Name: Programming and numerical methods - 34055

Type: Introductory **Semester:** 1st

ECTS: 7.5

Periodicity: annual

Departments involved: Department of Applied Physics (UPC)

Coordinator: Alvar Meseguer Serrano
Professors: Daniel Calvete Manrique, Fernando Mellibovsky Elstein, Alvaro Meseguer Serrano

Professors: Juan Sánchez Umbría

Language: English

Prerequisite: Calculus, Linear Algebra and Differential Equations.

Aims:

The aim of the subject is to provide the basic tools in numerical computation and programming in the languages FORTRAN 95 and C. To make the student being familiar with the systematic analysis of numerical algorithms along with their properties: stability, convergence, consistency. The goal is to make the student to understand the underlying mathematical theorems that build up numerical analysis, as well as its eventual implementation in high-level frameworks such as Matlab or their GNU equivalent Octave and/or SciLab.

Syllabus:

1. Introduction

- Floating point representation. Algorithms.
- Error propagation. Stability and instability of algorithms.
- Polynomials, numerical series and rational fractions.
- Introduction to languages FORTRAN-95 and Matlab. S

2. Interpolation and approximation of functions

- Interpolations of Lagrange, Newton and Hermite. Runge phenomenon, density of interpolation points and convergence.
- Approximation by least squares. Orthogonal polynomials.
- Trigonometric approximation. Fast Fourier Transform. Convolution products.

3. Zeros and minima of functions of a single variable

- Iterative methods. Fixed point methods, bisection, chord, Newton-Raphson and Brent.
- Minimization problems without derivatives.
- Golden rule and Brent's methods.
- Minimization problems with derivatives.

4. Differentiation and numerical integration

- Methods of indeterminate coefficients. Difference operators. Interpolatory differentiation.
- Newton-Cotes formulas. Trapezoidal and Simpson's rules. Euler-Maclaurin formula.

- Gaussian integration. Gauss-Legendre, Gauss-Chebyshev and Gauss-Lobatto methods.

5. Numerical integration of ordinary differential equations

- One step algorithms. Runge-Kutta methods.
- Boundary value problems. Shooting and finite difference methods.

Method and evaluation:

The theoretical lessons will consist in lecturers' presentations. There will be also problem solving classes and practical programming classes in a computer room.

The evaluation will be based on problem solving exercises:

1) Assignment on applied numerical methods (70% of final grade): Problem of scientific nature and of medium to high complexity where the student has to apply different numerical techniques within high-level frame

works such as Matlab, Scilab or Octave.

2) Assignment on Programming (30% of total grade): Problem of algorithmic nature where the student has to put in practice different programming skills in Fortran 90 or C.

Bibliography:

– Quarteroni, A., Sacco, R. and Saleri, F., Numerical Mathematics (2nd Ed.), Springer 2007.

– Dahlquist, G. and Bjorck, A., Numerical Methods, Dover 1974.

– Higham, D. J. and Higham, N. J., Matlab Guide, SIAM 2000.

– Glass, G., Ables, K., Unix for Programmers and Users, Prentice-Hall 1999.

– Chapman, S. J., Fortran 90/95 for scientists and engineers.