



UNIVERSITÀ DEGLI STUDI DELL'AQUILA

CORSI DI INGEGNERIA

A.A. 2017/2018

Ingegneria degli algoritmi (IAI)

- Frigioni Daniele -

(Aggiornato il 4-06-2018)

Contenuti del corso (abstract del programma):

Il corso intende approfondire gli aspetti prestazionali legati allo sviluppo del software, coniugando il progetto e l'analisi teorica di algoritmi e strutture dati efficienti con la loro effettiva codifica in un linguaggio di programmazione reale e la loro validazione sperimentale su casi reali. Il corso affronterà quindi sia aspetti teorico-metodologici che sperimentali-implementativi. Sono prerequisiti per il corso: Conoscenza delle tecniche di programmazione di base con esperienza di programmazione in un qualsiasi linguaggio. Conoscenza delle strutture di dati elementari come array e liste, e delle principali tecniche per la loro rappresentazione nella memoria RAM di un elaboratore elettronico. Familiarità con dimostrazioni e concetti matematici di base (induzione e dimostrazioni per assurdo). Conoscenza di nozioni elementari di probabilità.

Programma esteso:

Introduzione alle nozioni di algoritmo e struttura dati. Un esempio giocattolo: i numeri di Fibonacci. [1] cap. 1. Modelli di calcolo e metodologie di analisi. I modelli elementari di calcolo. La nozione di complessità asintotica e le notazioni O, Omega e Theta. Delimitazioni inferiori e superiori. Metodi di analisi. Analisi di algoritmi ricorsivi. [1] cap. 2 (esclusi 2.7 e 2.8), [2] cap. 1 (esclusi 4.4 e 5.4). Strutture di dati elementari. Tecniche per rappresentare collezioni di oggetti: array, liste concatenate, pile, code, alberi. [1] cap. 3. Il problema dell'ordinamento. Delimitazione inferiore al numero di confronti per il problema dell'ordinamento. Metodi di ordinamento: SelectionSort, InsertionSort, BubbleSort, HeapSort, MergeSort, QuickSort, IntegerSort, BucketSort, RadixSort. [1] cap. 4. Alberi di ricerca. Alberi binari di ricerca. Il problema del bilanciamento. Alberi AVL. Alberi 2-3. [1] cap. 6 (esclusi 6.3, 6.5 e 6.6). Tabelle Hash. Tabelle ad accesso diretto. Tabelle Hash. Risoluzione delle collisioni. [1] cap. 7, [2] cap. 11 (escluso 11.5). Code con priorità. Definizione e rappresentazione tramite strutture di dati elementari. Heap binari. Heap binomiali. [1] cap. 8 (escluso 8.3). Grafi e visite di grafi. Definizioni. Tecniche di rappresentazione in memoria di un grafo. Algoritmi di visita in ampiezza e profondità. Componenti connesse di un grafo non orientato. Componenti fortemente connesse di un grafo orientato. [1] cap. 12. Minimo albero ricoprente. Proprietà del minimo albero ricoprente. Algoritmo di Kruskal. Algoritmo di Prim. Algoritmo di Boruvka. [1] cap. 13 (escluso 13.2.1). Cammini minimi. Cammini minimi e distanze in un grafo. La tecnica del rilassamento. Algoritmo di Bellmann-Ford. Algoritmo per grafi diretti aciclici. Algoritmo di Dijkstra. Algoritmo di Floyd-Warshall. [1] cap. 14. Teoria della NP-completezza. Complessità di problemi decisionali. Le classi di complessità P e NP. Problemi NP-completi. Algoritmi di approssimazione: copertura di vertici e orientazione di grafi. [1] cap. 16, [2] cap.

34. Ingegneria degli algoritmi. Definizione e metodologie di base. Metodologie dell'ingegneria degli algoritmi: analisi dei requisiti, valutazione difficoltà, progetto algoritmo, analisi teorica, implementazione, analisi sperimentale e ingegnerizzazione. Un caso di studio: il problema dei cammini minimi. [3] cap. 1, più materiale fornito dal docente.

Modalità d'esame:

Prova unica

Risultati d'apprendimento previsti:

Alla fine del corso lo studente sarà in grado di definire e manipolare strutture dati in un linguaggio di programmazione per risolvere problemi algoritmici fondamentali che sono alla base dei moderni sistemi software. Acquisirà inoltre la capacità di analizzare le prestazioni di un programma utilizzando da una parte modelli di costo teorici, e dall'altra metodologie sperimentali su piattaforme di calcolo reali.

Link al materiale didattico:

<http://www.didattica.univaq.it/moodle/course/view.php?id=4540>

Testi di riferimento:

Camil Demetrescu, Irene Finocchi, Giuseppe F. Italiano. Algoritmi e Strutture Dati (seconda edizione), McGraw-Hill, 2008 (Testo di riferimento)

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein. Introduzione agli Algoritmi e Strutture Dati (seconda edizione), McGraw-Hill, 2005 (Testo di approfondimento)

Camil Demetrescu, Umberto Ferraro Petrillo, Irene Finocchi, Giuseppe F. Italiano. Progetto di Algoritmi e Strutture Dati in Java, McGraw-Hill, 2007 (Testo di approfondimento)

Matthias Muller-Hannemann, Stefan Schirra. Algorithm Engineering (Bridging the Gap Between Algorithm Theory and Practice). Lecture Notes in Computer Science n. 5971, Springer, 2010 (Testo di approfondimento).