

## ESERCIZI (alberi binari di ricerca)

1. Progettare un algoritmo che verifichi se un albero binario i cui nodi contengono chiavi intere è un albero binario di ricerca.
2. Dato un albero binario di ricerca con chiavi distinte, scrivere un algoritmo che lo trasformi nel suo inverso. L'albero risultante dovrà avere la seguente proprietà: per ogni nodo  $u$  dell'albero, tutti i nodi nel sottoalbero sinistro hanno chiave maggiore della chiave di  $u$  e tutti i nodi nel sottoalbero destro hanno chiave minore della chiave di  $u$ .
3. Dato un albero binario di ricerca  $T$ , progettare e descrivere in pseudocodice un algoritmo efficiente che, per ogni nodo  $u$  dell'albero, stampi la chiave di  $u$  e la chiave di valore minimo del sottoalbero di cui  $u$  è radice. Analizzare la complessità dell'algoritmo proposto.
4. Progettare un algoritmo per trasformare un array ordinato di  $n$  interi distinti in un albero binario di ricerca che sia il più bilanciato possibile, in tempo  $O(n)$ . Valutare esplicitamente la complessità dell'algoritmo proposto.
5. Sia  $T$  un albero binario di ricerca che implementa un dizionario. Sia  $v$  un nodo di  $T$ , e sia  $T_v$  il sottoalbero con radice  $v$ .
  - Si progetti un algoritmo efficiente `countLE(v, k)` che, ricevuto in input un nodo  $v \in T$  e una chiave  $k$  restituisca il numero di elementi in  $T_v$  con chiave **minore o uguale a**  $k$ .
  - Analizzare la complessità dell'algoritmo.
6. Siano  $x$  e  $y$ ,  $x < y$ , due chiavi in un albero binario di ricerca radicato nel nodo  $u$ . Progettare un algoritmo efficiente `Distanza(u,x,y)` per trovare la distanza (ovvero il numero minimo di archi) tra il nodo di chiave  $x$  e il nodo di chiave  $y$ .
7. Dato un intervallo  $[a, b]$  e un albero binario di ricerca, progettare un algoritmo efficiente per stampare in ordine crescente le chiavi  $k \in [a, b]$ .
8. Dato un albero binario di ricerca, stabilire se il nodo  $u$  è il nodo mediano del sottoalbero di cui è radice.