

Algoritmi e Strutture Dati

16 Settembre 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, né uscire dall'aula*
- *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 e tutti i passaggi matematici*
 - *dimostrare la complessità (con equazioni di ricorrenza se necessario)*
- *Nella valutazione dello scritto ogni esercizio conta 6 punti (e quindi si raggiunge 18 con 3 esercizi risolti correttamente e 30 con 5 esercizi risolti correttamente)*

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

```
function MISTERO( $n$ : integer): integer;  
  var  $j, k, m$ : integer;  
  begin  
     $m := n$ ;  
    while  $m > 1$  do begin  
       $k := \lfloor \frac{m}{2} \rfloor * j$ ;  
       $m := m - 1$   
    end;  
    if  $n > 13$  then  
      MISTERO :=  $2 * \text{MISTERO}(n - 1) + 2$   
    else  
      MISTERO := 1;  
  end;
```

2. Sia dato un vettore ordinato $V[1..n]$ contenente n elementi interi distinti appartenenti all'intervallo $1..n+1$. Si scriva una procedura Pascal, basata sulla ricerca binaria, per individuare in tempo $O(\log n)$ l'unico intero dell'intervallo $1..n+1$ che non compare in V .

3. Si indichi il contenuto di una tabella hash di dimensione 11, inizialmente vuota, dopo l'inserzione, nell'ordine, delle chiavi: M, A, R, I, E, M, O, N, T, I. Si usi la funzione hash $H(k) = k \bmod 11$ per la k -esima lettera dell'alfabeto italiano e il metodo di scansione lineare a passo unitario. Si indichi successivamente il contenuto della tabella dopo avervi cancellato, nell'ordine: E, S, A, M, E, e poi inserito, nell'ordine: V, O, T, O.

4. Si scriva la procedura Pascal *Breadth-First Search (DFS)* vista a lezione. Si esegua la procedura 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, *assumendo* che i vettori di adiacenza siano ordinati in modo *decescente* e mostrando il contenuto dei vettori di adiacenza.

5. Siano date due stringhe $P = p_1p_2\cdots p_m$ e $T = t_1t_2\cdots t_n$ di caratteri alfabetici. Si progetti un algoritmo di "programmazione dinamica" per individuare la piú lunga sottosequenza comune tra P e T (per esempio, se $P = 9, \underline{15}, 3, \underline{6}, 4, \underline{2}, 5, 10, \underline{3}$ e $T = 8, \underline{15}, \underline{6}, 7, 9, \underline{2}, 11, \underline{3}, 1$ allora il risultato è: 15, 6, 2, 3).

6. Si scriva una procedura Pascal di complessità *ottima* che, data una lista L di interi, la riordina in modo che tutti gli elementi dispari precedano, nello stesso ordine che avevano inizialmente in L , tutti gli elementi pari (p.e., se $L = 3, 7, 8, 1, 4$, si ottiene $L = 3, 7, 1, 8, 4$).