Università degli Studi dell'Aquila

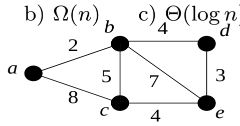
Prova Totale di Teoria del corso di **Algoritmi e Strutture Dati con Laboratorio**

Mercoledì 2 Febbraio 2011 – Prof. Guido Proietti

Scrivi i tuoi dati ⟹	Cognome:	Nome:	Matricola:	PUNTI
ESERCIZIO 1	Risposte Esatte:	Risposte Omesse:	Risposte Errate:	

ESERCIZIO 1 (25 punti): Domande a risposta multipla

Premessa: Questa parte è costituita da 20 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 25. Se tale somma è negativa, verrà assegnato 0.

1. Detto F_n l'n-esimo numero della sequenza di Fibonacci, e detta $\phi = 1,618\ldots$ la sezione aurea, quale delle seguenti relazioni asintotiche è vera? a) $F_n = \Theta(2^n)$ *b) $F_n = \Omega(\phi^n)$ c) $F_n = o(\phi^n)$ d) $F_n = \omega(\phi^n)$
2. $f(n) = \Theta(n)$ se e solo se:
a) $f(n) = O(n)$ e $f(n) = \omega(n)$ *b) $f(n) = O(n)$ e $f(n) = \Omega(n)$ c) $f(n) = o(n)$ e $f(n) = \omega(n)$ d) $f(n) = o(n)$ e $f(n) = \Omega(n)$
3. Quale delle seguenti relazioni asintotiche è falsa:
a) $n = \Theta(3n)$ b) $2^n = O(3^n)$ *c) $n^2 \log n^2 = \omega(n^2 \log n)$ d) $n = o(n \log n)$
4. Il numero di foglie dell'albero di decisione di un qualsiasi algoritmo per il problema della ricerca in un insieme ordinato è:
a) $\Theta(n \log n)$ b) $\Theta(\log n)$ *c) $\Omega(n)$ d) $\Omega(n!)$
5. La delimitazione inferiore al problema dell'ordinamento ottenibile dagli alberi di decisione è:
a) $\Theta(\log n)$ b) $\omega(n \log n)$ *c) $\Omega(n \log n)$ d) $\Theta(n)$
6. A quale delle seguenti classi appartiene la complessità dell'algoritmo Merge Sort:
*a) $\Omega(n \log n)$ b) $\Omega(n^2)$ c) $O(n)$ d) $\Theta(n^2)$
7. A quale delle seguenti classi appartiene la complessità del caso medio dell'algoritmo Quicksort:
a) $\Theta(n^2)$ *b) $\Theta(n \log n)$ c) $O(n)$ d) $\Omega(n^2)$
8. Siano $f(n)$ e $g(n)$ i costi degli algoritmi Heapsort e Quicksort, rispettivamente. Quale delle seguenti relazioni asintotiche è vera:
a) $g(n) = o(f(n))$ b) $f(n) = \Theta(g(n))$ c) $f(n) = \omega(g(n))$ *d) $g(n) = \omega(f(n))$
9. Quale dei seguenti vettori <u>non</u> rappresenta un heap:
a) A=[5,3,4,1,2] *b) A=[20,19,12,13,14,15] c) A=[5,4,3,2,1] d) A=[5]
10. La procedura *FixHeap* per il mantenimento di un heap, nel caso migliore costa:
a) $\Theta(\log n)$ b) $\Theta(n)$ *c) $\Theta(1)$ d) $\Theta(n \log n)$
11. Sia H_1 un heap binomiale costituito dagli alberi binomiali $\{B_0, B_1, B_2\}$, e sia H_2 un heap binomiale costituito dagli alberi binomiali $\{B_0, B_1, B_3\}$. Da quali alberi binomiali è formato l'heap binomiale ottenuto dalla fusione di H_1 e H_2?
*a) $\{B_1, B_4\}$ b) $\{B_0, B_1, B_2, B_3, B_4\}$ c) $\{B_0, B_0, B_1, B_1, B_2, B_3\}$ d) $\{B_0, B_1, B_2, B_3\}$
12. In un albero AVL di n elementi, la cancellazione di un elemento nel caso migliore induce un numero di rotazioni pari a:
*a) 0 b) 2 c) $\Theta(\log n)$ d) 1
13. In una tavola ad accesso diretto di dimensione m con un fattore di carico $\alpha = 1\%$, l'inserimento di un elemento di un dizionario di n elementi costa: a) $\Theta(m)$ b) $\Omega(n)$ c) $\Theta(\log n)$ *d) $\Theta(1)$
14. La visita in profondità del grafo 

eseguita partendo dal nodo d produce un albero DFS di altezza massima:
a) 1 b) 2 *c) 4 d) 5
15. Un grafo $G = (V, E)$ si dice *bipartito* se l'insieme V può essere partizionato in due sottoinsiemi V_1, V_2 tali che tutti gli archi in E hanno un nodo in V_1 e l'altro in V_2. Sia dunque $G = (V_1 \cup V_2, E)$ un grafo bipartito tale che $|V_1| = 4, |V_2| = 3$. Quanti archi sono necessari affinché G sia connesso?
a) 3 b) 4 c) 5 *d) 6
16. Dato un grafo pesato e completo con n vertici, l'algoritmo di Dijkstra realizzato con un heap binario costa:
*a) $\Theta(n^2 \log n)$ b) $\Theta(m + n \log n)$ c) $\Theta(n^2)$ d) $O(n \log n)$
17. Sia d_{xy}^k il costo di un cammino minimo k-vincolato da x a y, secondo la definizione di Floyd e Warshall. Risulta:
a) $d_{xy}^k = \min\{d_{xy}^{k-1}, d_{xv_k}^{k-1} + d_{v_k x}^{k-1}\}$ *b) $d_{xy}^k = \min\{d_{xy}^{k-1}, d_{xv_k}^{k-1} + d_{v_k y}^{k-1}\}$
c) $d_{xy}^k = \min\{d_{xy}^{k-1}, d_{xv_k}^{k} + d_{v_k y}^{k}\}$ d) $d_{xy}^k = \min\{d_{xy}^{k}, d_{xv_k}^{k-1} + d_{v_k y}^{k-1}\}$
18. L'operazione *Union*(A, B) di 2 insiemi disgiunti A, B con alberi *QuickFind* senza l'euristica dell'unione pesata costa nel caso peggiore:
a) $\Theta(\min(|A|, |B|))$ b) $\Theta(\max(|A|, |B|))$ c) $\Theta(|A|)$ *d) $\Theta(|B|)$
19. Dato un grafo pesato con n vertici ed m archi, l'algoritmo di Kruskal esegue un numero di operazioni Union(u, v) pari a:
a) $\Theta(m)$ *b) $\Theta(n)$ c) $\Theta(m \log n)$ d) $\Theta(\log n)$
20. Dato un grafo pesato con n vertici ed m archi, il costo di una fase dell'algoritmo di Borůvka è pari a:
*a) $O(m)$ b) $O(n)$ c) $\Theta(m + n \log n)$ d) $\Theta(m \log n)$

Griglia Risposte

	Domanda																			
Risposta	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
a																				
b																				
c																				
d																				

ESERCIZIO 2 (5 punti) (**Da svolgere sul retro della pagina!**)

La *somma* di 2 grafi $G_1 = (V_1, E_1)$ e $G_2 = (V_2, E_2)$ è un grafo $G = (V, E)$ in cui $V = V_1 \cup V_2$, ed $E = E_1 \cup E_2 \cup \{(x, y) | x \in V_1, y \in V_2\}$. Sia G il grafo ottenuto sommando un ciclo di 4 nodi ed un grafo connesso di 2 nodi. Numerare in modo arbitrario i vertici di G da 1 a 6, e pesare ogni arco come somma dei numeri associati ai vertici incidenti. Restituire quindi il minimo albero ricoprente di G, mostrando l'esecuzione passo per passo dell'algoritmo di Kruskal.