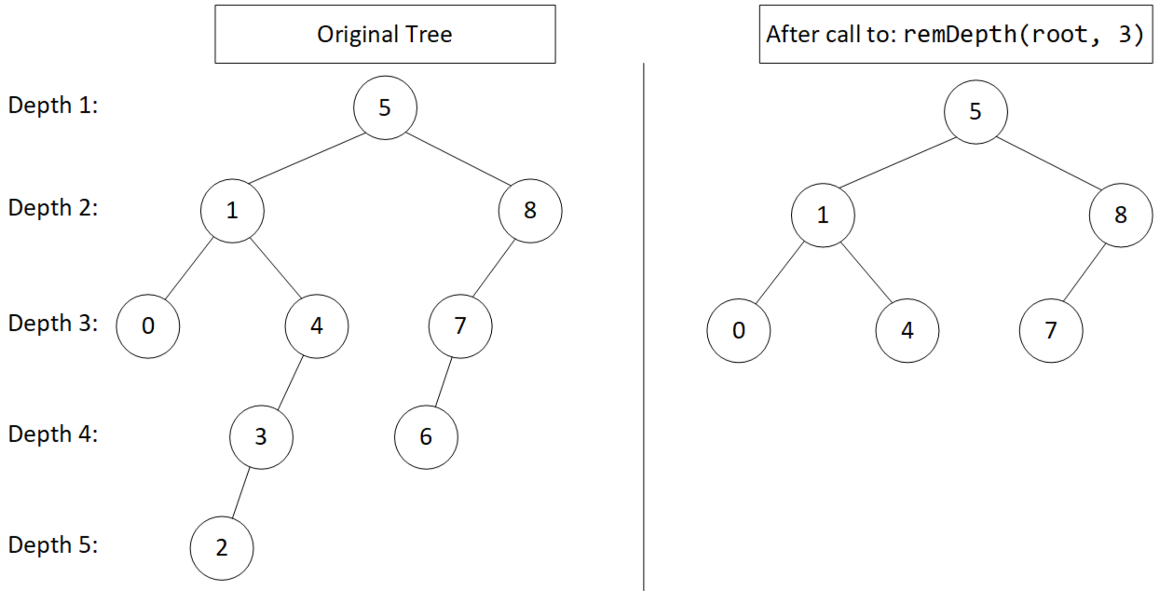
5. Binary Trees and Recursion (8 pts.)

Use the provided skeleton file `remdepth.cpp` to complete the code.

Write a recursive function: **`Node* remDepth(Node* root, int depth)`** to delete all nodes **`BELOW`** a certain depth from a **binary** tree (not necessarily BST), and return the pointer to the (potentially now NULL) root. Your implementation **MAY NOT** use loops anywhere. You may define helper function(s), as necessary.

An example, showing how we define depth and how the function should work is illustrated below.



Other examples: A call to **`remdepth(root, 0)`** would cause the entire tree to be deleted, in which case you should return **`nullptr`**. A call to **`remdepth(root, 1)`** would cause all nodes **except the root** to be deleted.

If no nodes are below the specified depth, simply do nothing (i.e. do not alter the tree in any way).

Your code should run in **Theta(n)**, where **n** is the number of nodes in the tree.