Introduction to Programming and Computing for Scientists (MNXB01 / NAFY018)

Introductory meeting

DIVISION OF PARTICLE PHYSICS (www.hep.lu.se)
teachers:

OXANA SMIRNOVA
CATERINA MARCON
FLORIDO PAGANELLI
BALAZS KONYA
ELSE LYTKEN

1. Learning outcomes

- Knowledge of computing and programming is **essential** in modern sciences
- Acquired knowledge and understanding:
  - Understanding and use of the concept of program libraries
  - Knowledge of processing, analysing and modelling scientific data using custom software
  - Basic understanding of code optimisation issues, implications of machine accuracy
  - Awareness of frameworks that use programmatic interface
- Acquired skills and abilities:
  - Ability to work in UNIX-based operating systems, particularly Linux
  - Ability to write computer programs using C++
  - Ability to compile from source, build and debug computing programs
  - Ability to develop and document program code in a collaborative environment

2. Required knowledge

- The course is aimed towards **beginners** who have little or no knowledge of computers
  - For those who already know programming some parts will still be interesting
- Some essential knowledge is still required:
  - Good knowledge of mathematics, including basic calculus
  - Basic knowledge of statistics is a bonus
  - Good knowledge of English

3. Course plan and logistics

- We are using the Canvas learning management platform, log in here:

https://canvas.education.lu.se

- The course uses Active Learning approach, with theory introductions followed by practical problem-solving classes and homework
  - Theory lectures every Wednesday in Sal F (K404) at 15:00
  - Problem-solving tutorials on Thursdays and Fridays, in the computer class Curie (K204)
    - Some tutorials will be in late afternoon, check the schedule!
  - Your *Stil-login* should work for all the computers and Canvas; if not, please get in touch with LU Servicedesk
- In total, 8 lectures and 16 tutorials
  - Presented by 5 teachers (best experts!)
  - Plus homework, which is mandatory
- It is important not to miss classes!

4. Assessment

Home assignments

- **50%** of the final grade
- 12 graded assignments and a questionnaire
- **Mandatory** to submit, even if incomplete
- Each assignment is graded on the scale from 0 to 5

Project

- **50%** of the final grade
- Done in teams of 3-4 students
- Task: use ROOT and C++ to write code that opens text files with data, performs analyses and plots the results
- Project reports must be handed in towards the end of the first half-term and presented by the teams (date to be defined)
- Projects are graded on the scale from 0 to 5; every team member receives the same grade, unless the contribution is outstanding either way

5. Course literature

- Main material: slides and references therein
  - Referred manuals are available on-line for free
- Official textbook: *“Problem solving with C++”, W. Savitch*, any edition
  - Covers ~50% of the course