

n. quesito	1	2	3	4	tot
punti	8	8	9	8	33

quesito 1

Calcolare la complessità del seguente comando in funzione del numero di nodi dell'albero binario quasi bilanciato t:

```
for (int i=0; i <= g(t)*f(t); i++) cout << g(t);
```

Indicare per esteso le relazioni di ricorrenza.

<pre>int g (Node* t) { if (!t) return 1; t->label+= f(t); cout << g (t->left); return 1+2*g(t->left); }</pre>	<pre>int f(Node* t) { if (!t) return 1; cout << f(t->left); return 1 + 4*f(t->right); }</pre>
---	---

Funzione f

$$T_f(0) = d$$

$$T_f(n) = c + 2T_f(n/2) \quad T_f(n) \text{ è } O(n)$$

$$R_f(0) = 1$$

$$R_f(n) = 1 + 4R_f(n/2) \quad R_f(n) \text{ è } O(n^2)$$

Funzione g

$$T_g(0) = d$$

$$T_g(n) = cn + 2 T_g(n/2) \quad T_g(n) \text{ è } O(n \log n)$$

$$R_g(0) = d$$

$$R_g(n) = c + 2 R_g(n/2) \quad R_g(n) \text{ è } O(n)$$

Calcolo del comando:

$$\text{numero iterazioni: } R_f(n) * R_g(n) = O(n^2) * O(n) = O(n^3)$$

$$\text{complessità della singola iterazione: } T_f(n) + T_g(n) = O(n) + O(n \log n) = O(n \log n)$$

$$\text{complessità del for: } O(n^4 \log n)$$

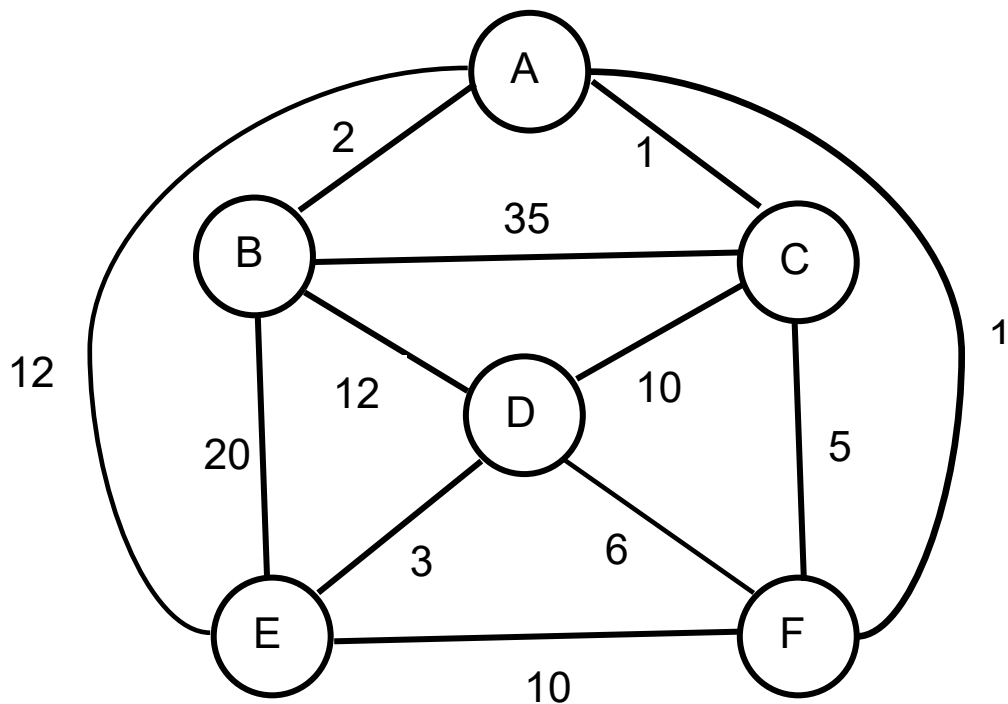
quesito 2

- a) 1 Dare la definizione di albero binario.
- b) 1 Specificare la differenza fra albero generico e albero binario.
- c) 2 Descrivere la corrispondenza fra le visite di un albero generico e quelle del suo trasformato binario.
- d) 4 Scrivere una funzione, che, dato un albero binario e un intero x, conta il numero di sottoalberi che hanno esattamente x nodi.

```
int conta (Node* t, int x, int & nodes) {  
    if (!t) { nodes =0; return 0;}  
    int nodes_l, nodes_r;  
    int conta_l= conta(t->left, nodes_l);  
    int conta_r= conta(t->right, nodes_r);  
    nodes=nodes_l+ nodes_r +1;  
    return conta_l + conta_r+ (x == nodes);  
}
```

quesito 3

- 1 A cosa serve l'algoritmo di Dijkstra?
- 3 Qual è la sua complessità e come si calcola?
- 1 Si può applicare a tutti grafi?
- 4 Applicarlo al grafo in figura a partire dal nodo E. Mostrare tutti i passaggi in una tabella fatta come nel disegno. Indicare i cammini minimi.



Insieme Q	Nodo scelto	A	B	C	D	E	F
	-	inf -	inf -	inf -	inf -	0 -	inf -
A,B,C,D,E,F	E	12 E	20 E	inf -	3 E	0 -	10 E
A,B,C,D,F	D	12 E	15 D	13 D	3 E	0 -	9 D
A,B,C,F	F	10 F	15 D	13 D	3 E	0 -	9 D
A,B,C	A	10 F	12 A	11 A	3 E	0 -	9 D
B,C	C	10 F	12 A	11 A	3 E	0 -	9 D
B		10 F	12 A	11 A	3 E	0 -	9 D

Cammini minimi: EDFA, EDFAB, EDFAC, ED, E, EDF

quesito 4

- a) 3 Descrivere brevemente tutti gli algoritmi di **ricerca** visti a lezione indicando la struttura a cui sono applicati e la complessità nel caso medio e nel caso peggiore.
- b) 2 Descrivere l'algoritmo di ordinamento **mergesort** e simularlo su sulla lista [5,8,1,2,9,4]mostrando tutte le chiamate ricorsive a mergesort.
- c) 3 Quali sono le regole per gli specificatori di accesso **private, public e protected** (assumere la derivazione pubblica)? Quale di questi specificatori non ha significato se non si ha ereditarietà fra classi?