

Principali informazioni sull'insegnamento	
Titolo insegnamento	Calcolo Numerico
Corso di studio	Informatica e Tecnologie per la Produzione del Software
Crediti formativi	4 (lezioni frontali) 2 (esercitazioni)
Denominazione inglese	Numerical Calculus
Obbligo di frequenza	No
Lingua di erogazione	Italiano

Docente responsabile	Nome Cognome	Indirizzo Mail
	Francesca Mazzia	francesca.mazzia@uniba.it

Dettaglio credi formativi	Ambito disciplinare	SSD	Crediti
	Matematico	MAT09	6

Modalità di erogazione	
Periodo di erogazione	primo semestre
Anno di corso	secondo
Modalità di erogazione	Lezioni frontali Esercitazioni/Laboratorio in aula

Organizzazione della didattica	
Ore totali	150
Ore di corso	62 (32: lezioni frontali, 30: esercitazioni/laboratorio)
Ore di studio individuale	88

Calendario	
Inizio attività didattiche	06/10/2020
Fine attività didattiche	13/01/2021

Syllabus	
Prerequisiti	Elementi di base di algebra lineare; tutti gli argomenti trattati nei corsi di analisi matematica; elementi di programmazione.
Risultati di apprendimento previsti (declinare rispetto ai Descrittori di Dublino) (si raccomanda che siano coerenti con i risultati di apprendimento del CdS, compreso i risultati di apprendimento trasversali)	<ul style="list-style-type: none"> • <i>Conoscenza e capacità di comprensione</i> <ul style="list-style-type: none"> ❖ Conoscere le tecniche e i metodi per la programmazione numerica finalizzati alla risoluzione di problemi nell'ambito delle discipline matematiche ed affini, con particolare enfasi ai problemi fondamentali nell'ambito dell'algebra lineare. ❖ Comprendere e saper illustrare le problematiche relative dell'uso del calcolatore per la risoluzione di problemi matematici. • <i>Conoscenza e capacità di comprensione applicate</i> <ul style="list-style-type: none"> ❖ Capacità di risolvere problemi matematici mediante algoritmi ottimizzati dal punto di vista del costo computazionale e della stabilità. ❖ Sviluppo delle capacità di programmare, documentare e testare algoritmi numerici, interpretandone correttamente i risultati. ❖ Sviluppo delle capacità di risolvere problemi matematici usando problem solving environments.

	<ul style="list-style-type: none"> • Autonomia di giudizio <ul style="list-style-type: none"> ➤ Saper individuare il metodo numerico più idoneo per risolvere numericamente un problema matematico tra quelli trattati nel corso. • Abilità comunicative <ul style="list-style-type: none"> ➤ Saper definire in modo rigoroso i problemi matematici trattati nel corso e saper esporre i relativi metodi numerici, delineandone le proprietà fondamentali. • Capacità di apprendere <ul style="list-style-type: none"> ➤ Capacità di studiare e risolvere problemi numerici simili ma non necessariamente uguali a quelli affrontati durante le lezioni.
<p>Contenuti di insegnamento</p>	<p>Il corso si propone come raccordo costruttivo fra la matematica e l'informatica, fornendo allo studente gli strumenti specifici di base per risolvere i problemi matematici usando il calcolatore mettendo in evidenza i rischi che si corrono con un uso ingenuo delle risorse di calcolo. Il corso presenta i metodi fondamentali per risolvere numericamente problemi matematici, mettendo in evidenza gli aspetti computazionali. Il programma comprende i metodi iterativi per equazioni non lineari, i metodi diretti per sistemi lineari, interpolazione, approssimazione, integrazione e calcolo di autovalori.</p> <p>1. Introduzione al Calcolo Scientifico Modelli matematici e metodi numerici, sorgenti di errori, il processo di risoluzione numerica, efficienza, testing, errori computazionali, ambienti computazionali, linguaggi per il calcolo scientifico, problem solving environments: MATLAB, SCILAB, R, PYTHON, SAGE.</p> <p>2. Analisi dell'errore. Rappresentazione dei numeri. IEEE singola e doppia precisione. Troncamento e Arrotondamento. Precisione di macchina. Errore assoluto e relativo. Operazioni con i numeri di macchina. Cancellazione di cifre significative. Condizionamento di un problema. Stabilità degli algoritmi. Propagazione degli errori. Introduzione a Python, il linguaggio, file di tipo script e function. Funzioni predefinite in NumPy. Introduzione alla grafica. Esempi Python sugli errori di arrotondamento.</p> <p>3. Elementi di algebra lineare Matrici e vettori. Operazioni con matrici. Inversa di una matrice. Sistemi lineari. Prodotto esterno fra vettori. Dipendenza lineare di vettori. Autovalori ed autovettori Norme di vettori e matrici. Spazi vettoriali lineari Base di uno spazio vettoriale. Trasformazioni lineari e matrici. Sottospazi vettoriali. Spazio vettoriale generato da un insieme di vettori. Nucleo e immagine di una trasformazione lineare. Sottospazio ortogonale.</p> <p>4. Algoritmi per la soluzione di sistemi lineari Il Python per l'algebra lineare, Memorizzazioni di vettori e matrici, operazioni sulle matrici. Sistemi triangolari inferiori e superiori. Matrici di permutazione e proprietà. Algoritmo di eliminazione di Gauss. Problematiche di stabilità. Teorema di esistenza della fattorizzazione LU con pivot. Studio del condizionamento di un sistema lineare. Studio del residuo. Metodo dei minimi quadrati. Pseudoinversa. Cenni sulla decomposizione ai valori singolari (SVD). Esempi di codici Python, documentazione e testing, confronti con il software esistente.</p> <p>5. Interpolazione e approssimazione Base delle potenze. Interpolazione di Lagrange. Interpolazione di Newton. Differenze divise. Interpolazione con nodi coincidenti. Interpolazione di Hermite. Errore nell'interpolazione polinomiale. Scelta dei nodi per l'interpolazione. Studio del condizionamento. Interpolazione lineare a tratti. Spline lineare. Approssimazione ai minimi quadrati nel discreto. Fitting di dati: polinomio di migliore approssimazione nel senso dei minimi quadrati. Retta di regressione lineare. Esempi di codici Python, documentazione e testing, confronti con il software esistente.</p> <p>6. Calcolo degli zeri di funzione Metodo delle bisezioni. Convergenza. Criteri di arresto e stime dell'errore. Ordine di convergenza. Iterazione funzionale. Teorema di contrazione. Ordine di convergenza. Il metodo di Newton. Criteri di stop. Metodi quasi newtoniani. Metodo della direzione costante. Metodo della falsa posizione. Il metodo delle secanti. Metodo di Brent. Errore accuratezza e numero di condizione. Esempi di codici Python, documentazione e testing, confronti con il software esistente.</p>

	<p><u>7. Calcolo degli autovalori.</u> Localizzazione degli autovalori mediante i cerchi di Gerschgorin, Metodo delle potenze e delle potenze inverse. Applicazione: google page rank. Esempi di codici Python.</p> <p><u>8. Calcolo degli integrali.</u> Metodo dei trapezi, di Simpson, dei trapezi composto, di Simpson composto per il calcolo di integrali definiti. Stime dell'errore e metodo dei trapezi adattativo. Esempi di codici Python.</p>
--	---

Programma	
Testi di riferimento	<p>Uri M. Ascher, Chen Greif, A First Course in Numerical Methods, SIAM, 2011</p> <p>L. Brugnano, C. Magherini, A. Sestini, Calcolo numerico, seconda edizione, Master, Università & Professioni, Firenze 2010.</p> <p>James F. Epperson, Introduzione all'analisi numerica, teoria, metodi, algoritmi. McGraw-Hill, Milano, 2003</p>
Note ai testi di riferimento	I libri di testo sono integrati con le slide, le dispense e gli appunti (elettronici) del docente distribuiti sulla piattaforma ADA
Metodi didattici	Lezioni frontali ed esercitazioni/laboratorio in aula
Metodi di valutazione (indicare almeno la tipologia scritto, orale, altro)	L'esame consiste in una prova orale che verterà su tutti gli argomenti svolti a lezione, inclusi le parti teoriche (definizioni, teoremi e dimostrazioni) e gli esercizi ad esse relative. L'esame prevede anche la discussione relativa alla soluzione di alcuni problemi matematici usando gli algoritmi trattati a lezione ed implementati in ambiente Python.
Criteri di valutazione (per ogni risultato di apprendimento atteso su indicato, descrivere cosa ci si aspetta lo studente conosca o sia in grado di fare e a quale livello al fine di dimostrare che un risultato di apprendimento è stato raggiunto e a quale livello)	<p>Lo studente deve comprendere e saper illustrare le problematiche relative dell'uso del calcolatore per la risoluzione dei problemi matematici analizzati durante il corso. Saper individuare il metodo numerico più idoneo per risolvere numericamente un problema matematico tra quelli trattati nel corso, conoscere le tecniche e i metodi per la programmazione numerica finalizzati alla sua risoluzione.</p> <p>Saper definire in modo rigoroso i problemi matematici trattati nel corso e saper esporre i relativi metodi numerici, delineandone le proprietà fondamentali.</p>
Altro	<p>Durante le lezioni verranno discussi, in modo partecipato, diversi quesiti ed esercizi simili per tipologia a quelli comunemente somministrati durante gli esami e gli esoneri. La finalità è duplice: monitorare in tempo reale lo stato di preparazione degli studenti frequentanti, perfezionandone la preparazione in vista dell'esame o degli esoneri; agevolare lo studio in itinere degli aspetti pratici della disciplina, motivando concretamente i corsisti a sostenere l'esame in tempi brevi, sfruttando possibilmente la modalità degli esoneri. Sono previsti due esoneri: il primo durante l'interruzione delle lezioni a metà corso, il secondo a fine corso. Entrambe le date sono concordate, nei limiti consentiti, con gli studenti frequentanti. Di seguito sono elencati ulteriori vantaggi riservati ai frequentanti.</p> <p>- Il superamento di un solo esonero consentirà allo studente di recuperare in sede di esame unicamente la parte mancante, la cui valutazione verrà poi mediata con il voto conseguito nella prova di esonero superata.</p> <p>- Lo studente che superi entrambi gli esoneri, può decidere comunque di rinunciare al voto conseguito in una delle due prove, recuperando in sede di esame la relativa parte (utile se c'è molta discrepanza nei voti conseguiti nelle due prove).</p> <p>- Lo studente che superi entrambi gli esoneri, presenterà e discuterà in sede di esame un lavoro applicativo sul calcolatore con l'obiettivo di migliorare la</p>

	sua valutazione finale (la valutazione verrà mediata con il voto conseguito).
--	---