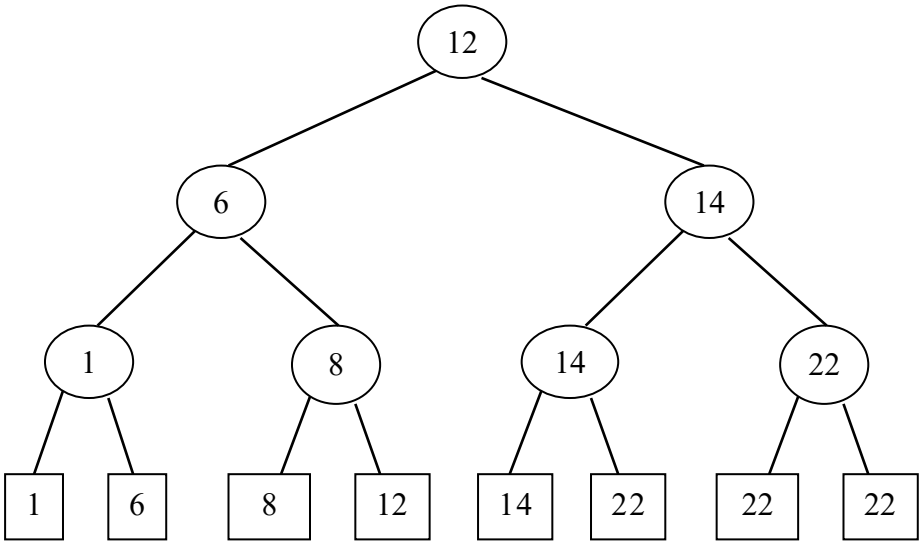
In this problem, we would like to provide a new structure for a binary search tree. The principles are as following:

- Values are stored in the tree leaves (leaf nodes)
- The left subtree of a node contains only nodes with keys less than the node's key.
- The right subtree of a node contains only nodes with keys greater than the node's key.
- Subtrees must be a binary tree having 2 child nodes (terminal or internal node)
- Each internal node stores the largest value contained in its left subtree.

The below figure shows an example for data = {1,6,8,12,14,22}, h = round_up($\log_2 6$) = 3, the binary tree will have n = 2^3 = 8 leaf nodes and the empty leaf nodes are filled with the maximum in the array.



a) 2pts - Propose an Abstract Data Type implementation for this binary tree
b) 7pts - Implement this data structure in C/C++ with the necessary functions.
c) 1pts - Test and display the results using any order