Università Degli Studi Di L'Aquila

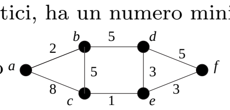
Prova di Recupero di **Algoritmi e Strutture Dati**

Martedì 23 Marzo 2004 – Prof. Guido Proietti

Scrivi i tuoi dati ⟹	Cognome:	Nome:	Matricola:	PUNTI
ESERCIZIO 1	Risposte Esatte:	Risposte Omesse:	Risposte Errate:	
ESERCIZIO 2				
ESERCIZIO 3	Correttezza:	Efficienza:	Analisi:	
TOTALE				

ESERCIZIO 1: Domande a risposta multipla (15 punti)

Premessa: Questa parte è costituita da 15 domande a risposta multipla. Per ciascuna domanda vengono fornite 4 risposte, di cui soltanto una è corretta. Per rispondere utilizzare la griglia annessa, barrando con una $\times$ la casella corrispondente alla risposta prescelta. È consentito omettere la risposta. In caso di errore, contornare con un cerchietto la $\times$ erroneamente apposta (ovvero, in questo modo $\otimes$) e rifare la $\times$ sulla nuova risposta prescelta. Se una domanda presenta più di una risposta, verrà considerata omessa. Per tutti i quesiti verrà attribuito un identico punteggio, e cioè: risposta esatta 3 punti, risposta omessa 0 punti, risposta sbagliata -1 punto. Il voto relativo a questa parte è ottenuto sommando i punti ottenuti e normalizzando su base 15. Se tale somma è negativa, verrà assegnato 0.

1. L'algoritmo Insertion Sort, nel caso peggiore costa:
   a) $O(n \log n)$ b) $o(n^2)$ c) $\Theta(n^2)$ d) $\omega(n^2)$
2. Se $f(n) = 4n \log n$ e $g(n) = 3n + 12n \log n^2$, quale delle seguenti relazioni asintotiche è vera:
   a) $f(n) = o(g(n))$ b) $f(n) = \omega(g(n))$ c) $g(n) = o(f(n))$ d) $f(n) = O(g(n))$
3. Siano $f(n)$ e $g(n)$ i costi dell'algoritmo Selection Sort nel caso migliore e in quello medio, rispettivamente. Quale delle seguenti relazioni asintotiche è vera:
   a) $f(n) = \Omega(g(n))$ b) $f(n) = \omega(g(n))$ c) $g(n) = o(f(n))$ d) $f(n) = o(g(n))$
4. L'algoritmo ottimale di fusione di due sequenze ordinate di lunghezza n e $\log n$ rispettivamente, ha complessità:
   a) $\Theta(n \log n)$ b) $\Omega(n \log n)$ c) $\Theta(\log n)$ d) $\Theta(n)$
5. A quale delle seguenti classi non appartiene la complessità dell'algoritmo Merge Sort:
   a) $o(n \log n)$ b) $\Omega(n)$ c) $O(n^2)$ d) $\Theta(n \log n)$
6. Quale dei seguenti vettori non rappresenta un heap binario:
   a) A=[10,9,6,7,5,1] b) A=[20,16,9,15,12,14] c) A=[20,16,9,15,12] d) A=[5,3,4]
7. La procedura *Heapify*$(A, 2)$ applicata al vettore $A = [12, 9, 2, 6, 5, 3]$ restituisce:
   a) $A = [12, 9, 3, 6, 5, 2]$ b) $A = [12, 6, 2, 5, 9, 3]$ c) $A = [12, 5, 3, 6, 9, 2]$ d) $A = [12, 9, 2, 6, 5, 3]$
8. La procedura Extract-Max(A) applicata alla coda di priorità rappresentata tramite vettore non ordinato $A = [12, 9, 3, 6, 5, 2]$, restituisce:
   a) $A = [9, 3, 6, 5, 2, nil]$ b) $A = [12, 9, 3, 6, 5, 2]$ c) $A = [2, 9, 3, 6, 5]$ d) $A = [2, 9, 3, 6, 5, nil]$
9. L'*inserimento* di un elemento in una lista lineare ordinata di n elementi costa:
   a) $\Theta(\log n)$ b) $\Theta(1)$ c) $\Theta(n)$ d) $O(n)$
10. Dato un albero binario di ricerca di n elementi ed altezza h, l'inserimento di un elemento restituisce un albero avente al massimo altezza:
    a) $h + 1$ b) $\Theta(\log h)$ c) $\Theta(\log n)$ d) h
11. Un grafo *non connesso* di n vertici, ha un numero minimo di archi pari a: a) 0 b) $n - 1$ c) $n - 2$ d) 1
12. La visita in profondità del grafo 

eseguita partendo dal nodo a non può restituire la sequenza di nodi:
    a) *acefdb* b) *abcdef* c) *abdfec* d) *acbdef*
13. L'albero dei cammini minimi radicato in b del grafo di domanda 12 ha altezza:
    a) 2 b) 3 c) 4 d) 5
14. Usando l'euristica dell'unione pesata, il problema della gestione di n insiemi disgiunti sottoposti ad m operazioni può essere risolto in:
    a) $\Theta(m + n \log n)$ b) $\Theta(m)$ c) $\Theta(m^2)$ d) $O(m + n \log n)$
15. Dato il grafo di domanda 12, l'arco di peso minimo che attraversa il taglio $(\{a, d\}, \{b, c, e, f\})$ è:
    a) (c, e) b) (d, e) c) (a, b) d)(d, f)

Griglia Risposte

	Domanda														
Risposta	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
a															
b															
c															
d															

ESERCIZIO 2: Domande a risposta aperta (6 punti)

Premessa: Questa parte è costituita da 2 domande a risposta aperta. Rispondere ad una sola domanda selezionata a piacere. La risposta giudicata corretta ed esaustiva è valutata 6 punti.

1. Illustrare dettagliatamente la tecnica per la determinazione del lower bound del problema dell'*ordinamento* tramite confronti.
2. Discutere il problema della *gestione di una collezione di insiemi disgiunti*, descrivendo dettagliatamente la tecnica di risoluzione attraverso l'uso di liste concatenate e l'euristica delle unioni pesate.

ESERCIZIO 3: Realizzazione di un algoritmo (9 punti)

Premessa: L'esercizio verrà valutato soltanto se corredato da adeguata descrizione del funzionamento dell'algoritmo, ed in base ai seguenti parametri: correttezza algoritmo (4 punti), efficienza algoritmo (3 punti) ed analisi della complessità (2 punti).

Data una componente connessa in un grafo pesato, il *peso della componente* è pari alla somma dei pesi di tutti gli archi che appartengono alla componente. Realizzare ed analizzare un algoritmo che, preso in input un grafo pesato e non orientato $G = (V, E)$, con n vertici ed m archi e rappresentato tramite liste di adiacenza, restituisca il peso della componente di peso massimo.