



Scrivi i tuoi dati ⇒	Cognome:	Nome:	Matricola:	PUNTI
ESERCIZIO 1	Risposte Esatte:	Risposte Omesse:	Risposte Errate:	

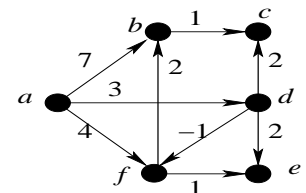
ESERCIZIO 1 (25 punti): Domande a risposta multipla

Premessa: Questa parte è costituita da 20 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. Quale dei seguenti insiemi rappresenta i primi sei numeri della sequenza di Fibonacci?
 a) {1, 2, 3, 5, 8, 13} b) {2, 3, 5, 8, 13, 21} *c) {1, 1, 2, 3, 5, 8} d) {1, 2, 3, 5, 7, 11}
2. Sia $f(n) = 6n + 5$; secondo la definizione della notazione asintotica O , per dimostrare che $f(n) = O(n^4)$, è sufficiente scegliere:
 a) $n_0 = 1, c = 2$ b) $n_0 = 2, c = 1$ c) $n_0 = 1, c = 1$ *d) $n_0 = 2, c = 2$
3. Quale delle seguenti relazioni asintotiche è falsa:
 a) $n \log n^3 = O(n \log n^2)$ b) $n \log n^2 = \Omega(n \log n)$ c) $n \log n^2 = \Theta(n \log n)$ *d) $n \log n^2 = \omega(n \log n)$
4. L'algoritmo di ricerca sequenziale in un array ordinato di n elementi nel caso medio ha complessità:
 *a) $\Theta(n)$ b) $O(1)$ c) $\Omega(\log n)$ d) $\Theta(\log n)$
5. Quale delle seguenti ricorrenze descrive la complessità dell'algoritmo di ricerca binaria in un array ordinato di n elementi?
 a) $T(n) = c + 2T(\lfloor n - 1/2 \rfloor)$ se $n > 1$, $T(1) = 1$ se $n = 1$ b) $T(n) = T(\lfloor n - 1/2 \rfloor)$ se $n > 1$, $T(1) = 1$ se $n = 1$
 *c) $T(n) = c + T(\lfloor n - 1/2 \rfloor)$ se $n > 1$, $T(1) = 1$ se $n = 1$ d) $T(n) = \log n + T(\lfloor n - 1/2 \rfloor)$ se $n > 1$, $T(1) = 1$ se $n = 1$
6. 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) $n(n - 1)/2$
7. L'algoritmo SELECTION SORT, nel caso migliore costa:
 a) $o(n^2)$ *b) $\Theta(n^2)$ c) $O(n \log n)$ d) $\Theta(n \log n)$
8. A quale delle seguenti classi appartiene la complessità del caso medio dell'algoritmo QUICKSORT:
 a) $\Theta(n^2)$ *b) $\Theta(n \log n)$ c) $O(n)$ d) $\Omega(n^2)$
9. La delimitazione inferiore al problema della ricerca di un elemento in un insieme non ordinato di n elementi è:
 a) $\Theta(\log n)$ b) $\Theta(n \log n)$ *c) $\Omega(n)$ d) $\Omega(n \log n)$
10. In un albero binario di ricerca di altezza h , il *successore* di un elemento può essere determinato in:
 a) $\Theta(\log h)$ b) $O(\log h)$ c) $\Theta(1)$ *d) $O(h)$
11. In un albero AVL di n elementi, l'inserimento di un elemento, nel caso migliore, ha complessità:
 *a) $O(\log n)$ b) $\Omega(n)$ c) $\Theta(n)$ d) $\Theta(1)$
12. In un heap binomiale di n elementi, la ricerca del minimo ha complessità:
 *a) $O(\log n)$ b) $\Omega(n)$ c) $\Theta(n)$ d) $\Theta(1)$
13. In una tavola ad accesso diretto con un fattore di carico $\alpha = 0, 1$, l'inserimento di un elemento costa:
 a) $O(\alpha)$ b) $\Omega(n)$ c) $\Theta(\log n)$ *d) $\Theta(1)$
14. Dato il grafo  qual è il grado dell'albero dei cammini minimi radicato in c ?
 a) 1 *b) 2 c) 3 d) 4
15. Dato un grafo pesato con n vertici ed m archi, l'algoritmo di Bellman e Ford ha complessità:
 a) $\Theta(n^2)$ b) $\Theta(n + m)$ c) $\Theta(n^3)$ *d) $O(nm)$
16. Dato un grafo pesato con n vertici ed m archi, l'algoritmo di Dijkstra realizzato con heap di Fibonacci ha complessità:
 a) $\Theta(n^2 \log n)$ *b) $\Theta(m + n \log n)$ c) $\Theta(n^2)$ d) $O(n \log n)$
17. Dato un grafo pesato con n vertici ed m archi, l'algoritmo di Floyd e Warshall ha complessità:
 a) $\Theta(n^2)$ b) $\Theta(n + m)$ *c) $O(n^3)$ d) $O(nm)$
18. 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) $O(n^2 \log n)$
19. Dato un grafo completo con n vertici, l'algoritmo di Boruvka ha una complessità pari a:
 a) $\Theta(m)$ b) $\Theta(n)$ c) $\Theta(m + n \log n)$ *d) $\Theta(n^2 \log n)$
20. Dato un grafo pesato con n vertici ed m archi, l'algoritmo di Kruskal esegue un numero di operazioni $\text{FIND}(u)$ pari a:
 a) m b) $\Theta(n)$ c) $\Theta(m \log n)$ *d) $\Theta(m)$

Griglia Risposte

Risposta	Domanda																				
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
a																					
b																					
c																					
d																					



ESERCIZIO 2 (5 punti) (Da svolgere sul retro della pagina!)

Mostrare l'intera esecuzione, passo per passo, dell'algoritmo di Bellman e Ford per determinare i cammini minimi con sorgente in a sul seguente grafo (gli archi vanno considerati in ordine lessicografico):