

# Algoritmi e Laboratorio

## 1° Compitino dell'11/04/17

Nome:

Cognome:

Matricola:

Corso:

### Esercizio 1 [punti 4 + 2]

Progettare un algoritmo basato sulla tecnica *Divide et Impera* il cui costo in tempo sia descritto dalla ricorrenza:

- $T(n) = 3 T(n/2) + O(n \log n)$  se  $n > 10$ ,
- $T(n) = \Theta(1)$  altrimenti.

Risolvere infine la ricorrenza.

### Esercizio 2 [punti 6]

Dato un array ordinato  $A[1..n]$  contenente  $n$  elementi interi distinti appartenenti all'intervallo  $[0, n + 1]$ , progettare un algoritmo efficiente che stabilisca se esiste un indice  $i$  tale che  $A[i]=i$ . Si valuti anche la complessità in tempo dell'algoritmo proposto.

### Esercizio 3 [punti 5]

Sia dato un vettore di interi  $A = [6, 14, 2, 9, 7, 10]$ , applicare l'algoritmo `Build_max_heap(A)` per la costruzione di un heap di massimo, disegnando la configurazione del vettore dopo ogni iterazione del ciclo **for**.

### Esercizio 4 [punti 4]

Si indichi la complessità in tempo al caso medio della ricerca con insuccesso all'interno di una tabella hash con  $n$  elementi e dimensione  $m$ , in cui le collisioni sono gestite con indirizzamento aperto, e se ne dimostri la correttezza.

### Esercizio 5 [punti 6]

Sia dato un albero binario  $T$  di dimensione  $n$ , si dice che un nodo  $u$  di  $T$  è **bilanciato** se i due sottoalberi radicati nei figli hanno la stessa altezza. Si progetti un algoritmo efficiente che conta il numero di nodi bilanciati di  $T$ . Si valuti anche la complessità in tempo dell'algoritmo proposto. [**Attenzione**: Se si fa uso di procedure viste in classe, occorre dettagliarle nella soluzione proposta]

### Esercizio 6 [punti 3]

Sia dato un palazzo di  $n$  piani, con  $n$  uguale a una potenza del 2, e un cesto che contiene un numero illimitato di uova. L'operazione consentita è quella del lancio di un uovo da un piano, con il risultato che questo si può rompere oppure no. Si assume che le uova abbiano lo stesso grado di resistenza. Si vuole stabilire qual è il piano più alto da cui si può lanciare un uovo senza che esso si rompa. Proporre una strategia che risolve il problema indicandone la complessità in termini di numero di lanci; stabilire un limite inferiore al problema; discutere l'ottimalità della soluzione proposta.