UNIVERSITÀ
DEGLI STUDI DI MILANO-BICOCCA

SYLLABUS DEL CORSO

Analisi e Progetto di Algoritmi

2526-3-E3101Q113

Obiettivi

Conoscenza e capacità di comprensione
Questo insegnamento fornisce le conoscenze basilari e capacità di comprensione relativamente a:

- tecnica algoritmica della programmazione dinamica per la risoluzione di problemi di ottimizzazione combinatoria
- algoritmi paradigmatici per problemi di ottimizzazione combinatoria su sequenze, insiemi e grafi (Longest Common Subsequence, Weighted Interval Scheduling, ... e loro varianti) basati sulla tecnica della programmazione dinamica e proprietà della sottostruttura ottima.
- tecnica algoritmica della programmazione greedy per la risoluzione di problemi di ottimizzazione combinatoria
- strutture dati per insiemi disgiunti e, in base alla rappresentazione ed ai modi di implementare le operazioni, complessità computazionale degli algoritmi che ne fanno uso
- algoritmi per il calcolo di alberi di copertura minimi (Kruskal, Prim)
- algoritmi per il calcolo di cammini minimi su grafi (Floyd-Warshall e Dijkstra)
- tabelle ad indirizzamento diretto e tabelle Hash
- le classi P, NP e problemi NP completi
- riduzione tra problemi

Conoscenza e capacità di comprensione applicate

Capacità di risolvere mediante la tecnica della programmazione dinamica problemi di ottimizzazione combinatoria (e versioni decisionali) su sequenze, insiemi e grafi, essendo in grado di

- definire i sottoproblemi, i relativi coefficienti (variabili) e la struttura dati per memorizzarli.
- riformulare il problema in termini ricorsivi fornendo il caso base ed il passo ricorsivo delle equazioni di ricorrenza
- fornire il valore ottimo del problema sulla base dei valori di tutti i coefficienti
- disegnare un algoritmo bottom-up efficiente per il calcolo dei valori di tutti i coefficienti e del valore ottimo,

stabilendone la complessità computazionale
- disegnare un algoritmo efficiente per la ricostruzione della soluzione del problema (sottosequenza di valore ottimo, sottoinsieme di valore ottimo, cammini sul grafo di valore ottimo)

Capacità di comprendere se e quando la tecnica di programmazione greedy puo' essere utilizzata con successo per risolvere un problema di ottimizzazione combinatoria.

Capacità di utilizzare le strutture dati per insiemi disgiunti nel contesto di algoritmi su grafi.

Capacità di costruire l'istanza di un problema a partire dall'istanza del problema dal quale esso è ridotto.

Autonomia di giudizio

Capacità di individuare la tecnica algoritmica e la struttura dati piu' idonee per risolvere in modo efficiente specifici problemi computazionali.

Abilità comunicative

Capacità di esporre in maniera chiara e rigorosa i contenuti teorici, le tecniche algoritmiche e gli algoritmi, incluse le dimostrazioni.

Capacità di apprendere

Capacità di ricercare e apprendere autonomamente nuovi algoritmi e strutture dati per rispolvere problemi computazionali.
Capacità di affrontare nuovi problemi computazionali.

Contenuti sintetici

L'insegnamento intende introdurre le principali tecniche algoritmiche (programmazione dinamica, greedy), con particolare attenzione agli aspetti di efficienza degli algoritmi, con i relativi strumenti di analisi. Verranno illustrati i principali algoritmi per risolvere vari problemi di ottimizzazione combinatoria specialmente su insiemi, sequenze e grafi, tra cui la ricerca di cammini minimi in un grafo pesato e la costruzione di alberi di copertura minimi.

Programma esteso

1. Strumenti matematici (ripasso)

- Crescita delle funzioni, notazioni asintotiche
- Calcolo del tempo di esecuzione per algoritmi iterativi
- Ricorsione e algoritmi ricorsivi
- Ricorrenze e tempi di calcolo di algoritmi ricorsivi

2. Tecniche algoritmiche: Programmazione Dinamica (DP)

- Esempi introduttivi
- Caratteristiche principali - Ricorsione e proprietà della sottostruttura ottima.
- Implementazione con matrici
- Problemi di ottimizzazione combinatoria su sequenze, insiemi e grafi. Varianti decisionali. Proprietà della

sottostruttura ottima.
- Risoluzione mediante: definizione dei coefficienti (variabili) associati ai sottoproblemi e relativa struttura dati per memorizzarli, formulazione delle equazioni di ricorrenza (caso base e passo ricorsivo) per il calcolo dei valori di tutti i coefficienti, individuazione del valore ottimo sulla base dei valori di tutti i coeffiecienti, algoritmo bottom-up per il calcolo dei coefficienti e del valore ottimo, algoritmo di ricostruzione di una soluzione (sottosequenza, sottoinsieme o cammini di valore ottimo).

3. Tecniche algoritmiche: il metodo Greedy (goloso)

- Esempi introduttivi .
- Analogie e differenze con la tecnica della programmazione dinamica.
- Sistemi di Indipendenza e Matroidi
- Problema di massimo/minimo associato ad un sistema d'indipendenza pesato e relativo algoritmo greedy.
- Teorema di Rado
- Matroide Grafico

4\ Strutture dati per insiemi digiunti

- Definizioni e operazioni.
- Rappresentazione mediante liste concatenate e mediante foreste

5. Alberi di copertura minimi

- Algoritmo generico
- Algoritmo di Kruskal
- Algoritmo di Prim

6. Problemi di cammino minimo

- Algoritmo di Dijkstra
- Algoritmo di Floyd-Warshall

7. Tabelle Hash

- Tabelle ad indirizzamento diretto
- Tabelle Hash

8. Introduzione a NP completezza e riduzioni

- le classi P, NP e problemi NP completi
- riduzione tra problemi: nozione di riduzione e vari esempi

Prerequisiti

Nozioni base di programmazione, algoritmi e strutture dati

Modalità didattica

Lezioni, esercitazioni ed esercitazioni laboratoriali in aula ed in presenza.

Le lezioni (32h) prevedono una modalità didattica erogativa.
Le ore di esercitazioni ed esercitazioni laboratoriali (20h+24h) prevedono una modalità didattica erogativa nella parte iniziale ed interattiva nella parte successiva.

La lingua del corso è l'italiano.

Materiale didattico

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

Materiale integrativo (lucidi ed esercizi) disponibili sul sito e-learning.

Periodo di erogazione dell'insegnamento

Primo semestre

Modalità di verifica del profitto e valutazione

Prova scritta: Essa consiste di:

- esercizi relativi ai contenuti del corso

- domande aperte relative alle nozioni teoriche presentate a lezione

La valutazione totale massima derivante da esercizi e domande aperte è di 31 punti.
L'esame è considerato superato se la valutazione finale è di almeno 18.
Possono essere assegnati ulteriori 3 punti aggiuntivi (relativi ad una domanda/esercizio facoltativo).
Il punteggio finale dà automaticamente origine al voto in trentesimi (30 e lode per punteggi superiori a 30).

Prove parziali:
La prova scritta puo' essere sostituita da due prove parziali, che si tengono a meta' e fine corso.
Ogni prova parziale verte sugli argomenti trattati nella corrispondente parte del corso. Consiste di esercizi e domande aperte relative alle nozioni teoriche presentate a lezione.

Ogni prova parziale ha valutazione massima di 31 punti: la valutazione finale si ottiene dalla media dei voti delle due prove parziali. L'esame è considerato superato se la valutazione di ogni prova parziale è maggiore di 14 e la valutazione finale è di almeno 18.
Possono essere assegnati ulteriori 3 punti aggiuntivi (relativi ad una domanda/esercizio facoltativo).

Il punteggio finale corrisponderà esattamente al voto in trentesimi (30 e lode per punteggi superiori a 30).

Orario di ricevimento

su appuntamento

Sustainable Development Goals