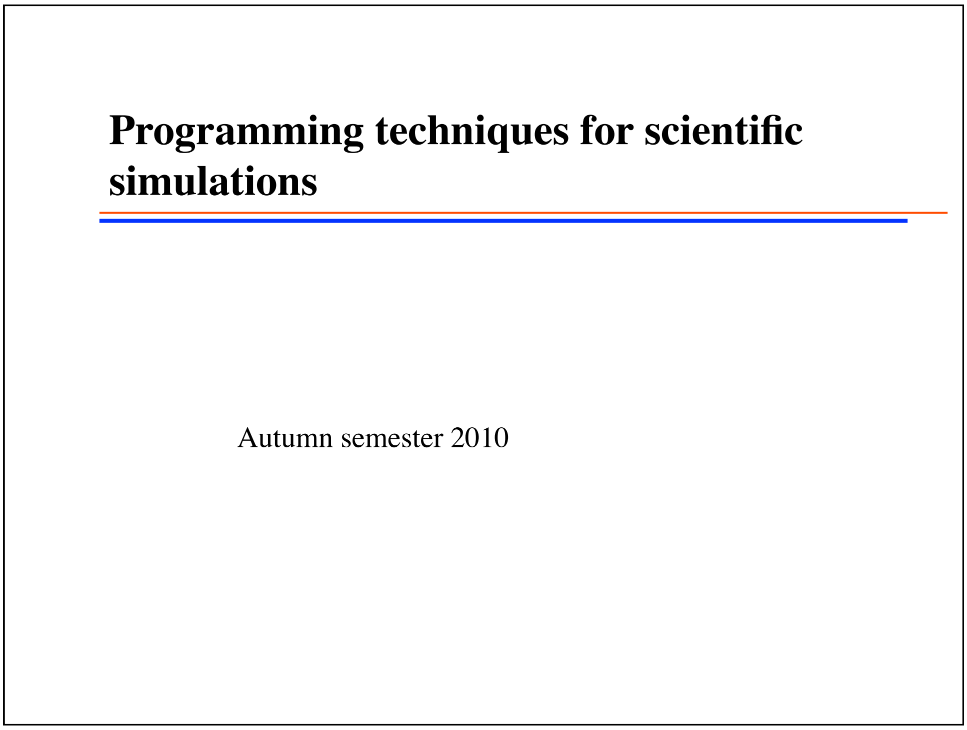
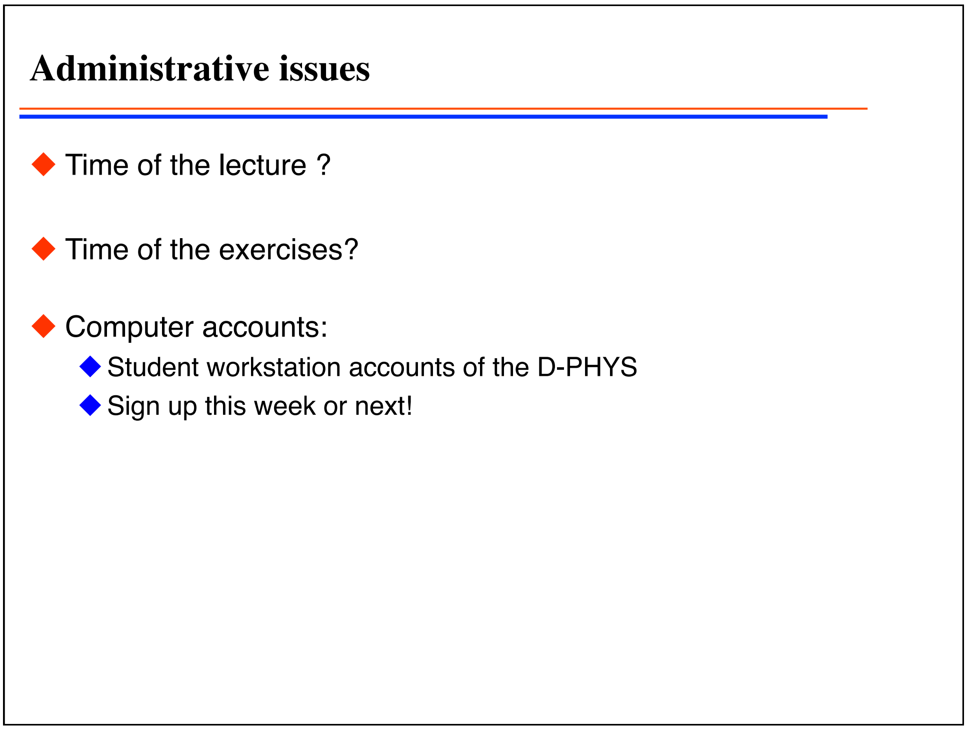


Programming techniques for scientific simulations

Autumn semester 2010

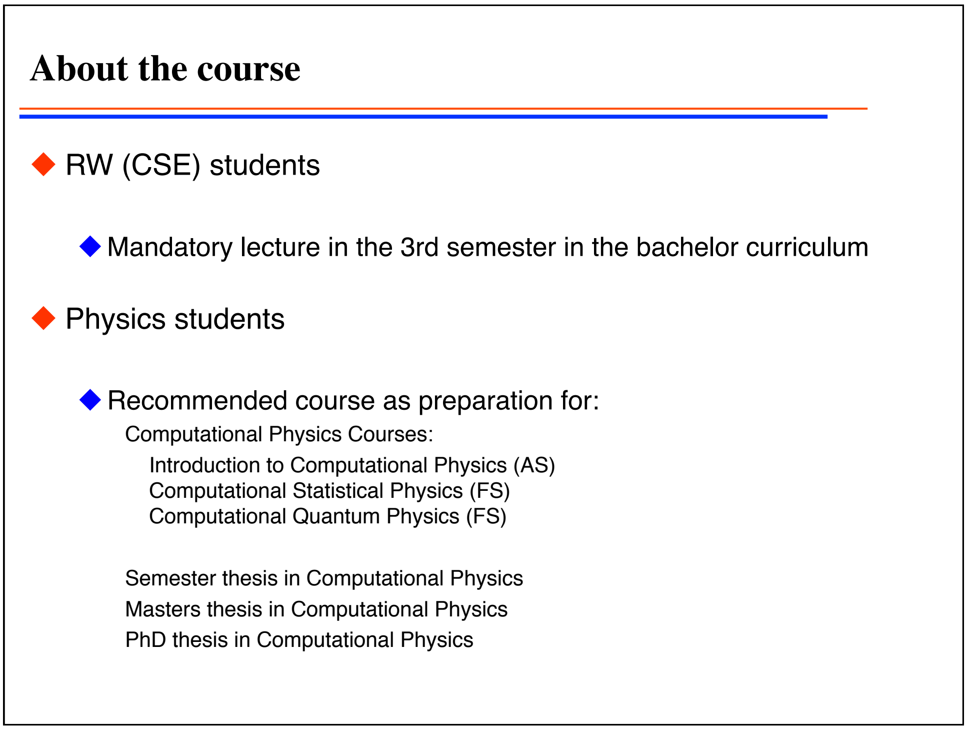
Information

- Prof. Dr. Matthias Troyer
  - Office: HIT K31.8
  - Tel.: 044/633 2589
  - E-Mail: troyer@phys.ethz.ch
- Exercises:
  - Tama Ma, pingnang@phys.ethz.ch
  - Jan Gukelberger, gukelberger@phys.ethz.ch



Administrative issues

- Time of the lecture ?
- Time of the exercises?
- Computer accounts:
  - Student workstation accounts of the D-PHYS
  - Sign up this week or next!



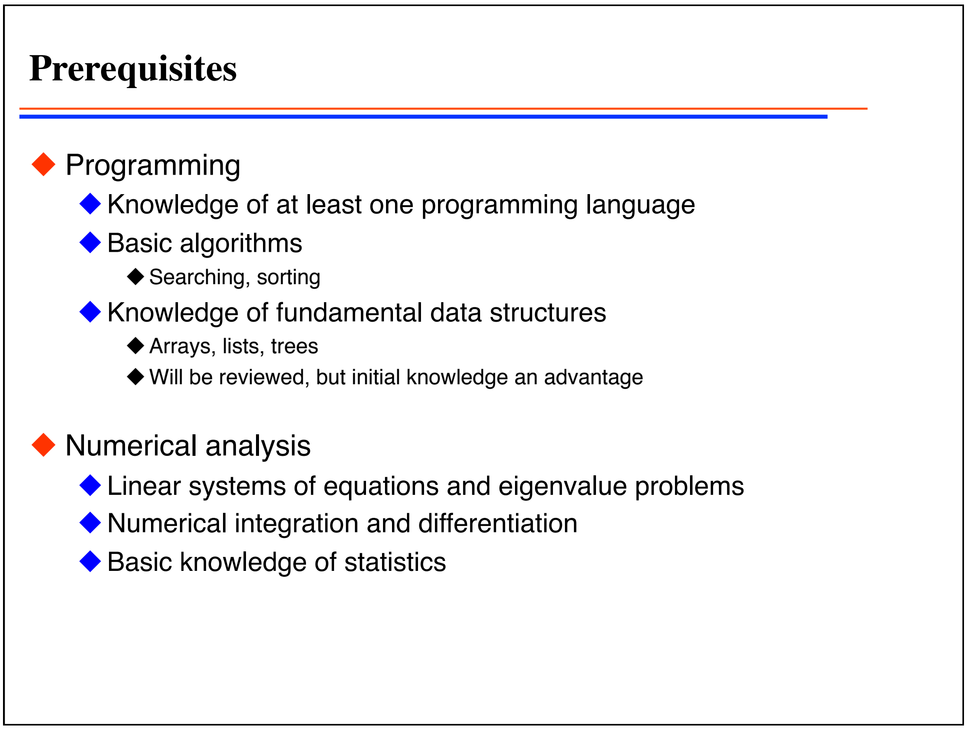
About the course

- RW (CSE) students
  - Mandatory lecture in the 3rd semester in the bachelor curriculum
- Physics students
  - Recommended course as preparation for:

    Computational Physics Courses:

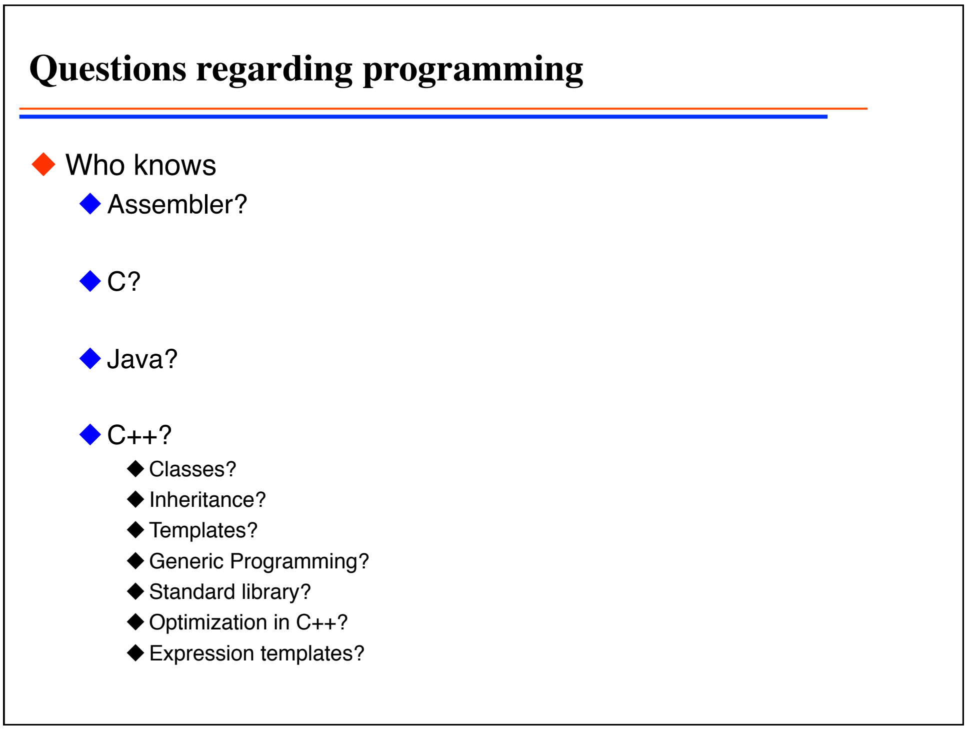
    Introduction to Computational Physics (AS)
    Computational Statistical Physics (FS)
    Computational Quantum Physics (FS)

    Semester thesis in Computational Physics
    Masters thesis in Computational Physics
    PhD thesis in Computational Physics



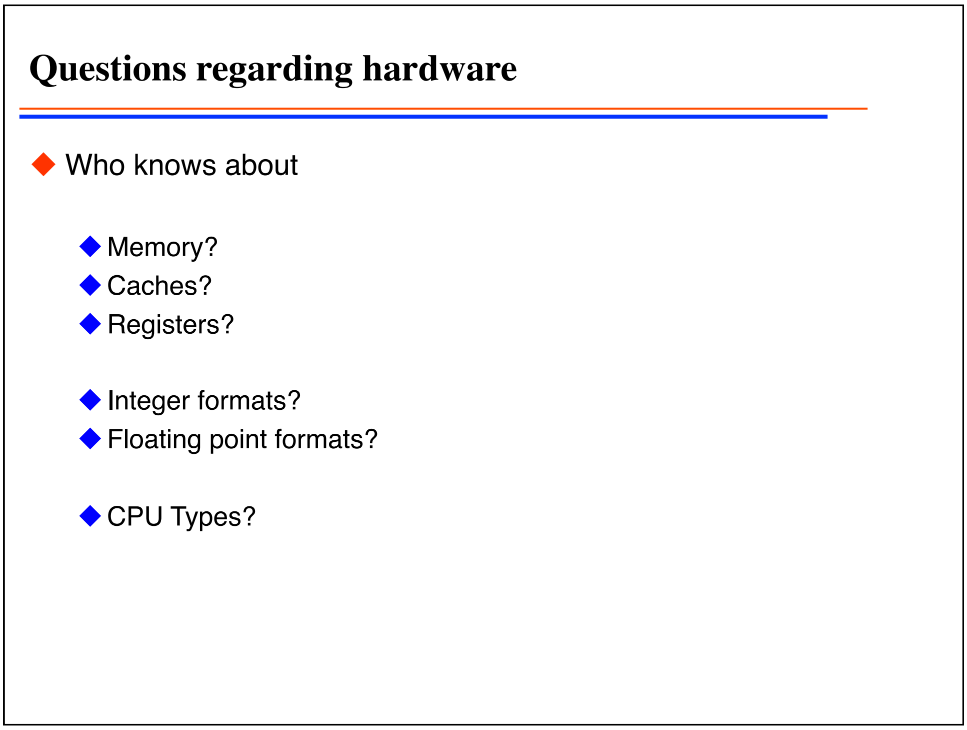
Prerequisites

- Programming
  - Knowledge of at least one programming language
  - Basic algorithms
    - Searching, sorting
  - Knowledge of fundamental data structures
    - Arrays, lists, trees
    - Will be reviewed, but initial knowledge an advantage
- Numerical analysis
  - Linear systems of equations and eigenvalue problems
  - Numerical integration and differentiation
  - Basic knowledge of statistics



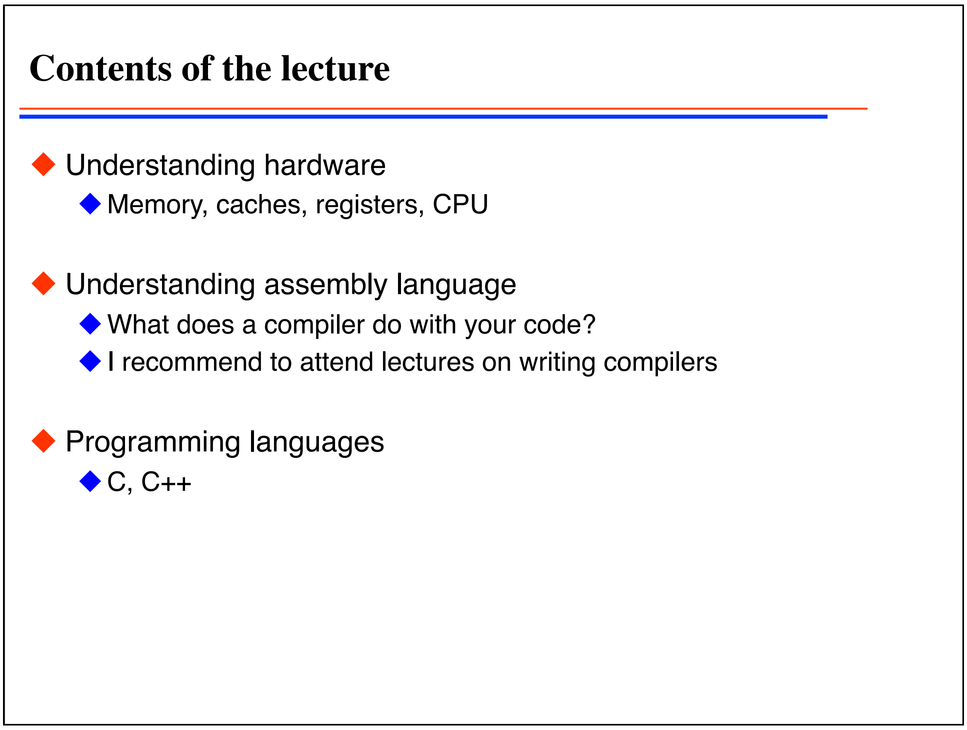
Questions regarding programming

- Who knows
  - Assembler?
  - C?
  - Java?
  - C++?
    - Classes?
    - Inheritance?
    - Templates?
    - Generic Programming?
    - Standard library?
    - Optimization in C++?
    - Expression templates?



Questions regarding hardware

- Who knows about
  - Memory?
  - Caches?
  - Registers?
  - Integer formats?
  - Floating point formats?
  - CPU Types?



Contents of the lecture

- Understanding hardware
  - Memory, caches, registers, CPU
- Understanding assembly language
  - What does a compiler do with your code?
  - I recommend to attend lectures on writing compilers
- Programming languages
  - C, C++

Contents of the lecture

- Abstractions for higher level programming
  - Object oriented programming and virtual functions
  - Generic programming and templates
- Libraries
  - High performance libraries
    - BLAS, ATLAS, LAPACK
  - C++ libraries
    - Standard library
    - Boost
  - Library design
    - Reusable components
    - Generic interfaces

Literature on C++ and optimization

- Andrew Koenig and Barbara E. Moo, *Accelerated C++,* Addison Wesley 2000
  - Good and short introduction
- Stanley B. Lippman, *Essential C++*, Addison Wesley 2000
  - Good and short introduction
- Bjarne Stroustrup, *The C++ Programming Language*, 3rd edition, Addison Wesley 1997
  - The reference book
- Todd Veldhuizen, *Techniques for Scientific C++*
  http://osl.iu.edu/~tveldhui/papers/techniques/techniques.ps