Università degli Studi di L'Aquila

Prima Prova Parziale di **Algoritmi e Strutture Dati con Laboratorio**

Mercoledì 23 Novembre 2011 – 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: 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 × la casella corrispondente alla risposta prescelta. È consentito omettere la risposta. In caso di errore, contornare con un cerchietto la × erroneamente apposta (ovvero, in questo modo ⊗) e rifare la × 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. Siano $f(n)$ e $g(n)$ i costi dell'algoritmo Insertion Sort nel caso medio e dell'algoritmo Selection Sort in quello migliore, rispettivamente. Quale delle seguenti relazioni asintotiche è falsa:
   *a) $f(n) = o(g(n))$ b) $f(n) = \Theta(g(n))$ c) $f(n) = O(g(n))$ d) $f(n) = \Omega(g(n))$
2. Quale delle seguenti funzioni $f(n)$ è $\Theta(n)$:
   a) $f(n) = n/\log n$ *b) $f(n) = n + \log n$ c) $f(n) = 1$ d) $f(n) = n^2$
3. Siano $f(n)$ e $g(n)$ i costi dell'algoritmo Merge Sort e dell'algoritmo Heap Sort, rispettivamente. Quale delle seguenti relazioni è vera:
   *a) $f(n) = \Theta(g(n))$ b) $f(n) = o(g(n))$ c) $f(n) = \omega(g(n))$ d) $f(n) = g(n)$
4. Un problema ha una delimitazione inferiore alla complessità temporale $\Omega(f(n))$ se:
   *a) Tutti gli algoritmi per la sua risoluzione hanno una delimitazione inferiore alla complessità computazionale pari a $\Omega(f(n))$
   b) Tutti gli algoritmi per la sua risoluzione hanno una delimitazione superiore alla complessità computazionale pari a $O(f(n))$
   c) Esiste un algoritmo per la sua risoluzione che ha una delimitazione inferiore alla complessità computazionale pari a $\Omega(f(n))$
   d) Esiste un algoritmo per la sua risoluzione che ha una delimitazione superiore alla complessità computazionale pari a $O(f(n))$
5. Dato un problema con una delimitazione inferiore alla complessità temporale pari a $\Omega(f(n))$, un algoritmo per la sua risoluzione si dice *ottimale* se ha un tempo di esecuzione $g(n)$ tale che:
   *a) $g(n) = \Theta(f(n))$ b) $g(n) = o(f(n))$ c) $g(n) = \omega(f(n))$ d) $g(n) = f(n)$
6. Durante l'esecuzione del Quicksort, applicando la procedura di partizione *in loco* al vettore $[23, 42, 7, 93, 15, 1, 27]$, con perno l'elemento 23, si ottiene
   *a) $[15, 1, 7, 23, 93, 42, 27]$ b) $[7, 1, 15, 23, 93, 42, 27]$ c) $[1, 7, 15, 23, 42, 93, 27]$ d) $[1, 7, 15, 23, 27, 42, 93]$
7. Qual è la complessità spaziale dell'algoritmo Integer Sort applicato ad un array A di n elementi in cui $A[i] = 3i^3 - 2i^2$ per $i = 1, .., n$?
   *a) $\Theta(n^3)$ b) $\Theta(n)$ c) $O(n^2)$ d) $\Theta(n \log n)$
8. La procedura *FixHeap* per il mantenimento di un heap, nel caso migliore costa:
   a) $\Theta(\log n)$ b) $\Theta(n)$ *c) $\Theta(1)$ d) $\Theta(n \log n)$
9. L'inserimento di un elemento in una coda di priorità di n elementi realizzata con un array ordinato costa:
   a) $\Theta(\log n)$ b) $O(\log n)$ c) $O(1)$ *d) $\Theta(n)$
10. Dato un albero binario di ricerca di n elementi, la cancellazione di un elemento restituisce un albero avente al massimo altezza:
    *a) $n - 2$ b) n c) $\Theta(\log n)$ d) $n - 1$

Griglia Risposte

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