

**Laurea Magistrale in "Informatica"**  
**Corso di "Algoritmi Avanzati"**  
**27 Gennaio 2015**

1. *Tempo disponibile 180 minuti.*
2. *Non è possibile consultare appunti, libri o persone, né uscire dall'aula.*
3. *Le soluzioni degli esercizi devono:*
  - *spiegare a parole l'algoritmo usato (anche con l'aiuto di esempi o disegni)*
  - *fornire e commentare lo pseudo-codice (dettagliando il significato delle variabili)*
  - *giustificare correttezza e complessità (con tutti i passaggi matematici necessari)*
4. *Un esercizio può ammettere più soluzioni: a soluzioni computazionalmente più efficienti e/o concettualmente più semplici sono assegnati punteggi maggiori.*

**1 (PRAM).** Data una sequenza  $S$  di  $n$  interi relativi distinti, si vuole riordinarla in modo che tutti gli elementi negativi o nulli precedano tutti quelli positivi, mantenendo l'ordine iniziale sia tra gli elementi negativi o nulli sia tra gli elementi positivi (per esempio, se la sequenza in input è  $S = -3, 2, -4, 0, 1, 6, -7, -4$ , allora si vuole ottenere in output  $S = -3, -4, 0, -7, -4, 2, 1, 6$ ). Si progetti un algoritmo *efficiente* per una EREW-PRAM (che richieda tempo poli-logaritmico in  $n$  utilizzando un numero di processori polinomiale in  $n$ ).

**2 (Reti a grado limitato).** Si descriva l'algoritmo Bitonic Merge Sort visto a lezione, sia sul modello PRAM che su un ipercubo, analizzandone la complessità.

**3 (Concorrenza).** Sia  $A$  un vettore *ordinato* di  $n$  valori interi (non necessariamente distinti) e si consideri il problema di individuare un valore che compare il massimo numero di volte. Per esempio, se  $A = 1, 2, 2, 3, 4$ , allora la risposta è 2. Proponi un algoritmo concorrente efficiente per risolvere il problema usando  $p \ll n$  processori, scriverne lo pseudocodice e discuterne la complessità (*attenzione*: un valore potrebbe apparire più di  $n/p$  volte e, al limite, anche fino ad  $n$  volte, come nel caso di  $A = 2, 2, 2, 2, 2$ ).

**4 (Algoritmi distribuiti).** Si descriva (a parole e con disegni) l'algoritmo distribuito di Gallager-Humblet-Spira per calcolare il minimo albero di copertura di un grafo non orientato e pesato, giustificandone la complessità.

**5 (VLSI - SOLO PER GLI IMMATRICOLATI PRIMA DELL'A.A. 2013/14 SPROVVISTI DEL PROGETTO DEL MODULO II).** Si spieghi il funzionamento dell'algoritmo di Preparata-Vuillemin per calcolare, con  $AT^2$  ottima e tempo logaritmico, il prodotto di due matrici quadrate. Si impostino e risolvano le relazioni di ricorrenza relative all'area ed al tempo del moltiplicatore ricorsivo e si mostri come l'algoritmo di Preparata-Vuillemin combini insieme moltiplicatori sistolici e ricorsivi.