# Università degli Studi dell'Aquila

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

Giovedì 17 Settembre 2015 – 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**

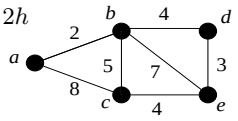
**Premessa:** La prova è costituita da 10 domande a risposta multipla. Per ciascuna domanda vengono fornite 4 risposte, di cui soltanto una è corretta. Per rispondere utilizzare la griglia annessa, barrando con una $\times$ la casella corrispondente alla risposta prescelta. È consentito omettere la risposta. 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 è dato dalla somma dei punti ottenuti. 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. Siano $f(n)$ e $g(n)$ i costi dell'algoritmo Insertion Sort (basato sul ciclo `while`) nel caso medio e 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) $f(n) = \Omega(g(n))$
3. 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 $O(f(n))$
4. Quale tra i seguenti è lo pseudocodice del Merge Sort (si supponga che la chiamata principale sia MergeSort$(A, 1, n)$)

   a) MergeSort$(A, i, f)$
   if $(i \geq f)$ then return
   $m = (i + f)/2$
   MergeSort$(A, i, m)$
   MergeSort$(A, m + 1, f)$
   Merge$(A, i, f, m)$

   b) MergeSort$(A, i, f)$
   if $(i \leq f)$ then return
   $m = (i + f)/2$
   MergeSort$(A, i, m)$
   MergeSort$(A, m + 1, f)$
   Merge$(A, i, m, f)$

   c) MergeSort$(A, i, f)$
   if $(i \geq f)$ then return
   $m = (i + f)/2$
   Merge$(A, i, m, f)$
   MergeSort$(A, i, m)$
   MergeSort$(A, m + 1, f)$

   *d) MergeSort$(A, i, f)$
   if $(i \geq f)$ then return
   $m = (i + f)/2$
   MergeSort$(A, i, m)$
   MergeSort$(A, m + 1, f)$
   Merge$(A, i, m, f)$
5. Siano $f(n)$ e $g(n)$ i costi degli algoritmi HeapSort e MergeSort nel caso migliore e in quello peggiore, rispettivamente. Quale delle seguenti relazioni asintotiche è vera:
   a) $g(n) = o(f(n))$ *b) $f(n) = \Theta(g(n))$ c) $f(n) = \omega(g(n))$ d) $g(n) = \omega(f(n))$
6. Come si esegue l'operazione `increaseKey`(elem $e$, chiave $\Delta$) di un elemento con chiave $k$ in un heap binomiale?
   a) eseguendo `insert`$(e, k + \Delta)$ b) eseguendo `decreaseKey`$(e, -\Delta)$
   c) eseguendo `delete`$(e)$ seguita da `insert`$(e, \Delta)$ *d) eseguendo `delete`$(e)$ seguita da `insert`$(e, k + \Delta)$
7. In un albero binario di ricerca di altezza $h$, il *predecessore* di un elemento può essere determinato, nel caso migliore, seguendo un cammino di lunghezza pari a:
   a) $h - 1$ b) $h$ *c) 1 d) $2h$
8. La visita in ampiezza del grafo 

   eseguita partendo dal nodo $b$ genera un albero BFS di altezza pari a:
   *a) 1 b) 2 c) 3 d) 4
9. Si consideri l'albero dei cammini minimi radicato in $c$ del grafo in figura 8. Qual è la massima distanza di un nodo dalla sorgente?
   a) 2 b) 5 *c) 7 d) 8
10. 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) $O(n \log n)$

## Griglia Risposte

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