Algoritmi e Strutture Dati
14 Gennaio 2003

- *Tempo disponibile 180 minuti (è ammesso ritirarsi entro 90 minuti)*
- *Sono ammessi al più 3 scritti consegnati per A.A.*
- *Non è possibile consultare appunti, libri o persone*
- *Non è possibile uscire dall'aula durante la prova scritta*
- *Le soluzioni degli esercizi devono:*
  - *Spiegare a parole l'algoritmo usato (anche con eventuali disegni)*
  - *commentare l'eventuale procedura Pascal (dettagliando il significato delle variabili)*
  - *giustificare la correttezza*
  - *dimostrare la complessità (impostando e risolvendo equazione di ricorrenza se necessario)*
  - *giustificare tutti i passaggi matematici*

1. Si valuti l'ordine di grandezza della complessità $T(n)$ della seguente funzione Pascal:

```
function MISTERO(n: integer): integer;
   var j,k: integer;
   begin
      for j := 1  to  n  do  k := n * j;
      if n > 44  then
         MISTERO := ( 3 * MISTERO(n - 2) ) div 2
      else
         MISTERO := 1;
   end;
```

2. Si consideri una nuova operazione delle liste che, data una lista L di interi, la modifica cancellando tutti gli elementi pari da L e copiando tutti quelli dispari in un'altra lista M, mantenendo sia in L che in M l'ordine che gli elementi avevano originariamente in L (p.e. se l'ingresso è L = 4, 1, 6, 3, allora il risultato è L = 1, 3 ed M = ~~4, 6~~ 1, 3). Si scriva una procedura Pascal di *complessità ottima* utilizzando gli operatori per le liste visti a lezione.

3. Si consideri una nuova operazione degli alberi binari che, dato un albero binario T contenente elementi interi, lo modifica cancellando ogni foglia che contiene un elemento pari. Si scriva una procedura Pascal di complessità ottima assumendo che l'albero sia *realizzato con puntatori*.

4. Si riscriva la procedura Pascal *Breadth-First Search (BFS)* vista a lezione e la si esegua sul grafo $G=(N,A)$, $N=\{1,2,3,4,5\}$, $A=\{(1,2),(1,3),(1,5),(3,5),(3,4),(4,1),(4,2),(5,2)\}$ a partire dal nodo 4. Si disegni il grafo e si mostrino la memorizzazione di G con vettori di adiacenza e l'ordine di visita dei nodi e degli archi (assumendo che i vettori di adiacenza siano ordinati in modo *crescente*).

5. Nel problema delle SOMME PREFISSE, data una sequenza $a_1,\ldots,a_n$ di interi, si vuole calcolare la sequenza $b_1,\ldots,b_n$ dove b_i è uguale alla somma $a_1+\ldots+a_i$. Si scriva una procedura Pascal che utilizzi la tecnica *divide-et-impera con partizione bilanciata dei dati*, assumendo che le sequenze siano realizzate con *vettori*, e se ne analizzi la complessità.

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.