1. Compute the height of the tree

Use the array in the example.

1. Extract the items in the B-tree into a sorted list.

Create a empty list. For the function, when get the leaf, insert to the list, else keep going to each leaves and append the node.

1. Return the minimum element in the tree at a given depth d.

Check the level is correct or not, if correct, check is the depth code want or not. If not, go to next level. If yes, return the first number.





1. Return the maximum element in the tree at a given depth d.

Check the level is correct or not, if correct, check is the depth code want or not. If not, go to next level. If yes, return the last number.





1. Return the number of nodes in the tree at a given depth d.

Check is the depth is correct/what we want

If yes, return the number of node

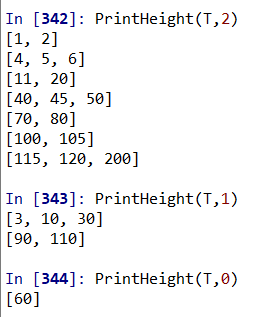
If not, go to child with +1 to number of node

1. Print all the items in the tree at a given depth d.

Check is the depth is correct/what we want

If yes, print number

If not, go to child



1. Return the number of nodes in the tree that are full.

Check is leaf or not

If yes, return the number of node

If not, go to child and +1 to node



1. Return the number of leaves in the tree that are full.

Check is leaf or not

If yes, return count the leaves

If not, go to child

9. Given a key k, return the depth at which it is found in the tree, of -1 if k is not in the tree.