Scinetific calculation and Programming 1

Component
École Nationale Supérieure d'Électrotechnique d'Électronique

In brief

Code ELP: N5AM01B

Presentation

Objectives

It is an introductory course in computer science for scientific computing. The objectives are for students in terms of knowledge and know-how :

- Get a culture on the history of computer science
- Know the most common computer architectures
- Understand what is a system, a language, a terminal, a file
- Learn basic BASH shell commands (no script)
- Learn about algorithmics to solve scientific problems
- Learn the basic commands of a structured language (fortran)
- Know how to make a library, modules and compile a set of objects (Makefile)
- Be able to understand the error messages provided by a compiler
- Being able to translate a problem (from mathematics, engineering, a model) into a functional, efficient and reusable computer program.
- Being able to understand and modify sources of code written by a stranger
- Maintain a critical mind over numerical results obtained in a code
- Learn how to use basic tools for scientific computing (draw curves, write a report, use a powerful text editor)
- Lead a project as a team.

Description

Non-contractual information.
Last update on 18 December 2024

The teaching is presented by themes: history, introduction to computers, architectures, systems, languages and solving scientific problems (mathematics, structure, aerodynamics). The teaching consists of about 25% of courses, the rest being machine work and a long-term design office. Self-employment is also required.
All the course documents, or corrected exercises or practical work, as well as themed documents are available on Moodle to facilitate a remote work of the alternating students.
The teacher offers extremely varied exercises in all aspects of computer language: variables, table, structures, functions, input-output, modules and the structuring of a code (Makefile, library, modules).
It also shares many programming techniques to optimize the size of a code and make it modular, usable and editable by an unsuspecting user.
Awareness of free software and intellectual property (plagiarism) is also proposed.

The assessment is in the form of continuous monitoring, personal programs to be rendered, a test of knowledge on Moodle and the report (report + sources) of the Design Office.
A personal investment and a certain rigor are required to master at least the language and the techniques taught.

Non-contractual information.
Last update on 18 December 2024