

Algoritmi e Complessità

a.a. 2015-2016

Competenze attese

Si presuppone che lo studente conosca i concetti base dell'informatica e abbia esperienza di programmazione. In particolare si presuppone la conoscenza di alcune tecniche di programmazione, come la tecnica "Divide et Impera" e la tecnica "Greedy", e familiarità con la nozione di complessità concreta per algoritmi iterativi e ricorsivi.

Programma

Progetto e analisi di Algoritmi

- Programmazione dinamica. Uso "top-down" e "bottom-up" di definizioni ricorsive di funzioni. Esempi: Distanza di edit, Massima sottostringa crescente, Zaino, Cammini minimi nei grafi, Prodotto di una sequenza di matrici.
- Backtracking: algoritmi che generano permutazioni e sottoinsiemi. Esempi: Colorazione dei vertici di un grafo, Il problema delle n regine, Commesso Viaggiatore.
- Branch and Bound. Esempi: Zaino, Commesso Viaggiatore.
- La tecnica di annotazione.

Complessità polinomiale in tempo

- Certificati polinomiali, algoritmi non deterministici, le classi P e NP.
- Riducibilità polinomiale, NP-completezza.

Algoritmi di approssimazione

- Approssimazione assoluta ed errore relativo.
- Schemi di approssimazione pienamente polinomiali.

Algoritmi probabilistici

- Algoritmo di Miller-Rabin
- Non commutatività di matrici
- Funzioni one-way.

Modalità d'esame

L'esame consiste di una prova orale e della presentazione di un seminario.

Gli argomenti per la presentazione in aula verranno proposti agli studenti divisi in gruppi di due/tre persone. I seminari verranno tenuti nelle ultime lezioni del semestre ed è richiesta la presenza degli studenti di tutti i gruppi ai seminari dei compagni.

Gli studenti che intendessero sostenere l'esame senza aver partecipato ai seminari sono pregati di contattare il docente per la definizione del programma.