UNIVERSITÀ
DEGLI STUDI DI MILANO-BICOCCA

COURSE SYLLABUS

Algoritmi e Strutture Dati

2627-1-E3102Q106

Obiettivi

Scopo del corso è insegnare allo studente come progettare, valutare e implementare algoritmi efficienti, utilizzando in modo opportuno le tecniche di programmazione e le strutture dati adeguate.

Conoscenza e capacità di comprensione

- Conoscenza dei concetti fondamentali relativi ai problemi computazionali
- Conoscenza delle tecniche base per dimostrare la correttezza di un algoritmo
- Conoscenza del funzionamento dei paradigmi di programmazione ricorsiva e divide-et-impera
- Conoscenza del funzionamento di algoritmi di ordinamento
- Conoscenza delle tecniche per valutare l'efficienza di algoritmi e strutture dati
- Conoscenza del funzionamento e delle caratteristiche di diverse strutture dati fondamentali

Conoscenza e capacità di comprensione applicate

- Capacità di analizzare la correttezza formale di un algoritmi iterativo e ricorsivo
- Capacità di applicazione dei paradigmi di programmazione ricorsiva e divide-et-impera per risolvere nuovi problemi computazionali
- Capacità di applicazione delle tecniche di valutazione dell'efficienza degli algoritmi iterativi, ricorsivi e divide-et-impera
- Capacità di simulare il funzionamento di algoritmi complessi e strutture dati fondamentali su istanze specifiche
- Capacità di utilizzare correttamente le strutture dati fondamentali per risolvere nuovi problemi computazionali

Autonomia di giudizio

- Capacità di scegliere l'algoritmo più efficiente e/o più indicato per risolvere nuovi problemi computazionali specifici

- Capacità di scegliere la struttura dati più efficiente e/o più indicato per risolvere nuovi problemi computazionali specifici

Abilità comunicative

- Saper utilizzare correttamente lo pseudocodice per presentare formalmente e in modo non ambiguo un algoritmo
- Saper comprendere e utilizzare il lessico corretto per comprendere o descrivere le caratteristiche di un algoritmo e di una struttura dati

Capacità di apprendere

- Essere in grado di ricercare e selezionare criticamente un nuovo algoritmo più indicato per risolvere un nuovo problema computazionale
- Essere in grado di ricercare e selezionare criticamente una nuova struttura dati più indicata per risolvere un nuovo problema computazionale

Contenuti sintetici

Metodologie di base per progettare algoritmi e analizzarne l'efficienza. Strutture dati fondamentali: definizioni e utilizzo

Programma esteso

- Introduzione: Algoritmo, problema, istanza.
- Analisi di algoritmi: Valutazione dei tempi di esecuzione, caso pessimo, ottimo e medio.
- Programmazione ricorsiva e approccio Divide-et-Impera: Mergesort e Quicksort.
- Valutazione del tempo di esecuzione di algoritmi ricorsivi: equazioni di ricorrenza.
- Altri algoritmi di ordinamento: ordinamento in tempo lineare.
- Strutture dati fondamentali: Array, liste, pile e code.
- Alberi Binari e Alberi Binari di Ricerca
- Heap e code con priorità. Heapsort.
- Grafi e loro rappresentazione in memoria.
- Algoritmi di visita dei grafi: BFS e DFS.

Prerequisiti

Nozioni base di programmazione e di analisi matematica

Modalità didattica

Lezioni, esercitazioni e approfondimenti sperimentali. Attività di studio individuali supportate dal libro di testo consigliato.

Le lezioni sono tenute in italiano. Tutte le attività verranno svolte in presenza: 32 ore di lezioni frontali di 2/3 ore ciascuna in modalità erogativa, 44 ore (20 ore di esercitazione di 2/3 ore ciascuna e 24 ore di laboratorio di 4 ore ciascuna) in modalità interattiva.

Materiale didattico

T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, Introduzione agli Algoritmi e Strutture dati, ed. Mc. Graw Hill.

Periodo di erogazione dell'insegnamento

Secondo semestre

Modalità di verifica del profitto e valutazione

L'accertamento dei risultati di apprendimento avviene mediante una **prova scritta individuale**, finalizzata a verificare il raggiungimento degli obiettivi formativi dell'insegnamento relativi alla conoscenza degli algoritmi e delle strutture dati, alla capacità di analizzarne correttezza ed efficienza e alla capacità di applicare tali conoscenze alla soluzione di problemi computazionali.

La prova scritta è valutata in **trentesimi** e comprende:

- Esercizi di progettazione e sviluppo di algoritmi ricorsivi o iterativi per problemi non affrontati direttamente durante il corso, finalizzati a verificare la capacità di applicare conoscenze e metodologie alla risoluzione di problemi (*Descrittore di Dublino 2*);
- Esercizi di simulazione e analisi del comportamento di algoritmi e strutture dati studiati durante il corso, finalizzati a verificare la comprensione dei meccanismi di funzionamento e la capacità di analizzarne correttezza ed efficienza (*Descrittori di Dublino 1 e 2*);
- Domande aperte finalizzate a verificare la conoscenza dei concetti, delle definizioni e delle proprietà fondamentali degli algoritmi e delle strutture dati, nonché la capacità di utilizzare in modo appropriato il lessico disciplinare (*Descrittori di Dublino 1 e 4*).

Le domande di teoria contribuiscono complessivamente fino a **8 punti**.

In alternativa alla prova scritta finale sono previste **due prove parziali** riservate agli iscritti al primo anno, svolte a metà corso e nel primo appello di giugno. Ciascuna prova è strutturata come la prova scritta completa e verifica i risultati di apprendimento relativi agli argomenti trattati fino a quel momento.

Ciascuna prova parziale è valutata in **trentesimi** ed è considerata sufficiente se raggiunge un punteggio di almeno **16/30**. Se entrambe le prove parziali sono sufficienti, allora il voto finale è ottenuto dalla media dei due punteggi delle due prove parziali, approssimata per eccesso. La prova si intende superata se il voto ottenuto è almeno **18/30**.
È possibile recuperare **una sola prova parziale** nell'appello di luglio in caso di assenza, insufficienza o per il miglioramento del risultato conseguito.

Criteri di valutazione

La valutazione delle prove scritte tiene conto dei seguenti aspetti:

- correttezza formale delle soluzioni proposte;
- capacità di individuare e applicare algoritmi e strutture dati appropriati;
- capacità di analizzare correttezza, complessità computazionale ed efficienza delle soluzioni;
- completezza e accuratezza delle risposte;
- chiarezza espositiva e uso corretto del linguaggio tecnico e dello pseudocodice.

La graduazione del voto segue indicativamente i seguenti criteri:

- 18-21/30: raggiungimento dei risultati di apprendimento essenziali;
- 22-25/30: buona comprensione degli argomenti e corretta applicazione delle tecniche principali;
- 26-29/30: conoscenza approfondita degli argomenti e capacità di risolvere autonomamente problemi complessi;
- 30/30: padronanza completa degli argomenti, correttezza rigorosa delle soluzioni e pieno raggiungimento delle capacità comunicative.

Il punteggio di 30 e lode è attribuito a prove eccellenti caratterizzate da pieno raggiungimento degli obiettivi formativi e particolare rigore, completezza e chiarezza espositiva. Il punteggio di 30 e lode conseguito in una prova parziale corrisponde a 31 punti ai fini del calcolo della media.

Orario di ricevimento

Su appuntamento

Sustainable Development Goals