

## *Algoritmi e Strutture Dati - 28/01/13*

### **Esercizio 1 – Punti $\geq 6$**

Supponendo che il caso base sia  $O(1)$  si calcoli l'andamento asintotico delle seguenti equazioni di ricorrenza:

1.  $A(n) = 4A(n/2) + n^2 \log n$ .
2.  $B(n) = 4B(n/2) + n^2$ .
3.  $C(n) = nC(n-1)$ .

### **Esercizio 2 – Punti $\geq 6$**

Scrivere un algoritmo efficiente che, dato in input un albero binario  $T$ , restituisce **true** se e solo se  $T$  rappresenta un albero binario completo

### **Esercizio 3 – Punti $\geq 9$**

Si consideri un grafo non orientato  $G = (V, E)$  in cui a ciascun nodo  $v \in V$  è associato un peso reale  $w(v)$  (che può essere positivo o negativo). Un cammino  $\langle v_1, v_2, \dots, v_k \rangle$  si dice monotono se  $w(v_1) < w(v_2) < \dots < w(v_k)$ . In altre parole in un cammino monotono i pesi dei nodi attraversati devono essere in ordine strettamente crescente.

1. Dimostrare che se  $\langle v_1, v_2, \dots, v_k \rangle$  è un cammino monotono, allora è aciclico
2. Descrivere un algoritmo efficiente che, dato in input un grafo non orientato  $G = (V, E)$  con nodi pesati, e due nodi  $s, d \in V$ , restituisce **true** se e solo se esiste un cammino monotono che inizia dalla sorgente  $s$  e termina nella destinazione  $d$ . L'algoritmo deve anche stampare i nodi che compongono tale cammino (i nodi possono essere stampati nell'ordine  $v_1, v_2, \dots, v_k$  oppure nell'ordine inverso  $v_k, v_{k-1}, \dots, v_1$ )
3. Determinare il costo computazionale dell'algoritmo descritto al punto 2, motivando la risposta

### **Esercizio 4 – Punti $\geq 12$**

Siano dati un intero positivo  $T$  ed un insieme di interi positivi  $A = \{a_1, \dots, a_k\}$ . Progettare un algoritmo di Programmazione Dinamica che ritorni il valore **true** se esiste o meno un sottoinsieme  $B \subseteq A$  tale che  $T = \sum_{b \in B} b$ , il valore **false** nel caso contrario.