

Syllabus del corso.
Corso di Laurea in Informatica
Prof. A. Monti

1. Presentazione del corso

2. I grafi

- a. Grafi e grafi orientati
- b. Rappresentazione di grafi tramite matrici di adiacenza, liste di adiacenza, dizionari. Vantaggi e svantaggi.
- c. Esercizio: il problema del pozzo universale

3. Visita di grafi in profondità (DFS)

- a. Implementazione ricorsiva con
 - i. vettore booleano
 - ii. insieme dei visitati
- b. implementazione iterativa
- c. Esercizio: il problema della colorazione

4. Applicazioni della DFS

- a. Ricerca delle componenti connesse
- b. L'albero DFS e vettore dei padri
- c. Ricerca dei ponti di un grafo

5. Applicazioni della DFS per grafi diretti

- a. Tipologie degli archi a seguito della visita
- b. Ricerca di cicli
- c. Ordinamento topologico
- d. Esercizio: ricerca delle sorgenti di un nodo.

6. Componenti fortemente connesse

- a. Algoritmo di Kosaraju

7. Visita di grafi in ampiezza (BFS)

- a. Implementazioni con:
 - i. Lista
 - ii. Cancellazione logica
 - iii. Deque
- b. Albero BFS e cammini minimi
- c. Applicazione: il diametro di un grafo

8. Grafi pesati (prima parte)

- a. Il problema dei cammini minimi
- b. Algoritmo di Dijkstra
 - i. Correttezza dell'algoritmo
 - ii. Implementazione con array
 - iii. Implementazione con heap

9. Grafi pesati (seconda parte)

- a. Il problema del minimo albero di copertura
- b. Algoritmo di Kruskal
 - i. Correttezza dell'algoritmo
 - ii. Implementazione tramite insiemi disgiunti

10. La tecnica Greedy

- a. Esempi e correttezza:
 - i. Selezione di attività
 - ii. Assegnazione di aule

11. Problemi di ottimizzazione

- a. Euristiche ed algoritmi di approssimazione
 - i. Il problema della copertura minima tramite vertici
 - 1. Euristiche greedy basate sul grado dei nodi
 - 2. Euristiche greedy con rapporto d'approssimazione 2

12. Algoritmi d'approssimazione per problemi difficili

- a. Il problema della minimizzazione del makespan
- b. Il problema del massimo taglio
- c. Il problema del bin packing

13. La tecnica del divide et impera (prima parte)

- a. Il problema della selezione
 - i. Algoritmo di tempo medio lineare
 - ii. Algoritmo lineare

14. La tecnica del divide et impera (seconda parte)

- a. Il problema della coppia di punti a distanza minima
- b. Il problema del conteggio delle inversioni

15. La tecnica della programmazione dinamica (prima parte)

- a. L'approccio top-down (Memoizzazione)
- b. L'approccio bottom-up (Tabulazione)
- c. Esempi con tabelle unidimensionali
 - i. Stringhe binarie senza zeri adiacenti
 - ii. Sistemazione in albergo
 - iii. Somma massima per sottosequenze non consecutive
 - iv. Ricerca di sottovettore di somma massima
 - v. Ricerca di sottosequenza crescente di lunghezza massima

16. La tecnica della programmazione dinamica (seconda parte)

- a. Esempi con tabelle bidimensionali
 - i. Cammino di somma massima in una matrice
 - ii. Il problema dello zaino
 - iii. Sottosequenza comune più lunga

17. La tecnica della programmazione dinamica (terza parte)

- a. Cammini minimi su grafi con pesi anche negativa
 - i. Algoritmo di Bellman-Ford
 - ii. Algoritmo di Floyd e Warshal

18. La tecnica del Backtracking (parte prima)

- a. Il backtracking come enumeratore:
 - i. Enumerare stringhe
 - ii. Enumerare matrici
 - iii. Enumerare permutazioni
- b. Introduzione dei vincoli e funzioni di taglio

19. La tecnica del Backtracking (seconda parte)

- a. Enumerazione con vincoli, esempi per stringhe matrici e permutazioni

20. La tecnica del Backtracking (terza parte)

- a. Applicazione a problemi di ricerca
 - i. Tricolorazione
 - ii. Cicli hamiltoniani
- b. Applicazione a problemi di ottimizzazione
 - i. Il problema dello zaino