

# Algoritmi e Strutture Dati

Docente: Sabrina De Capitani di Vinercati  
Appello online del 20 Aprile 2017  
Tempo a disposizione: 2:30 ore

## Domanda 1)

Rispondere brevemente, ma in modo completo, alle seguenti domande.

- √ 1. Quale è la complessità computazionale per la ricerca del valore minimo in un albero binario di ricerca?
- √ 2. Cosa si intende per *ordinamento topologico* di un grafo?
- √ 3. Nell'ambito delle tabelle di hash, descrivere i metodi di *scansione lineare*, *quadratica* e *hashing doppio*.
4. Dire cosa si intende per complessità computazionale asintotica e definire la notazione asintotica "o grande" e "omega grande".
- √ 5. Cosa è un *max-heap*? Si richiede di fornire un esempio.
- √ 6. Si richiede di enunciare il *Teorema di Cook-Levin* e di descrivere le sue conseguenze.
- √ 7. Definire il concetto di *componente connessa* in un grafo non orientato.
- √ 8. Descrivere la struttura dati *coda*.
- √ 9. Si richiede di descrivere il problema dello *string matching* e di illustrare un esempio.
- √ 10. Si richiede di calcolare la complessità computazionale della seguente funzione.

```
algo(n) /* n: numero intero */
{
  if (n <= 2) then return n;
  else return algo(n-1) - 2*algo(n-2);
}
```

✓ **Esercizio 1)**

Si richiede di utilizzare l'algoritmo di ordinamento QuickSort per ordinare il seguente array:  
25, 10, 21, 7, 99, 33, 41, 11, 67, 58

Si richiede di mostrare chiaramente il pivot che si utilizza ad ogni passo e come si spostano man mano gli elementi nell'array.

✓ **Esercizio 2)**

Si consideri un albero binario di ricerca non bilanciato, inizialmente vuoto. Disegnare la struttura degli alberi risultanti dopo l'inserimento di ognuna delle seguenti chiavi, nell'ordine indicato: 20, 3, 37, 30, 1, 15, 25, 32. Si richiede inoltre di disegnare l'albero dopo la rimozione del valore 37.

✓ **Esercizio 3)**

Dato un grafo non orientato  $G = (V, E)$ , non necessariamente connesso, e un vertice  $s \in V$ , si richiede di scrivere (pseudocodice) un algoritmo che restituisca il numero di nodi che fanno parte della componente connessa di cui fa parte il nodo  $s$ .