

# “Algoritmi e Strutture Dati”

15 Gennaio 2008

1. Tempo disponibile 180 minuti. È ammesso ritirarsi entro 90 minuti.
2. Si possono consegnare al più 3 scritti tra Giugno 2007 e Febbraio 2008.
3. Non è possibile consultare appunti, libri o persone, né uscire dall'aula.
4. Le soluzioni degli esercizi devono:
  - a. spiegare a parole l'algoritmo usato (anche con eventuali disegni)
  - b. commentare l'eventuale procedura Pascal (dettagliando il significato delle variabili)
  - c. giustificare la correttezza e tutti i passaggi matematici
  - d. dimostrare la complessità (con equazioni di ricorrenza se necessario)

**1.** Si scriva una procedura Pascal che riceva come parametro d'ingresso un vettore di  $n$  elementi e tale che il suo tempo di esecuzione  $T(n)$  verifichi la ricorrenza:

$$\begin{aligned} T(n) &= c && \text{per } n < 7 \\ T(n) &= 3T(n/3) + dn && \text{altrimenti} \end{aligned}$$

con  $c$  e  $d$  costanti. Si calcoli anche l'ordine di grandezza di  $T(n)$  risolvendo la ricorrenza.

**2.** Sia dato un albero binario  $A$  implementato con puntatori, in cui ogni nodo contiene un valore intero. Scrivere una funzione Pascal di *complessità ottima* che modifichi  $A$  aggiungendo ad ogni foglia, che contenga un valore pari, un figlio sinistro contenente 0.

**3.** Data una lista  $L$  di interi, si vuole modificarla duplicando tutti gli elementi divisibili per 5 mantenendo lo stesso ordine che gli elementi avevano in  $L$  (p.e. se l'ingresso è  $L = 5, 1, 10, 4$ , allora il risultato è  $L = 5, 5, 1, 10, 10, 4$ ). Si scriva una procedura Pascal di complessità ottima *utilizzando gli operatori* per le liste visti a lezione..

**4.** Si scriva la procedura Pascal *Depth-First Search (DFS)* vista a lezione. Si esegua la procedura DFS sul grafo *non orientato*  $G = (N, A)$ ,  $N = \{1, 2, 3, 4, 5\}$ ,  $A = \{[1,2], [1,3], [1,5], [2,5], [3,4], [3,5]\}$  a partire dal nodo 2, assumendo che i vettori di adiacenza siano ordinati in modo *crescente* e mostrando il contenuto dei vettori di adiacenza.

**5.** Dato un vettore  $a_1, \dots, a_n$  di  $n$  interi, si vuole calcolarne la sommatoria  $a_1 + \dots + a_n$ . Si scriva una procedura (o funzione) Pascal efficiente che utilizzi la tecnica *divide-et-impera con partizione bilanciata dei dati*, e se ne analizzi la complessità impostando e resolvendo le opportune relazioni di ricorrenza.

**6.** Dati un insieme  $A$  di interi ed un intero  $k$ , si vuole decidere se esiste una partizione di  $A$  in tre sottoinsiemi  $B$ ,  $C$  e  $D$  tali che la somma degli elementi di  $B$  sia  $k$  e la somma degli elementi di  $C$  sia uguale al prodotto degli elementi di  $D$ . Si scriva una procedura Pascal *non deterministica* di complessità polinomiale.