HTTP://WWW.HEP.LU.SE/COURSES/MNXB01

Introduction to Programming and Computing for Scientists

Introductory meeting

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

OXANA SMIRNOVA
KATJA MANKINEN
FLORIDO PAGANELLI
BALAZS KONYA
PETER CHRISTIANSEN

Introductory meeting outline

1. Learning outcomes
2. Required knowledge
3. Course plan and schedule
4. Assessment (projects)
5. Course literature
6. Conclusion

1. Learning outcomes

- Knowledge of computing and programming is **essential** in physics, astronomy and other 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 schedule

- The course consists of theoretical lectures, practical tutorials and homework
  - Lectures every Wednesday (Sal F in September at 13:00 and K262 in October at 15:00)
  - Tutorials on Thursdays and Fridays, in the “big” computer class (H321)
  - Homework feed-back hours
  - **MAKE SURE** that you have **login** for the computers! (Check with the student administration – Yvonne)
- In total, 7 lectures and 14 tutorials
  - Presented by 5 teachers (best experts!)
  - Plus several homework feed-back sessions
- It is important not to miss the tutorials!
- Practical homework will complement the tutorials

4. Assessment

- Projects will be assigned in order to assess the knowledge at the end of the course
  - Will consist of programming tasks, similar to those performed during the course
  - A project is a team work of 2-3 students
- Students will have to hand in projects in November
- The projects will be evaluated and graded by the course teachers

5. Course literature

- Main material: hand-outs 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

6. Conclusion

- The course is still evolving, suggestions for improvements are welcomed!
- We brought our best experts to teach you
- The best result is achieved if you try what you learn at home!
  - Daring students are welcomed to use their notebooks for the tutorials