More on Complete Binary Trees

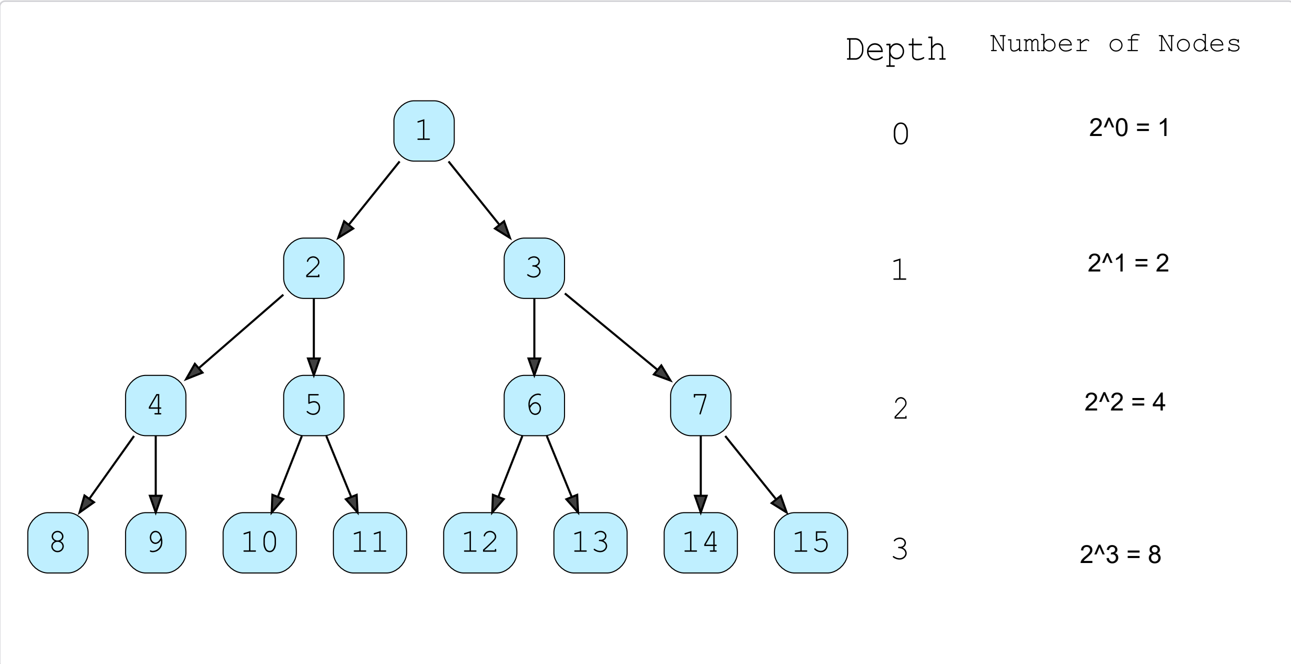
In this lesson, we are going to discuss what Complete Binary Trees are and how elements are inserted into them.

We'll cover the following

Introduction

We touched upon complete binary trees in the last lesson, but here are some more detailed properties of them.

- All the levels are completely filled except possibly the last one
- Nodes at the last level are as far left as possible
- The total number of nodes, n, in a complete binary tree of height **"h"** are: $2^h \le nodes \le 2^{h+1} - 1$. This is again based on the Geometric Series (https://en.wikipedia.org/wiki/1_%2B_2_%2B_4_%2B_8_%2B_%E2%8B%AF) formula: $2^0 + 2^1 + 2^3 + 2^4 + ... + 2^r = 2^{r+1} - 1$



- The total number of non-leaf nodes, n_i in a complete binary tree of height **"h"** are expressed as a range like so:

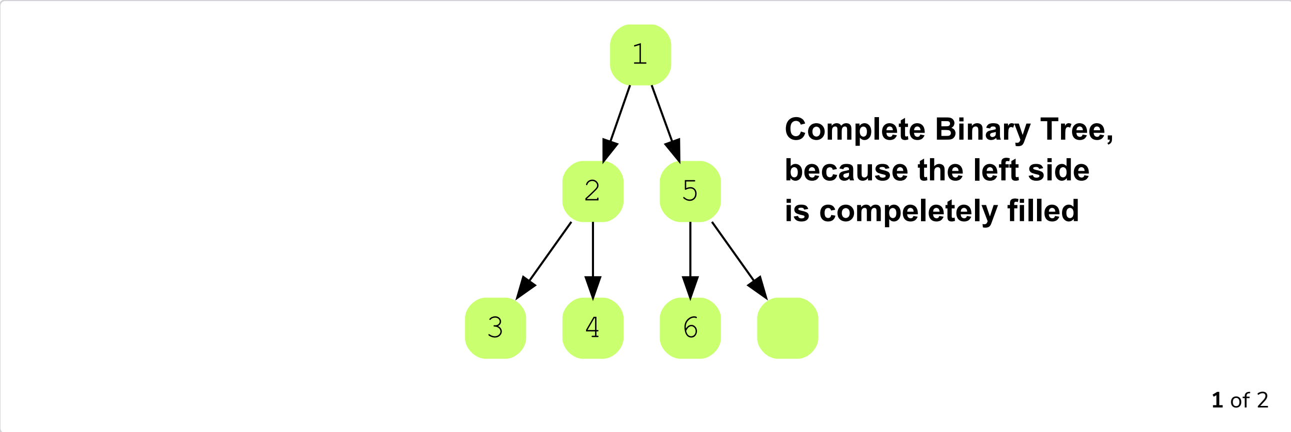
$$floor(2^{h-1}) \le n_i \le 2^h - 1$$

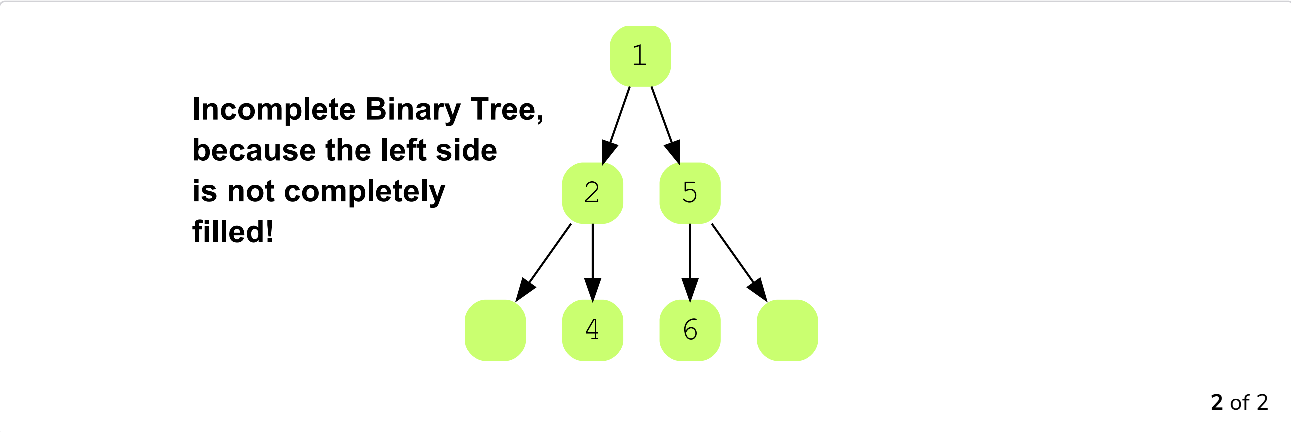
- The total number of leaf-nodes, n_e in a complete binary tree of height **“h”** is expressed as a range like so:

$$2^{h-1} \leq n_e \leq 2^h$$

- The nodes, n, are present in between the range of:

$$2^h \leq n \leq 2^{h+1} - 1$$

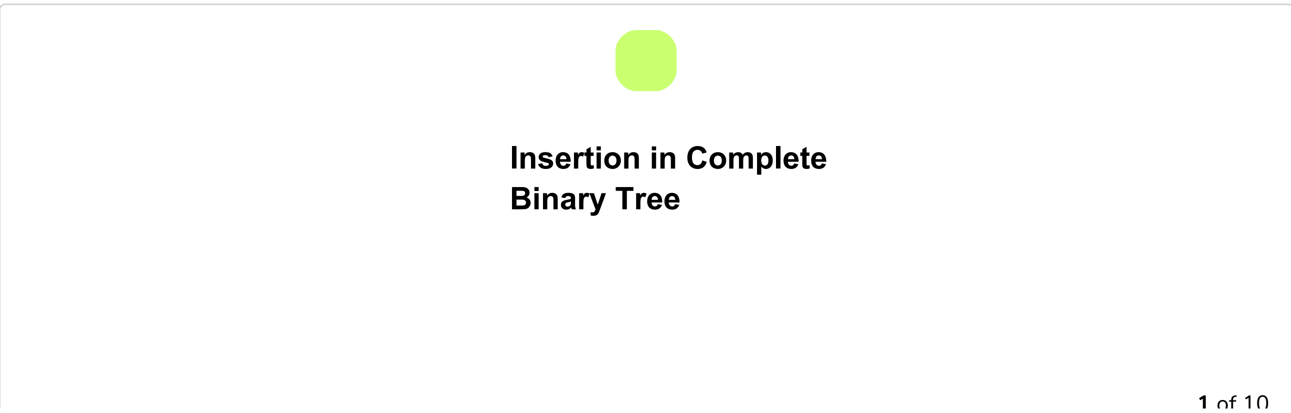




Insertion in Complete Binary Trees

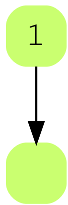
The following rules apply when inserting a value in a Complete Binary Tree:

- Nodes are inserted level by level
- Fill in the left-subtree before moving to the right one



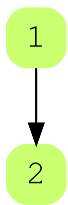
Insert 1!

2 of 10



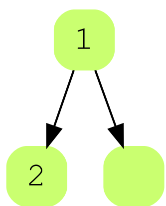
Insert 2!

3 of 10



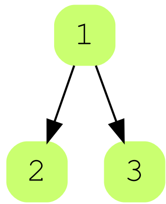
Insert 2!

4 of 10



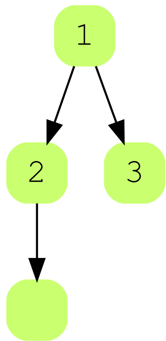
Insert 3!

5 of 10



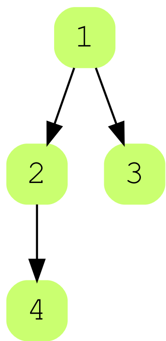
Insert 3!

6 of 10



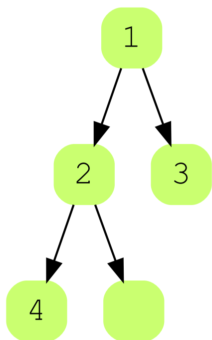
Insert 4!

7 of 10



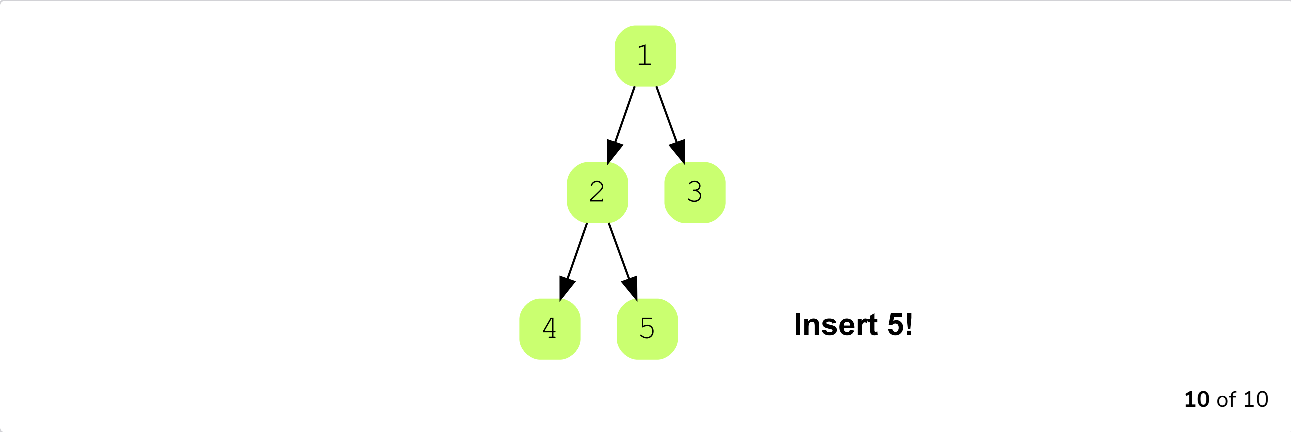
Insert 4!

8 of 10



Insert 5!

9 of 10



Explanation

As you can see in the animation above, Node 4 was inserted as a left child of Node 2 to meet the property of complete binary trees. In a Complete Binary Tree there exist no node that has a right child but not a left child. So during Insertion, make sure to insert a node as a left child first if it's empty to fill in the left sub-tree before moving to right sub-tree.

In the next lesson, we will study *Skewed Trees* which is another variation of Binary Trees!

← Back

What is a Binary Tree?

Next →

Skewed Binary Trees

Mark as Completed

Report an Issue

Ask a Question (https://discuss.educative.io/tag/more-on-complete-binary-trees__introduction-to-trees__data-structures-for-coding-interviews-in-python)