Academic subject: Mathematical Computing Laboratory

Degree Class: L-35 – Scienze Matematiche		**Degree Course:** Mathematics	**Academic Year:** 2018/2019	
		Kind of class: Mandatory	**Year:** 2	**Period:** 1
			ECTS: 7 divided into **ECTS lessons:** **ECTS exe/lab/tutor:**	

Time management, hours, in–class study hours, out–of–class study hours

lesson: 30 exe/lab/tutor: 40 in–class study: 70 out–of–class study: 105

Language:	**Compulsory Attendance:**	
Italian	no	

Subject Teachers:	**Phone / e-mail:**	**Office:** (Dept. of Mathematics)	**Office days and hours:** (Other days by appointment)
Pierluigi Amodio (primary instructor)	+39 080 5442703 pierluigi.amodio@uniba.it	Room 2, 4th Floor	Tuesday 13:00—14:00
Felice Iavernaro (primary instructor)	+39 080 5442703 felice.iavernaro@uniba.it	Room 2, 4th Floor	Monday 15:00—16:00
Lorenzo D’Ambrosio	+39 080 5442692 lorenzo.dambrosio@uniba.it	Room 16, 3rd Floor	Tuesday 12:00—13:00
Roberto Lascala	+39 080 5442674 roberto.lascala@uniba.it	Room 28, 2nd Floor	Monday 11:00—13:00
Francesco Bastianelli	+39 080 5442664 francesco.bastianelli@uniba.it	Room 18, 2nd Floor	Monday 16:30—18:30

Prerequisites:
The knowledge gained in the course “Computer Science”, classical analysis of one and several variables, fundamental linear algebra.

Educational objectives:
Acquiring some knowledge about the main properties and issues related to the use of finite arithmetic as opposed to real arithmetic. Acquiring the basic tools to operate in Matlab and Sage environments, with special attention to structured programming.

Expected learning outcomes (according to Dublin Descriptors)

Knowledge and understanding:
- Understanding and being able to explain issues related to the use of a computer for solving elementary mathematical problems.

Applying knowledge and understanding:
- Acquiring skills in programming, testing numerical algorithms and consistently interpreting computer results.

Making judgements:
- Being able to detect a proper programming strategy to solve elementary mathematical problems.

Communication:
- Being able to provide rigorous definitions and analysis of the principal aspects of finite arithmetic.
- Being able to communicate with computers ☺

Lifelong learning skills:

	➢ Capability of studying and solving, both numerically and symbolically, problems similar, but not necessarily equivalent, to those faced during the teaching activities.

Course program

1. **INTRODUCTION TO SCIENTIFIC COMPUTING AND ERROR ANALYSIS.** Mathematical models and numerical methods, errors sources, the process of the numerical approach to solve problems, computational environments, some languages for the scientific computing, problem solving environments: MATLAB, SAGE. Representing real numbers in a computer, IEEE standard, single and double precision. Truncating and rounding techniques. Absolute and relative errors. Machine precision. Floating-point operations. Errors propagations. Conditioning of a problem. Stability of an algorithm. Computational complexity.

2. **MATLAB.** Introduction to Matlab, the language, script and function files. Built-in functions in Matlab. The workspace. Introduction to graphics in one and two dimensions. Some Matlab examples about rounding errors. Handle vectors and matrices in Matlab. Basic operations involving vector and matrices. Implementation of some numerical algorithms in Matlab: approximation of derivatives of functions by means of divided difference formulae, using the Taylor polynomial to approximate transcendent functions, Laplace formula, Cramer rule and related computational cost. Examples on unstable algorithms.

3. **SAGE.**

- Programming basics, graphs of functions, derivation, integration, linear and nonlinear equations. Sequences, discrete dynamical systems, linear and nonlinear difference equations applied to biology, medicine, finance. Logistic equation and bifurcation diagram. Hints on the solution of differential equations. Iterated functions systems, fractals.

- Elementary operations in vector spaces. Generating random vectors and matrices. Orthogonal matrices, group actions, visualization of orbits generated by the standard action of the groups O(3) and O(2) on R^3. Linear transformations and solution of some classical problems in linear algebra. Explicit construction of affinities or isometries. Classification of planar isometries: examples of a procedure that splits an isometry in axial symmetries and of a procedure that classifies a given isometry. Examples on the computation of the group of symmetries of a finite set of points. Construction of projectivities. Visualization of the five non-degenerate quadrics of the Euclidean space.

- Euclidean algorithm, extended Euclidean algorithm, Bezout coefficients, algorithms to find the first n prime numbers, algorithms for the prime factorization of a natural number. RSA Cryptography.

Teaching methods:
Lectures and exercise sessions. Exercise sessions in the Computer Centre,

Auxiliary teaching:
Handouts, notes and Matlab codes will be made available on the net.

Assessment methods:
The exam consists in solving two exercises, in Matlab and Sage, on a computer and an oral test which includes a discussion of the Matlab codes prepared during the course lectures.

Bibliography:

- Handout on machine arithmetic, available at the url http://www.dm.uniba. it/~iavernaro/studenti.htm
- "Introduzione al Matlab", available at the url http://www.dm.uniba.it/~iavernaro/ studenti.htm
- Handouts supplied during the course lectures.
- Uri M. Ascher and Chen Greif, A First Course on Numerical Methods, SIAM, 2011.