Università degli Studi dell'Aquila

Prova Scritta di **Algoritmi e Strutture Dati con Laboratorio**

Mercoledì 27 Gennaio 2016 – Prof. Guido Proietti (Modulo di Teoria)

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

ESERCIZIO 1: Domande a risposta multipla

Premessa: 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 finale è ottenuto sommando i punti ottenuti e normalizzando su base 30. Se tale somma è negativa, verrà assegnato 0.

1. Quale delle seguenti relazioni di ricorrenza descrive la complessità dell'algoritmo più efficiente per il calcolo della sequenza di Fibonacci basato sul prodotto di matrici?
   a) $T(n) = 2T(n/2) + O(1)$ se $n \geq 2$, $T(1) = O(1)$ se $n = 1$ b) $T(n) = 2T(n/4) + O(1)$ se $n \geq 2$, $T(1) = O(1)$ se $n = 1$
   *c) $T(n) = T(n/2) + O(1)$ se $n \geq 2$, $T(1) = O(1)$ se $n = 1$ d) $T(n) = 2T(n/2) + O(1)$ se $n \geq 2$, $T(1) = O(n)$ se $n = 1$
2. Sia dato in input l'array $A = [5, 7, 4, 3, 10]$; quanti confronti tra elementi esegue l'algoritmo InsertionSort2 per ordinare in ordine non decrescente A?
   a) 10 b) 5 c) 25 *d) 7
3. Siano $f(n)$ e $g(n)$ i costi dell'algoritmo MergeSort e QuickSort nel caso migliore, rispettivamente. Quale delle seguenti relazioni asintotiche è falsa:
   *a) $f(n) = o(g(n))$ b) $f(n) = \Omega(g(n))$ c) $g(n) = O(f(n))$ d) $f(n) = \Theta(g(n))$
4. Come si esegue l'operazione `increaseKey`(elem e, chiave Δ) di un elemento con chiave k in un heap binomiale?
   a) eseguendo `insert`$(e, k + \Delta)$ seguita da `delete`(e) b) eseguendo `decreaseKey`$(e, -\Delta)$
   c) eseguendo `delete`(e) seguita da `insert`(e, Δ) *d) eseguendo `delete`(e) seguita da `insert`$(e, k + \Delta)$
5. Sia dato un AVL di n elementi nel quale si eseguono in successione $O(1)$ cancellazioni e $\Theta(\log n)$ inserimenti. Nel caso peggiore, quante rotazioni subirà l'AVL?
   a) $\Theta(1)$ b) $\Theta(n)$ c) $\Theta(\log^2 n)$ *d) $\Theta(\log n)$
6. Si supponga di inserire la sequenza di chiavi $26, 14, 6$ (in quest'ordine) in una tavola hash di lunghezza $m = 3$ (ovvero con indici $0, 1, 2$) utilizzando l'indirizzamento aperto con funzione hash $h(k) = k \mod 3$, e risolvendo le collisioni con il metodo della scansione quadratica con $c_1 = c_2 = 1$. Quale sarà la tavola hash finale?
   a) $A = [14, 6, 26]$ b) $A = [26, 6, 14]$ c) $A = [6, 26, 14]$ *d) $A = [6, 14, 26]$
7. Si consideri il grafo $G = (V, E)$ con $V = \{1, 2, 3, 4, 5\}$ ed $E = \{(1, 2), (1, 5), (2, 4), (2, 5), (3, 5)\}$. Quali delle seguenti affermazioni è <u>falsa</u>:
   *a) G è bipartito b) il BFS di G radicato in 2 ha altezza 2 *c) il *diametro* di G, ovvero la distanza massima tra due nodi in G, è pari a 2 d) G ha grado 3
8. Sia dato un grafo connesso G con n vertici, numerati da 1 ad n, ed $m = \Theta(n \log n)$ archi orientati, disposti in modo arbitrario, ma in modo tale da garantire l'aciclicità. Si applichi ora l'algoritmo di ordinamento topologico rispetto al nodo sorgente etichettato 1. La complessità risultante è pari a:
   a) $\Theta(n^2)$ b) $\Theta(n)$ *c) $\Theta(n \log n)$ d) indefinita (non è detto che l'algoritmo possa essere applicato)
9. Sia dato un grafo pesato $G = (V, E)$ con n nodi ed m archi, senza cicli negativi, e si consideri il problema di trovare i cammini minimi in G tra tutte le coppie di nodi. Quando è conveniente (asintoticamente) applicare l'algoritmo di Floyd&Warshall rispetto ad un'applicazione ripetuta dell'algoritmo di Dijkstra con heap binari?
   *a) $m = \omega(n^2/\log n)$ b) $m = \Theta(n)$ c) per ogni valore di m d) per nessun valore di m
10. Sia dato un grafo pesato $G = (V, E)$ con n nodi ed m archi, e si consideri il problema di trovare il minimo albero ricoprente di G. Quando è strettamente conveniente (asintoticamente) l'implementazione di Prim con heap di Fibonacci rispetto all'algoritmo di Borůvka?
    *a) $m = \omega(n)$ b) $m = \Theta(n)$ c) per ogni valore di m d) per nessun valore di m

Griglia Risposte

	Domanda									
Risposta	1	2	3	4	5	6	7	8	9	10
a										
b										
c										
d										