

Insegnamento di: Calcolo Numerico			
Classe di laurea: L-31 – Scienze e Tecnologie Informatiche		Corso di Laurea in: Informatica	
		Anno accademico: 2019/2020	
Denominazione inglese insegnamento: Numerical Calculus		Tipo di insegnamento: Obbligatorio	
		Anno: 2	Semestre: 1
Tipo attività formativa: a - Attività di base	Ambito disciplinare: Formazione Matematico- Fisica	Settore scientifico-disciplinare: MAT/08	CFU totali: 6 di cui CFU lezioni: 4 CFU ese/lab/tutor: 2
Modalità di erogazione, ore di didattica assistita ed ore dedicate allo studio individuale ore di lezione: 32 ore di esercitazione/laboratorio/tutorato: 30 totale ore didattica assistita: 62 totale ore di studio individuale: 88			
Lingua di erogazione: Italiano	Obbligo di frequenza: no		
Docente: Alessandro Pugliese	Tel: +39 080 5442689 e-mail: alessandro.pugliese@uniba.it	Ricevimento studenti: Dip. Matematica, Stanza 12, III piano	Giorni e ore ricevimento: Martedì 15:00-16:00
Conoscenze preliminari: Analisi matematica classica in una variabile, elementi di algebra lineare.			
Obiettivi formativi: Apprendimento di alcuni metodi classici dell'analisi numerica e acquisizione delle competenze necessarie ad affrontare la risoluzione di problemi matematici per mezzo del computer.			
Risultati di apprendimento previsti	Conoscenza e capacità di comprensione: Il corso presenta alcune tecniche dell'analisi numerica classica, con attenzione sia all'analisi teorica degli algoritmi che alla loro implementazione al computer in linguaggio Python.		
	Conoscenza e capacità di comprensione applicate: Il corso consente allo studente di sviluppare la capacità di risolvere numericamente problemi matematici, sia di interesse teorico che pratico, per mezzo del computer.		
	Autonomia di giudizio: Il corso contribuisce ad aumentare le abilità dello studente nel ragionamento logico deduttivo, nel problem solving e nella programmazione in ambito scientifico. Inoltre il corso insegna allo studente come validare o confutare una tesi mediante esperimenti numerici, prima di passare, eventualmente, ad argomentazioni rigorose.		
	Abilità comunicative: Lo studente è incentivato a consultare libri di testo classici del settore proposti dal docente, in lingua sia italiana che inglese. Inoltre l'esame consta di una combinazione di: esposizione orale sui temi del corso, prova di laboratorio in ambiente Python, prove scritte di valutazione in itinere. Ciò aiuta a migliorare le sue abilità comunicative.		
Capacità di apprendere: Il corso fornisce allo studente abilità che gli saranno utili nel proseguo dei suoi studi (nelle discipline di ambito numerico/applicativo, e non solo) e nella sua futura attività lavorativa.			
Programma del corso			
<ol style="list-style-type: none"> Metodi numerici per il calcolo degli zeri di funzione: condizionamento degli zeri di funzione, metodo delle successive bisezioni, metodo Regula Falsi, ordine di convergenza, polinomio di Taylor (richiamo), metodo di Newton ed alcune sue varianti (corde, secanti, quasi-Newton), successioni definite per ricorrenza e teoria generale dei metodi iterativi con applicazioni al calcolo degli zeri di funzione. Aritmetica di macchina ed analisi dell'errore: sorgenti di errore, rappresentazione di un numero reale in una base arbitraria, insieme dei numeri di macchina, standard IEEE 754 a singola e doppia precisione, operazione di arrotondamento, errori assoluto e relativo, errore relativo commesso nell'arrotondamento, condizionamento di aritmetica di macchina e valutazione di funzione. Elementi di algebra lineare, algebra lineare numerica: richiami di algebra lineare (matrici e vettori, operazioni 			

algebriche tra matrici, determinante e regola di Laplace, regola di Cramer, sistemi di equazioni lineari), metodo di eliminazione di Gauss, fattorizzazione LU, fattorizzazione LU con pivoting parziale, norme vettoriali e matriciali, condizionamento della risoluzione di un sistema lineare, forma “a gradini” di una matrice, rango di una matrice e Teorema di Rouché-Capelli, risoluzione di sistemi lineari sotto-determinati e sovra-determinati, metodo dei minimi quadrati, regressione lineare e polinomiale, autovalori ed autovettori (cenni), metodo delle potenze (cenni), algoritmo *Google PageRank*.

4. **Interpolazione polinomiale:** polinomio d'interpolazione, polinomi caratteristici di Lagrange, formule dell'errore con nodi arbitrari e nodi equidistanti, nodi di Chebyshev-Gauss, interpolazione lineare a tratti.
5. **Spazi vettoriali:** corpi e spazi vettoriali, sottospazi vettoriali, combinazione lineare degli elementi di uno spazio vettoriale, generatori di uno spazio vettoriale, elementi linearmente indipendenti, base di uno spazio vettoriale, dimensione di uno spazio vettoriale. Prodotto scalare, angolo tra vettori e perpendicolarità. Nucleo e immagine di un'applicazione lineare tra spazi vettoriali.
6. **Elementi di programmazione in ambiente Python:** tipi di dati, variabili, operazioni algebriche e funzioni elementari, variabili predefinite, operatori logici e di relazione, istruzione *if-elif-else*, cicli *for* e *while*, creazione di script e funzioni Python, implementazione dei seguenti algoritmi: metodo delle successive bisezioni, metodo di Newton, interpolazione mediante polinomi di Lagrange, calcolo del determinante di una matrice quadrata mediante la regola di Laplace, fattorizzazione LU con e senza pivoting parziale.

Metodi di insegnamento:

Lezioni ed esercitazioni in aula.

Supporti alla didattica:

Algoritmi ed esercizi distribuiti in modalità elettronica.

Controllo dell'apprendimento e modalità d'esame:

Prove scritte in itinere, prova di laboratorio, prova orale.

Testi di riferimento principali:

1. Atkinson K.E., *An introduction to Numerical Analysis - 2nd Ed.*. John Wiley & Sons
2. Quarteroni A., Saleri F., Gervasio P.. *Calcolo Scientifico - Esercizi e problemi risolti con MATLAB e Octave - 5a edizione*. Springer Italia.