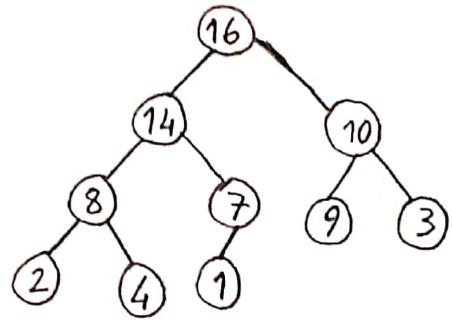


Consider the binary max-heap given (from lecture slides).

Preorder : 16, 14, 8, 2, 4, 7, 1, 10, 9, 3

Inorder : 2, 8, 4, 14, 1, 7, 16, 9, 10, 3

Postorder : 2, 4, 8, 1, 7, 14, 9, 3, 10, 16

None of the traversal types resulted in a sorted output since heaps are not ordered similar to the binary search tree. Although parent nodes have always greater keys than its children in max-heap (reverse is true for min-heaps), there is no exact relation between left and right child. For example, node 8's left child 2 is smaller than right child 4 but node 10's left child 9 is greater than right child 3. Hence, there is no guarantee for sorted output in each traversal since child nodes are ordered in no particular order.

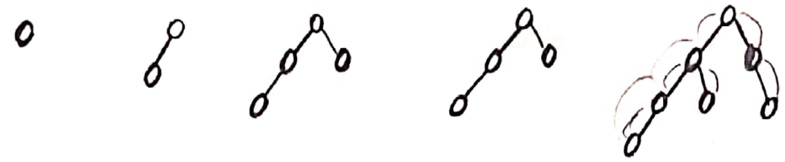
height = 0 1 2 3 4 ...

min. nodes = 0 1 2 4 7 ...

Min. no of nodes (height h)

$N(h) = 1 + N(h-1) + N(h-2)$

$N(0) = 0$, $N(1) = 1$ } base cases



$N(0) = 0$ $N(3) = 4$ $N(6) = 20$ $N(9) = 88$ $N(12) = 376$ $N(15) = 1596$

$N(1) = 1$ $N(4) = 7$ $N(7) = 33$ $N(10) = 143$ $N(13) = 609$

$N(2) = 2$ $N(5) = 12$ $N(8) = 54$ $N(11) = 232$ $N(14) = 986$

Minimum number of nodes in an AVL tree of height 15 = 1596.