

Laurea in “Scienze di Internet”
Corso di “Algoritmi e Strutture Dati”
Appello del 12 Luglio 2004

1. *Tempo disponibile 180 minuti (è ammesso ritirarsi entro 90 minuti)*
2. *Sono ammessi al più 3 scritti consegnati per A.A.*
3. *Non è possibile consultare appunti, libri o persone, né uscire dall’aula*
4. *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)*
5. *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 valuti l’ordine di grandezza della complessità $T(n)$ della seguente funzione Pascal:

```
function PIPPO(n: integer): integer;  
  var j, k: integer;  
  begin  
    for j := 11 downto 4 do k := n * j;  
    if n > 17 then  
      PIPPO := ( 3 * PIPPO(n div j) + PIPPO(n div 4) ) div 3  
    else  
      PIPPO := 13;  
  end;
```

2. Dato un intero n , si vogliono stampare tutte le potenze di 3 minori o uguali ad n . Si scriva una procedura Pascal di complessità ottima (Nota: giustificare accuratamente la complessità).

3. Data una lista L di interi, si vuole cancellare gli elementi pari da L e copiare quelli divisibili per 4 in un'altra lista M , mantenendo in entrambe le liste l'ordine che gli elementi avevano originariamente in L (p.e. se l'input è $L = \{1, 6, 3\}$, allora l'output è $L = 1, 3$ e $M = 4$). Si scriva una procedura Pascal di complessità *ottima* assumendo che la lista sia realizzata con *puntatori*.

4. Si scriva la procedura Pascal *Breadth-First-Search (BFS)* vista a lezione. Si esegua la procedura BFS 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 1, assumendo che i vettori di adiacenza siano ordinati in modo *crescente* e mostrando il contenuto dei vettori di adiacenza.

5. Si scriva la procedura HEAPSORT vista a lezione. La si esegua (a mano) per ordinare alfabeticamente i 6 elementi: O, R, D, I, N, A, illustrando il contenuto dello heap durante ogni passo dell'esecuzione.

6. Si indichi il contenuto di una tabella hash di dimensione 11, inizialmente vuota, dopo l'inserzione, nell'ordine, delle chiavi: C, H, E, F, I, G, U, R, A. Si usi la funzione hash $H(k) = k \bmod 11$ per la k -esima lettera dell'alfabeto italiano e il metodo di scansione lineare con passo $h=2$. Si indichi successivamente il contenuto della tabella dopo avervi cancellato, nell'ordine: C, A, S, S, A, N, O, e poi inserito, nell'ordine: T, O, T, T, I.