Università degli Studi dell'Aquila

Prova di Esame del Corso di **Algoritmi e Strutture Dati con Laboratorio**

Martedì 3 Settembre 2013 – Proff. Guido Proietti e Giovanna Melideo

Scrivi i tuoi dati $\Longrightarrow$	Cognome: ..... Nome: ..... Matricola: .....

PROVA SCRITTA DI TEORIA: Domande a risposta multipla

Premessa: La prova è costituita da 10 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 finale della prova è ottenuto sommando i punti attribuiti e normalizzando su base 30. Se tale somma è negativa, verrà assegnato 0.

1. L'algoritmo di ordinamento crescente Insertion Sort applicato ad una sequenza di input ordinata in modo decrescente esegue un numero di confronti tra elementi pari a:
   a) $n-1$ b) $n(n+1)/2$ c) $n+1$ *d) $n(n-1)/2$
2. Siano $f(n)$ e $g(n)$ i costi dell'algoritmo Insertion Sort nel caso migliore e Quicksort in quello medio, rispettivamente. Quale delle seguenti relazioni asintotiche è vera:
   *a) $f(n) = o(g(n))$ b) $f(n) = \Theta(g(n))$ c) $f(n) = \omega(g(n))$ d) $f(n) = \Omega(g(n))$
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 ha complessità:
   a) $\Theta(n \log n)$ b) $\Omega(n \log n)$ c) $\omega(n)$ *d) $\Theta(n)$
5. Per $n = 2^k$, quale delle seguenti equazioni di ricorrenza descrive più precisamente la complessità dell'algoritmo Merge Sort:
   a) $T(n) = 2 \cdot T(n/2) + n, T(1) = \Theta(1)$ *b) $T(n) = 2 \cdot T(n/2) + \Theta(n), T(1) = 1$ c) $T(n) = 2 \cdot T(n/2) + O(n), T(1) = \Theta(1)$
   d) $T(n) = 2 \cdot T(n/2) + n, T(1) = 1$
6. Quale dei seguenti vettori rappresenta un heap binario:
   a) A=[10,9,4,7,6,5] *b) A=[20,16,9,15,12,1] c) A=[20,16,9,15,12,10] d) A=[5,6,4]
7. Un heap binomiale di 11 elementi è costituito dai seguenti alberi binomiali:
   *a) B_0, B_1, B_3 b) B_0, B_1, B_2 c) B_0, B_2, B_3 d) B_1, B_2, B_3
8. 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
9. L'algoritmo di Bellman e Ford applicato ad un grafo pesato con un numero di archi $m = \Theta(n)$, ha complessità:
   *a) $\Theta(n^2)$ b) $\Theta(n+m)$ c) $\Theta(n^3)$ d) $O(m \log n)$
10. Dato un grafo connesso di n nodi ed m archi, si supponga di voler trovare le distanze tra tutte le coppie di nodi del grafo. Per quale valore (asintotico) di m si ha che l'implementazione ripetuta dell'algoritmo di Dijkstra con heap di Fibonacci ha la stessa complessità temporale dell'implementazione di Floyd e Warshall (entrambi implementati utilizzando liste di adiacenza)?
    a) $m = \Theta(n)$ *b) $m = \Theta(n^2)$ c) $m = \Theta(n \log n)$ d) mai

Griglia Risposte

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