

## Algoritmi e Strutture Dati (A.A. 2013-2014)

Corso di Laurea in Ingegneria dell'Informazione  
Sapienza Università di Roma

### Prima prova di esonero (20/11/2013) – Durata 1h 30'

<p>Cognome: _____</p> <p>Nome: _____</p> <p>Matricola: _____</p>	<p>Autorizzo la pubblicazione del voto di questo esame sul sito web <a href="http://www.dis.uniroma1.it/~demetres/didattica/asd">http://www.dis.uniroma1.it/~demetres/didattica/asd</a>, secondo quanto prevede il decreto legislativo 196/2003 (codice in materia di protezione dei dati personali) che dichiaro di conoscere. In fede,</p> <p>_____</p>
--	---

**Nota bene:** risposte non motivate saranno considerate nulle.

---

#### Domanda 1 [10 punti]

- Risolvere la relazione di ricorrenza  $T(1) = 1$ ,  $T(n) = 4T(n/2) + n^2$ ,  $n > 1$ , utilizzando:
  - il metodo dell'iterazione;
  - il teorema fondamentale delle ricorrenze.
- Si enuncino poi i tre casi del teorema fondamentale delle ricorrenze e si fornisca una dimostrazione dettagliata del caso 3.

---

#### Domanda 2 [6 punti]

Dimostrare che un limite inferiore al problema di ordinare  $n$  valori mediante confronti è  $\Omega(n \log n)$ .

---

#### Domanda 3 [6 punti]

Sia  $A(n)$  il numero di asterischi in funzione di  $n$  stampati dal seguente frammento di programma C:

```
for (i=1; i<n; i*=4)
    for (j=0; j<i; j++) printf("*");
```

Trovare una funzione  $f(n)$  tale che  $A(n) = \Theta(f(n))$ .

---

#### Domanda 4 [10 punti]

Si progetti un algoritmo efficiente che, data una collezione di elementi (es. numeri), trova l'elemento che appare più frequentemente nella collezione. A parità di frequenza massima, restituire un elemento qualsiasi.

- Assumere che la collezione in input sia data sotto forma di array di  $n$  interi e fornire codice C o pseudocodice dettagliato che mostra i passi dell'algoritmo.
- Analizzare il tempo di esecuzione  $T(n)$  dell'algoritmo usando la notazione asintotica. **L'algoritmo deve avere un tempo di esecuzione meno che quadratico.**