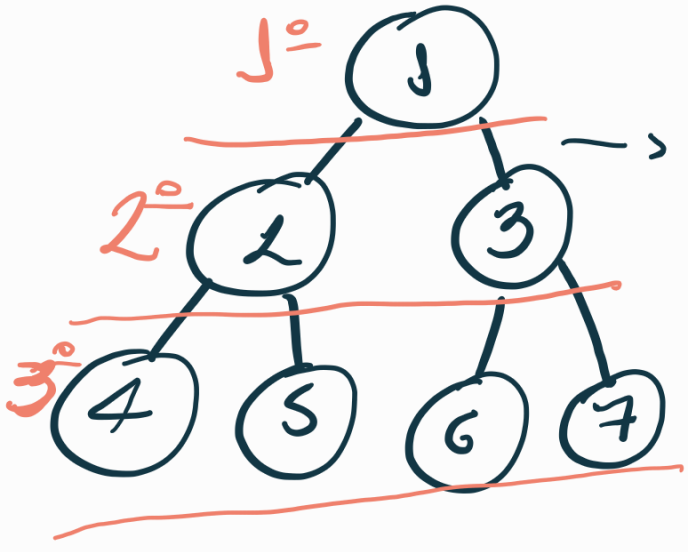
→ given an BST, for each level of the tree, return a list with all nodes contained at this layer.

L> each layer/level should have a list with all node values sorted from left to right.



→ [1], [2,3], [4,5,6,7]

→ first, i need an global var to store values for each layer

L> Map<int, list<int>

→ for each dive onto levels i need to check if current level exists in <K, V>, if it does, then add value to this Key, and if not, create the key and input the value.

→ I also need to assure that it will be first traversing the whole left side starting from root, to only then account for right side.

→ to know at what level im into start a counter at 1 at root, and increment it at each recursion.

→ After having filled storage need to format final answer

- iterate over keys in order
  - get list of keys and sort it into a list
    - get this list with sorted (asc) values, get each value (list) and place it in wrapper list as final answer.