

Laurea in “Informatica”

Corso di “Algoritmi e Strutture Dati”

20 Gennaio 2010

1. Tempo disponibile 180 minuti. È ammesso ritirarsi entro 90 minuti.
2. Sono ammessi al più 3 scritti consegnati per l'A.A. 2008/09 (Giugno 2009 - Febbraio 2010)
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. Data una lista L di n interi positivi distinti, si vuole modificarla cancellando ogni elemento che vale il doppio del precedente, lasciando però inalterato l'ordine relativo tra gli elementi (p.e. se in ingresso $L = 2, 4, 8, 5, 3, 6, 7, 14, 28$, allora in uscita si vuole $L = 2, 5, 3, 7$). Si scriva una procedura Pascal utilizzando gli operatori per le liste visti a lezione.

2. Siano dati un albero binario T contenente elementi interi positivi ed un intero k . Si vuole modificare T cancellando ogni foglia che è figlia di un nodo che contiene k . Si scriva una procedura Pascal di complessità ottima assumendo che l'albero sia realizzato con puntatori.

3. Si valuti l'ordine di grandezza della complessità $T(n)$ della funzione PIPPO:

```
function PIPPO(n: integer): integer;
  var i, j: integer;
begin
  for j := 2 to n div 4 do i := j;
  if n ≤ 10 then PIPPO := 2
    else if n > 31 then PIPPO := 5*PIPPO(i) + n*i
      else PIPPO := 7*PIPPO(n-2) + 5*i
end;
```

4. Siano dati un intero k e due vettori A e B , ciascuno di n interi, tali che i $2n$ interi siano tutti distinti. Si vuole decidere se esistono due elementi, uno appartenente al vettore A e l'altro appartenente al vettore B , tali che la loro somma sia uguale a k . Si scriva una procedura Pascal di complessità $O(n \log n)$.

5. Sia dato un grafo non orientato connesso tale che ogni arco è etichettato 0 oppure 1. Si scriva una procedura Pascal di complessità ottima per verificare se il sotto-grafo formato dagli archi etichettati 1 è connesso oppure no.

6. Si scriva un algoritmo non deterministico di complessita' polinomiale per risolvere il problema del COMMESSE VIAGGIATORE (Date n citta', le distanze tra esse ed un intero k , e' possibile partire da una citta', attraversare ogni citta' una e una sola volta, e ritornare alla citta' di partenza, percorrendo una distanza complessiva non superiore a k ?).