

# INFORMATICA GENERALE

## Esame Scritto

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

29 gennaio 2019

**Esercizio 1 (10 punti)** Si consideri l'algoritmo di Quick Sort. Di esso:

1. (**3 punti**) Si dia la descrizione a parole, inclusa la funzione di partizione;
2. (**3 punti**) Si scriva lo pseudocodice (inclusa la funzione di partizione);
3. (**1 punto**) Si mostri in dettaglio l'esecuzione dello pseudocodice sul vettore  $[6, 2, 3, 4, 5, 1]$ , specificando come cambia il valore delle variabili introdotte nello pseudocodice;
4. (**1 punto**) Si scriva l'equazione di ricorrenza che ne descrive il costo computazionale, giustificandola;
5. (**2 punti**) La si risolva con un metodo a piacere (diverso da quello principale) nei due casi migliore e peggiore, dettagliando i passaggi.

**Esercizio 2 (10 punti)** Considerare il problema di selezionare il suffisso (cioè la parte finale) lungo  $n$  di una lista. Ad esempio il suffisso lungo 3 di  $\langle 7, 2, 5, 4, 6 \rangle$  è la lista  $\langle 5, 4, 6 \rangle$ .

Scrivere le seguenti funzioni C:

1. Una funzione di prototipo:

```
list suffisso(list L, int n)
```

che restituisca gli ultimi  $n$  elementi della lista  $L$  *senza allocare nuova memoria* e deallocando il prefisso della lista.

2. Una funzione di prototipo:

`list suffissoFun(list L, int n)`

che crea una nuova lista contenente gli ultimi  $n$  elementi della lista  $L$  lasciando  $L$  inalterata.

In entrambi i casi, se la lunghezza di  $L$  è maggiore di  $n$  viene semplicemente restituita una lista contenente tutti gli elementi di  $L$ .

In entrambi i casi, sono particolarmente apprezzate (ma non necessarie) soluzioni che fanno un'unica scansione della lista di ingresso  $L$ .

**Esercizio 3 (10 punti)** Sia dato un albero binario  $T$  memorizzato tramite record e puntatori le cui chiavi sui nodi siano numeri relativi. Si progetti un algoritmo che, dato in input il puntatore alla radice di  $T$  ed un intero  $h$ , restituisca TRUE se la somma di tutte le chiavi sui nodi di  $T$  è al più  $h$ , FALSE altrimenti.

Dell'algoritmo proposto:

1. **(3 punti)** si dia la descrizione a parole;
2. **(3,5 punti)** si scriva lo pseudocodice;
3. **(2,5 punti)** si calcoli il costo computazionale, dettagliando comunque il costo delle funzioni studiate eventualmente usate;
4. **(1 punto)** si facciano le opportune osservazioni sul costo computazionale nel caso migliore e peggiore se abbiamo l'ipotesi che tutti i nodi abbiano chiavi positive.