

# Laurea in “Informatica”

## Corso di “Algoritmi e Strutture Dati”

### 17 Febbraio 2007

1. Tempo disponibile 180 minuti. È ammesso ritirarsi entro 90 minuti.
2. Sono ammessi al più 3 scritti consegnati tra Giugno 2006 e Febbraio 2007.
3. Non è possibile consultare appunti, libri o persone, né uscire dall'aula.
4. Per raggiungere la sufficienza occorrono almeno 3 esercizi risolti senza alcun errore.
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 la complessità  $T(n)$  della seguente funzione Pascal:

```
function PIPPO(n: integer): integer;  
  var i, j: integer;  
  begin  
    for i := 0 to 1 do j := 2;  
    if n > 10 then  
      PIPPO := 5*PIPPO(n div 2) + PIPPO(n div j) + j*n div 2  
    else if n > 2 then  
      PIPPO := 5*PIPPO(n div 2) + PIPPO(n - j) + j*n div 2  
    else  
      PIPPO := 1  
  end;
```

2. Dato un albero binario  $A$  implementato con *puntatori*, in cui ogni nodo contiene un intero, scrivere una funzione Pascal di *complessità ottima* che modifichi  $A$  cancellando ogni foglia che è un figlio sinistro e contiene un elemento pari.

3. Data una lista  $L$  di interi, si vuole togliere da  $L$  ogni elemento pari e inserirlo in una nuova lista  $M$ , mantenendo in entrambe le liste l'ordine originario degli elementi (p.e. se in input  $L = 2, 5, 1, 4, 5, 7, 4$ , allora in output  $L = 5, 1, 5, 7$  e  $M = 2, 4, 4$ ). Si scriva una procedura Pascal efficiente utilizzando gli *operatori* delle liste visti a lezione.

4. Si scriva la procedura Pascal per la visita *DFS* e la si esegua (a mano) sul grafo  $G = (N, A)$  a partire dal nodo 2, dove  $N = \{1, 2, 3, 4\}$ ,  $A = \{(1,3), (1,4), (2,1), (2,4), (3,4)\}$ , dopo aver mostrato la memorizzazione di  $G$  con vettori di adiacenza.

5. Dato un vettore ordinato  $V[1..n]$  contenente  $n$  elementi interi distinti appartenenti all'intervallo  $1..n+1$ , si vuole trovare l'unico intero dell'intervallo  $1..n+1$  che non compare in  $V$ . Si scriva una procedura Pascal, basata sulla ricerca binaria, che richieda tempo  $O(\log n)$ .

**6.** Descrivere la tecnica GREEDY per la progettazione di algoritmi, illustrando inoltre un esempio di algoritmo basato su tale tecnica.