Università degli Studi dell'Aquila

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

Martedì 6 Luglio 2010 – Proff. Guido Proietti e Giovanna Melideo

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. L'algoritmo più efficiente per il calcolo dell' n-esimo numero della sequenza di Fibonacci ha complessità
   a) $\Omega(n)$ b) $\Theta(n)$ *c) $O(\log n)$ d) $\Theta(n \log n)$
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. Durante l'esecuzione del Quicksort, applicando la procedura di partizione *in loco* al vettore $[28, 47, 12, 98, 20, 6, 32]$, con perno l'elemento 28, si ottiene
   *a) $[20, 6, 12, 28, 98, 47, 32]$ b) $[12, 6, 20, 28, 98, 47, 32]$ c) $[6, 12, 20, 28, 47, 98, 32]$ d) $[6, 12, 20, 28, 32, 47, 98]$
5. Qual è la complessità spaziale dell'algoritmo Integer Sort applicato ad un array A di n elementi in cui $A[i] = 2i^2 + i$ per $i = 1, .., n$?
   a) $\Theta(n^3)$ b) $\Theta(n)$ *c) $O(n^2)$ d) $\Theta(n \log n)$
6. Dati due elementi u, v appartenenti ad un universo totalmente ordinato U, una funzione hash $h(\cdot)$ si dice *perfetta* se:
   a) $u = v \Rightarrow h(u) \neq h(v)$ b) $u \neq v \Rightarrow h(u) = h(v)$ c) $u = v \Rightarrow h(u) = h(v)$ *d) $u \neq v \Rightarrow h(u) \neq h(v)$
7. Qual è la distanza tra le stringhe *tredici* e *sedici*?
   a) 0 b) 1 *c) 2 d) 3
8. 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) $O(n^2 \log n)$
9. 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) $O(n \log n)$
10. 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) $O(n^2 \log n)$

Griglia Risposte

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