Università degli Studi dell'Aquila

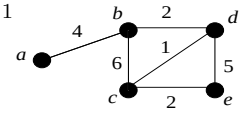
Prova di Teoria del corso di **Algoritmi e Strutture Dati con Laboratorio**

Martedì 21 Giugno 2011 – Prof. Guido Proietti

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

ESERCIZIO 1 (Teoria): Domande a risposta multipla

Premessa: Questa parte 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 relativo a questa parte è ottenuto sommando i punti ottenuti e normalizzando su base 30. Se tale somma è negativa, verrà assegnato 0.

1. Quale delle seguenti relazioni asintotiche è vera:
   a) $n \log^2 n = O(n \log n^2)$ b) $n = \Theta(4^{\log n})$ c) $2^{n+1} = \omega(2^n)$ *d) $n \log n^2 = \Theta(n \log n)$
2. 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) $O(n!)$
3. L'algoritmo di ordinamento non crescente Insertion Sort applicato ad una sequenza di input ordinata in modo non crescente esegue un numero di confronti tra elementi pari a:
   *a) $n-1$ b) n c) $n+1$ d) $n(n-1)/2$
4. Una coda di priorità realizzata con una lista lineare ordinata supporta l'estrazione del massimo in:
   *a) $O(1)$ b) $\Theta(n)$ c) $\Theta(\log n)$ d) $\Omega(\log n)$
5. In un albero AVL di n elementi, l'inserimento di un elemento nel caso peggiore induce un numero di rotazioni pari a:
   a) 0 *b) 2 c) $\Theta(\log n)$ d) 1
6. La visita in ampiezza del grafo  eseguita partendo dal nodo a genera un albero BFS di altezza pari a:
   a) 1 b) 2 *c) 3 d) 4
7. L'algoritmo di Bellman e Ford applicato ad un grafo pesato con un numero di archi $m = \Theta(n \log n)$, ha complessità:
   a) $\Theta(n^2)$ b) $\Theta(n+m)$ c) $\Theta(n^3)$ *d) $O(n^2 \log n)$
8. Usando gli alberi *QuickUnion* e l'euristica dell'unione pesata *by size*, il problema della gestione di n insiemi disgiunti sottoposti ad $n-1$ *Union* ed $m = n^2$ *Find* può essere risolto in:
   a) $\Theta(n)$ b) $\Theta(n+m)$ c) $\Theta(n^2)$ *d) $O(n^2 \log n)$
9. Il *minimo albero ricoprente* del grafo della domanda (6) ha peso totale:
   *a) 9 b) 13 c) 14 d) 5
10. Dato un grafo pesato con n vertici ed m archi, l'algoritmo di Boruvka ha una complessità pari a:
    a) $\Theta(m)$ b) $\Theta(n)$ 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
a										
b										
c										
d										