

# INFORMATICA GENERALE

## Esame Scritto

docenti: G. BONGIOVANNI, T. CALAMONERI, I. SALVO  
Sapienza Università di Roma

10 settembre 2018

**Esercizio 1 (10 punti)** Si consideri la seguente equazione di ricorrenza:

$$T(n) = 3T(n/3) + \Theta(n)$$
$$T(1) = \Theta(1).$$

Si risolva tale equazione di ricorrenza utilizzando:

- (3 punti) il metodo iterativo;
- (2 punti) il metodo dell'albero;
- (1 punto) il metodo principale;
- (4 punti) il metodo di sostituzione.

**Esercizio 2 (10 punti)** \*\*\*IVANO FARE\*\*\*

Considerare il problema di verificare se una lista di caratteri è palindroma, cioè se letta da sinistra a destra o da destra a sinistra dà la stessa sequenza di caratteri (come le parole 'non', 'otto', 'radar', 'ingegni', 'onorarono', 'ossesso', ...).

- (3 punti) Considerate il seguente programma (dove la funzione `eqList` verifica se due liste sono uguali e `reverse` restituisce il puntatore alla testa di una lista che contiene gli elementi di `L` in ordine rovesciato):

```
int palindroma(charList L){
    if (eqList(L, reverse(L)) return 1;
    else return 0;
}
```

Sotto quali ipotesi sulla funzione `reverse` la funzione dà risultati corretti?

2. **(6 punti)** Scrivere una funzione `C ricorsiva int palindroma(charList L)` che restituisce 1 solo se la lista  $L$  in ingresso è palindroma *senza* usare strutture dati ausiliarie.
3. **(1 punto)** Quale struttura dati ausiliaria viene in realtà allocata *implicitamente* durante l'esecuzione della vostra funzione?

**Esercizio 3 (10 punti)** Sia  $G = (V, E)$  un grafo e si assuma che i nodi in  $G$  siano colorati mediante il colore rosso oppure verde. Si consideri il problema di determinare se  $G$  possiede un ciclo semplice contenente solo nodi verdi. Si descrivano le strutture dati che si preferisce usare per rappresentare il grafo ed il colore dei nodi **(2 punti)**. Poi si progetti un algoritmo efficiente che risolva il problema. Di tale algoritmo:

1. **(2 punti)** Si dia la descrizione a parole;
2. **(4 punti)** Si scriva lo pseudocodice, dettagliando tutte le funzioni, anche quelle studiate eventualmente usate;
3. **(2 punti)** Si calcoli il costo computazionale.