



UNIVERSITÀ DEGLI STUDI DI PALERMO

DIPARTIMENTO	Matematica e Informatica
ANNO ACCADEMICO OFFERTA	2022/2023
ANNO ACCADEMICO EROGAZIONE	2023/2024
CORSO DILAUREA	INFORMATICA
INSEGNAMENTO	ALGORITMI E STRUTTURE DATI
TIPO DI ATTIVITA'	B
AMBITO	50166-Discipline Informatiche
CODICE INSEGNAMENTO	16670
SETTORI SCIENTIFICO-DISCIPLINARI	INF/01
DOCENTE RESPONSABILE	GIANCARLO RAFFAELE Professore Ordinario Univ. di PALERMO
ALTRI DOCENTI	
CFU	9
NUMERO DI ORE RISERVATE ALLO STUDIO PERSONALE	153
NUMERO DI ORE RISERVATE ALLA DIDATTICA ASSISTITA	72
PROPEDEUTICITA'	05880 - PROGRAMMAZIONE E LABORATORIO C.I.
MUTUAZIONI	
ANNO DI CORSO	2
PERIODO DELLE LEZIONI	2° semestre
MODALITA' DI FREQUENZA	Facoltativa
TIPO DI VALUTAZIONE	Voto in trentesimi
ORARIO DI RICEVIMENTO DEGLI STUDENTI	GIANCARLO RAFFAELE Lunedì 15:00 17:00 Stanza 106 Dipartimento di Matematica ed Informatica Giovedì 15:00 17:00 Stanza 106 Dipartimento di Matematica ed Informatica

DOCENTE: Prof. RAFFAELE GIANCARLO

PREREQUISITI	Conoscenze base di programmazione e di matematica discreta, così come specificato dai programmi dei corsi propedeutici all'insegnamento.
RISULTATI DI APPRENDIMENTO ATTESI	<p>Conoscenza e capacità di comprensione Acquisizione degli strumenti di base per l'analisi ed il progetto di algoritmi (relazioni di ricorrenza, notazione asintotica, paradigmi per progetto di algoritmi e strutture dati). Capacità di utilizzare il linguaggio specifico proprio della disciplina.</p> <p>Capacità di applicare conoscenza e comprensione Capacità di sviluppare software basati su algoritmi efficienti per problemi elementari. In particolare, algoritmi seminumerici come algoritmi di ricerca e visita su alberi e grafi, algoritmi di sorting,</p> <p>Autonomia di giudizio Essere in grado di valutare le implicazioni e i risultati degli studi algoritmici esposti durante il corso e della complessità computazionale dei problemi ad essi associati.</p> <p>Abilità comunicative Capacità di esporre i risultati salienti degli studi algoritmici, anche ad un pubblico non esperto. Essere in grado di evidenziare le ricadute tecnologiche delle teorie studiate.</p> <p>Capacità d'apprendimento Capacità di aggiornamento con la consultazione di testi avanzati e pubblicazioni scientifiche propri del settore dell'algoritmica. Capacità di seguire, utilizzando le conoscenze acquisite nel corso, sia corsi di master di primo livello, che corsi di laurea magistrali.</p> <p>Risultati attesi Acquisizione di strumenti per lo sviluppo e analisi di software seminumerico, su strutture dati elementari, alberi, grafi e su insiemi ordinati e non. Acquisizione di strumenti analitici e del loro utilizzo nell'illustrare la complessità di un algoritmo, e che motivino la scelta di un algoritmo piuttosto che un altro per la soluzione di un problema.</p>
VALUTAZIONE DELL'APPRENDIMENTO	<p>Prova Orale, Prova Scritta, Prova Pratica. La prova orale consiste in un colloquio, che basandosi sui risultati della prova scritta e di laboratorio, verifica il possesso delle competenze e delle conoscenze disciplinari previste dal corso; la valutazione viene espressa in trentesimi.</p> <p>Le domande (input) per la prova scritta e la prova di laboratorio, sia aperte sia semi-strutturate e appositamente pensate per testare i risultati di apprendimento previsti, tenderanno a verificare a) le conoscenze acquisite; b) le capacità elaborative, c) il possesso di un'adeguata capacità espositiva.</p> <p>Gli aspetti menzionati fin qui, contribuiranno alla formazione del voto, tenendo conto della prova scritta e sua discussione (contributo al voto del 60%) e della prova pratica e sua discussione (contributo al voto del 40%). La scala di merito finale è la seguente:</p> <p>a) Conoscenza di base dell'algoritmica, e capacità limitata di applicarla autonomamente in situazioni nuove, sufficiente capacità di analisi algoritmica e di esposizione delle procedure concettuali seguite (voto 18-21); b) Buona conoscenza di base dell'algoritmica e buona capacità di applicarla autonomamente in situazioni nuove, buona capacità di analisi algoritmica e di esposizione delle procedure concettuali seguite (voto 22-25); c) Conoscenza approfondita dell'algoritmica avanzata, e capacità di applicarla prontamente ed autonomamente in situazioni nuove, ma con qualche incertezza; capacità di analisi algoritmica e di esposizione delle procedure concettuali seguite molto buona (voto 26-28); d) Conoscenza approfondita e diffusa dell'algoritmica, ottima capacità di applicarla prontamente ed autonomamente in situazioni nuove; ottima capacità di analisi algoritmica e di esposizione delle procedure concettuali seguite. (voto 29-30L);</p>
OBIETTIVI FORMATIVI	Esporre lo studente a tecniche fondamentali di progetto ed analisi di algoritmi. In particolare, si copre tutto lo spettro delle strutture dati fondamentali e dei principali paradigmi algoritmici, con cenni allo studio di complessità computazionale di problemi intrattabili. Si trattano anche aspetti ingegneristici fondamentali per l'implementazione di algoritmi efficienti.
ORGANIZZAZIONE DELLA DIDATTICA	Lezioni frontali/Lezioni laboratorio
TESTI CONSIGLIATI	

	<p>A.V. Aho, J.E. Hopcroft, J.D. Ullman, The Design and Analysis of Computer Algorithms, Addison Wesley. 1974- ISBN 9780201000290. Libro di Testo Principale- Main Textbook</p> <p>Per i seguenti argomenti specifici, si utilizzeranno i seguenti testi (For the specific topics listed below, the textbook is as follows).</p> <p>T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein - Introduzione agli Algoritmi e strutture dati, McGraw Hill, 2010. ISBN-10 883866515X</p> <p>Hashing</p> <p>C. Demetrescu, I. Finocchi, G.F. Italiano, Algoritmi e Strutture Dati, McGraw-Hill, 2008. ISBN-10 8838664684.</p> <p>Nozioni Introduttive. Tecniche Empiriche e Matematiche per l'Analisi di Algoritmi. Algoritmi di Ordinamento (Introductory Notions. Mathematical and Empirical Techniques for the Analysis of Algorithm. Sorting Algorithms).</p> <p>H. Horowitz, S. Sahni, Fundamentals of Computer Algorithms, Computer Science Press, 1984. ISBN-10 : 9780914894223.</p> <p>Paradigma Greedy (Greedy Paradigm).</p> <p>Materiale distribuito dal docente (Handouts from the instructor). Tutta la parte di Lezioni di Programmazione Algoritmica. (All part regarding Algorithmic Programming).</p>
--	---

PROGRAMMA

ORE	Lezioni
2	NOZIONI INTRODUTTIVE Algoritmi e Strutture Dati. Nozioni introduttive per la soluzione algoritmica di un problema, diverse soluzioni per uno stesso problema. Il problema dell'efficienza di un algoritmo.
6	TECNICHE EMPIRICHE E MATEMATICHE PER L'ANALISI DI ALGORITMI Analisi degli algoritmi. Velocità di crescita delle funzioni. Ricorrenze Fondamentali. Soluzioni delle equazioni di ricorrenza. Metodo dell'iterazione e Master Theorem.
4	MODELLI di CALCOLO, COMPLESSITA' COMPUTAZIONALE E ALGORITMI Random Access Machines, Complessità Computazionale RAM, Macchine di Turing e relazione fra esse (linee generali).
3	ALGORITMI DI ORDINAMENTO Lower bound per gli algoritmi di ordinamento: caso pessimo e caso medio. Principali algoritmi di sorting
5	PARADIGMI PER IL PROGETTO DI ALGORITMI EFFICIENTI: Divide et Conquer. Esempi: Ricerca Minimo e Massimo, Moltiplicazione d'interi, Moltiplicazione di Matrici; Mergesort; Il Quicksort. Analisi worst case e analisi caso medio.
5	PARADIGMI PER IL PROGETTO DI ALGORITMI EFFICIENTI: Programmazione Dinamica e Tecniche Greedy. Esempi: Prodotto di n matrici. Longest Common Subsequence, Riconoscimento Grammatiche Context Free. Algoritmi Greedy: Optimal Storage on Tapes. Il Problema dello Zaino (versione "greedy").
5	STRUTTURE DATI AVANZATE ED OPERAZIONI SU INSIEMI-TABELLE E STRUTTURE STATICHE Operazioni Fondamentali su Insiemi. Tabelle Hash. Union-find.
5	STRUTTURE DATI AVANZATE ED OPERAZIONI SU INSIEMI: ALBERI Alberi di Ricerca Ottimi, Schemi di Alberi Bilanciati, Dizionari e Code a Priorità, Mergeable Heaps, Code Concatenabili.
5	ALGORITMI SU GRAFI: RAPPRESENTAZIONE e CONNETTIVITA' Rappresentazione di Grafi, Visite su Grafi, Biconnettività e Connettività Forte,
5	ALGORITMI SU GRAFI: Problemi di Ottimizzazione. Algoritmi di Spanning Tree Minimo, Algoritmi per Cammini Ottimi.
3	TEORIA DELL' NP- COMPLETEZZA Macchine di Turing Non Deterministiche. Cenni sulle Classi P NP ed NP Completi.
1	Lezione di Programmazione Algoritmica. STRUTTURE DATI ELEMENTARI IN C (Riepilogo da Programmazione) Array, liste concatenate, stringhe e loro implementazione in C: Riepilogo da Programmazione

PROGRAMMA

ORE	Lezioni
3	Lezioni di Programmazione Algoritmica. STRUTTURE DATI ASTRATTE IN C Pile, Code e loro implementazione in C mediante array e liste concatenate. Valutazione di un'espressione in forma postfissa mediante una pila e sua implementazione in C. Tali implementazioni sono un riepilogo da Programmazione.
5	Lezioni di Programmazione Algoritmica. ALGORITMI DI SORTING IN C BASATI SU CONFRONTI. Esempi di implementazione: selectionsort; insertionsort; bubblesort; Mergesort, Heapsort, Quicksort. Implementazione di algoritmi di sorting in C.
4	Lezioni di Programmazione Algoritmica. -ALGORITMI DI SORTING IN C: IL CASO DEGLI INTERI. Esempi di implementazioni: bucketsort; sorting lessicografico.
6	Lezioni di Programmazione Algoritmica. PARADIGMI DI PROGETTO DI ALGORITMI IN C. Ricorsione. Divide et Impera: ricerca del minimo e del massimo, ricerca binaria e loro implementazione in C. Programmazione Dinamica: Prodotto Ottimo di Serie di Matrici
5	Lezioni di Programmazione Algoritmica. ALBERI E GRAFI. Strutture dati per la rappresentazione di grafi ed alberi in C. Algoritmi di visita su alberi in C. Visita DFS e BFS in C.