

Cognome e Nome:
Numero di Matricola:

Spazio riservato alla correzione

1	2	3	4	5	Totale
/18	/20	/22	/18	22/	/100

Si ricorda che per i punti che richiedono l'analisi di un algoritmo occorre fornire un limite superiore asintotico quanto migliore e' possibile al tempo di esecuzione dell'algoritmo giustificando la risposta.

1. Analisi degli algoritmi e notazione asintotica

a) Indicare quali delle seguenti affermazioni sono vere e quali sono false.

1. $n(\log n)^4 + 1000n^2 = O(n^4)$
2. $n^n = \Omega(n^{3/4} n^{1/4})$
3. $n^{1/2} = O(\log n)$
4. $n^3 + 1000n^2 + 100 = \Theta(n^3)$
5. $\log(\log n) = \Omega((\log n)^{1/2})$

b) Si dimostri che se $0 < f(n) = O(h(n))$ e $0 < g(n) = O(p(n))$ allora $f(n)g(n) = O(h(n)p(n))$. Occorre utilizzare solo la definizione di O e nessuna altra proprieta`.

c) Si analizzi il tempo di esecuzione nel caso pessimo del seguente segmento di codice fornendo una stima asintotica **quanto migliore e' possibile** per esso. **Si giustifichi in modo chiaro la risposta.**

```
FOR(i=1; i<n; i=i+1){  
  FOR(j=1; j<2^n; j=j*2) {  
    print(j);  
  }  
}
```

2. Divide et Impera

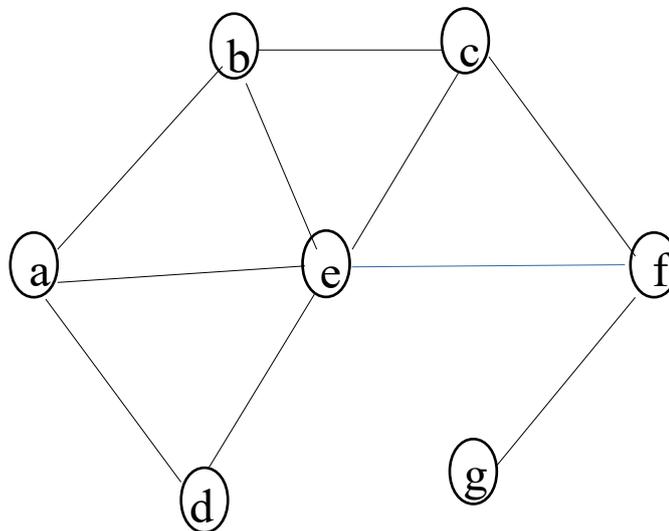
- a) Si scriva lo pseudocodice di un algoritmo ricorsivo **cerca** che restituisce il numero di occorrenze di x in un array A . L'elemento x e l'array A vengono passati in input all'algoritmo. Ad esempio, se l'array A contiene $\langle 3\ 1\ 4\ 1\ 2\ 3\ 1\ 4\ 5\ 2\ 1\ 4\ 5\ 1 \rangle$ e $x=4$ allora l'algoritmo restituisce 3.

Se non si è in grado di scrivere l'algoritmo richiesto, si scriva lo pseudocodice dell'algoritmo ricorsivo **ricercaBinaria** che effettua la ricerca binaria in un array ordinato. **Il punteggio massimo per l'algoritmo ricercaBinaria è la metà di quello per l'algoritmo cerca.**

- b) Si fornisca la relazione di ricorrenza che esprime un limite superiore al tempo di esecuzione dell'algoritmo da voi fornito al punto a) .
- c) A partire dalla relazione di ricorrenza da voi fornita al punto b), si fornisca una funzione $h(n)$ tale $T(n)=O(h(n))$. Giustificare la risposta usando o il metodo iterativo o quello della sostituzione (induzione).

3. Grafi

- a) Si scriva lo pseudocodice **dell'algoritmo di Prim** per il minimo albero ricoprente aggiungendo anche le linee di codice per la costruzione dell'albero. Si analizzi il tempo di esecuzione dell'algoritmo proposto quando la coda a priorit     implementata con un heap binario.
- a) Si disegni l'albero DFS generato da una visita DFS del seguente grafo a partire dal nodo sorgente **a**. Si assuma che i nodi siano disposti nelle liste di adiacenza in base all'ordine crescente delle proprie etichette.



- b) Si definisca il concetto di ordinamento topologico di un grafo direzionato e si dimostri che se un grafo direzionato G   un DAG allora G ha un ordinamento topologico.

4. Algoritmi greedy

- a) Si spieghi in che cosa consiste un'istanza (input) del problema dell'interval scheduling e in cosa consiste una soluzione (output) del problema. Se dalla risposta a questo punto si evincerà che lo studente non sa in cosa consiste il problema dell'interval scheduling, i punti successivi dell'esercizio non saranno valutati.

- b) Si fornisca un'istanza del problema dell'interval scheduling con $n=6$ per cui il valore della soluzione ottima è 3. Si specifichino i valori numerici che descrivono l'input. **Non è sufficiente fornire un disegno che descriva l'input.**

- c) Si scriva lo pseudocodice dell'algoritmo greedy che restituisce la soluzione ottima per il problema dell'interval scheduling.

5. **Programmazione dinamica**

- a) Fornire una formula per il calcolo del valore della soluzione ottima OPT del problema subset sums in termini di valori delle soluzioni ottime per sottoproblemi di taglia piu' piccola. Spiegare **in modo chiaro**
1. cosa rappresenta la funzione OPT e cosa rappresentando i suoi parametri
 2. come si arriva alla formula da voi fornita.
- b) Scrivere l'algoritmo di programmazione dinamica per subset sums. Questo punto` sara` valutato solo se dalla risposta al punto a) si evincera` che lo studente sa in cosa consiste il problema di subset sums.
- c) Si forniscano i valori $p(j)$ per la seguente istanza del problema dell'Interval Scheduling Pesato.

$s_1=7$	$f_1=9$	$w(1)=4$
$s_2=1$	$f_2=4$	$w(2)=3$
$s_3=5$	$f_3=6$	$w(3)=6$
$s_4=4$	$f_4=8$	$w(4)=1$

Attenzione: gli indici j di $p(j)$ non corrispondono necessariamente agli indici j dei valori input s_j , f_j e w_j .