

Testi del Syllabus

Resp. Did.	LIESSI DAVIDE	Matricola: 007587
Docenti	BREDA DIMITRI, 0 CFU LIESSI DAVIDE, 6 CFU	
Anno offerta:	2020/2021	
Insegnamento:	MA0428 - LABORATORIO DI MATEMATICA COMPUTAZIONALE	
Corso di studio:	767 - MATEMATICA	
Anno regolamento:	2020	
CFU:	6	
Settore:	MAT/08	
Tipo Attività:	B - Caratterizzante	
Anno corso:	1	
Periodo:	Secondo Periodo	



Testi in italiano

Contenuti

Dopo una generale introduzione sulla matematica computazionale, sui numeri di macchina e sull'analisi degli errori che si possono generare nel percorso dalla modellizzazione di un problema matematico alla sua risoluzione al calcolatore, si prevede una suddivisione in singole unità distinte per tematica, ciascuna comprendente una parte di introduzione teorica all'argomento trattato ed alle sue implicazioni attuali, seguita da una parte di sviluppo, implementazione e testing al calcolatore. Per ogni unità si prevede l'utilizzo di software e ambienti matematici anche diversi (ad esempio GeoGebra, Mathematica, MATLAB), come strumenti sia di calcolo numerico e simbolico, sia di visualizzazione grafica e gestione dati. Tra le potenziali tematiche si includono:

- equazioni nonlineari e metodo di Newton (proposta applicativa: i frattali di Newton);
- matrici e sistemi lineari (proposta applicativa: il PageRank di Google);
- equazioni differenziali e sistemi dinamici (proposta applicativa: analisi di biforcazione e caos);
- approssimazione di dati e funzioni (proposta applicativa: FFT e compressione JPEG).

Gli argomenti trattati forniranno dunque un pretesto per affrontare alcune delle problematiche che interessano l'interazione con il calcolatore e le sue potenzialità e/o limitazioni nello studio di problemi matematici: rappresentazione di oggetti matematici e loro manipolazione, risoluzione di problemi e simulazione di modelli, gestione di dati.

Testi di riferimento

Non ci sono fonti univoche per l'intero programma, riferimenti specifici saranno consigliati durante il corso a seconda della tematica trattata. Vengono fornite le dispense del corso scritte dal docente in lingua Inglese, complete di attività di laboratorio, relativi codici e bibliografia di riferimento rispetto ai diversi argomenti trattati, rese disponibili sulla pagina e-learning del corso previa iscrizione (<https://elearning.uniud.it/moodle/>).

Obiettivi formativi

Il corso vuole fornire allo studente un'adeguata familiarità nell'utilizzo del calcolatore come efficace ausilio allo studio teorico della matematica ed alle attività didattiche, applicative e di ricerca ad essa connesse. Si propone quindi di risolvere sperimentalmente alcuni problemi matematici che nascono in diversi contesti applicativi, accompagnando lo studente

dal modello allo sviluppo di codici.

Lo studente dovrà:

Conoscenza e comprensione:

- conoscere gli aspetti base della mutua interazione tra matematica e calcolatore
- comprendere la classe di problemi matematici a cui ascrivere il proprio modello
- apprendere le linee guida per tradurre il problema matematico in un problema trattabile al calcolatore

Capacità di applicare conoscenza e comprensione:

- sviluppare una capacità di auto-apprendimento di software matematici in genere
- saper selezionare il software matematico meglio adatto alla risoluzione del problema e programmare i relativi codici per la risoluzione stessa

Autonomia di giudizio:

- essere in grado di analizzare in maniera autonoma e critica i risultati del calcolatore in relazione a quelli attesi dalla teoria

Abilità comunicative:

- saper spiegare i processi computazionali in modo chiaro e comprensibile
- saper rappresentare i risultati computazionali in maniera efficace

Capacità di apprendimento:

- saper affrontare in modo autonomo e critico problemi matematici con tecniche computazionali

Prerequisiti

Preferibile una preparazione base di matematica:

- basi di analisi: numeri reali e complessi, funzioni, limiti, derivate, integrali
- basi di algebra: vettori, matrici, sistemi lineari
- basi di analisi numerica: metodi per sistemi lineari, metodo di Newton, interpolazione.

Le modalità di presentazione degli argomenti e la struttura stessa del corso permettono la frequenza e l'apprendimento anche a chi risultasse privo di alcuni dei requisiti di cui sopra.

Metodi didattici

- lezioni teoriche frontali
- laboratori ed esercitazioni di apprendimento ed utilizzo di software matematici
- possibili brevi seminari su argomenti specifici.

Altre informazioni

- i crediti del corso sono riconosciuti per il Percorso Formativo 24 CFU - DM 616/2017 (A-28 "Matematica e Scienze")
- lingua di insegnamento: italiano (l'insegnamento può essere tenuto in lingua inglese, su proposta della struttura didattica competente)
- gli argomenti proposti per il progetto possono essere legati alle attività concernenti la prova finale di tesi per quanto riguarda gli eventuali aspetti computazionali
- le dispense del corso, scritte dal docente in lingua inglese, risultano complete ed auto-contenute rispetto al programma, comprensive anche delle attività di laboratorio previste e dei relativi codici.

Modalità di verifica dell'apprendimento

L'esame consiste in:

- realizzazione di un progetto di laboratorio di carattere computazionale, su argomento da concordare con il docente, accompagnata da una breve relazione scritta completa di una breve descrizione teorica del problema, degli obiettivi del progetto e di quanto fatto ed ottenuto;
- discussione orale sulle attività svolte e sul programma del corso, con domande di verifica sia teoriche che applicative ed esercizi su problemi di natura computazionale.



Testi in inglese

Contents

After a general introduction on computational mathematics, machine numbers and analysis of the errors that can be generated in the path from the modelling of the mathematical problem to the computation of its solution, the plan is to divide the course in separate units, distinguished by argument, each of which includes a part of theoretical introduction to the treated subject and to its present implications,

followed by a part of development, implementation and testing on a computing system. For every unit, the use of different mathematical software and environments (e.g., GeoGebra, Mathematica, MATLAB) is programmed, as tools for both numerical and symbolic computation, and graphical visualization and data management. Among the potential themes are:

- nonlinear equations and Newton method (application proposal: Newton fractals);
- matrices and linear systems (application proposal: Google's PageRank);
- differential equations and dynamical systems (application proposal: bifurcation analysis and chaos);
- approximation of data and functions (application proposal: FFT and JPEG compression).

The treated arguments will thus furnish an occasion to tackle some of the issues concerning the interaction with a computing system and its potentialities/limitations with respect to the study of mathematical problems: representation of mathematical objects and their manipulation, problem solving and simulation of models, data management.

Texts

Materials given by the lecturer. References will be given during the course depending on the specific argument. Course notes will be given, written by the lecturer in English, including laboratory activities, relevant codes and specific references with respect to the treated arguments, made available on the e-learning webpage of the course subject to registration (<https://elearning.uniud.it/moodle/>).

Objectives

The course aims at getting the student adequately used to the utilization of computing facilities as effective and helpful tools towards the theoretical study of mathematics and of its related teaching, application and research activities. It therefore consists in solving experimentally mathematical problems arising in diverse application contexts, driving the student from modelling to coding.

The student will have to:

Knowledge and comprehension:

- know the basic aspects of the mutual interaction between mathematics and computer
- understand the class of mathematical problems in which the model resides
- learn the guidelines to translate the mathematical problem into a computable one

Capacity of applying knowledge and comprehension:

- develop skills in self-learning general mathematical software
- know how to select the mathematical software best suited to the solution of the problem and to program the relevant codes for obtaining the solution itself

Autonomy of judgement:

- be able to analyze in a critical and autonomous manner computer results in relation with theoretical expectation

Communication skills:

- know how to illustrate the computational processes in a clear and comprehensible fashion
- know how to represent effectively the computational results

Learning skills:

- know how to tackle critically and autonomously mathematical problems with computational techniques

Prerequisites

Advisable a basic preparation in Mathematics:

- analysis: real and complex numbers, functions, limits, derivatives, integrals
- algebra: vectors, matrices, linear systems
- numerical analysis: methods for linear systems, Newton method, interpolation.

The presentation modalities and the course structure itself allow attendance and learning also for those who possibly do not possess some of the above requisites.

Teaching Methods

- theoretical lectures
- laboratory activities and exercises on learning and using mathematical software

- possible seminars on specific arguments.

More Information

- the credits are valid for Percorso Formativo 24 CFU - DM 616/2017 (A-28 "Matematica e Scienze")
- teaching language: Italian (the course can be taught in English on proposal of the competent didactic structure)
- the arguments proposed for the project can be related to the final thesis for what concerns the possible computational aspects
- the course notes, written by the lecturer in English, are complete and self-contained relatively to the program, and include laboratory activities and relevant codes.

Verification of learning

The exam consists in:
- laboratory project of computational character, on a subject to be arranged with the lecturer, accompanied by a brief written essay including a brief theoretical description of the problem, of the targets of the project and of what done and obtained;
- oral discussion on the developed activities and on the course program, with questions on both theory and applications, and exercises on computational problems.