

Chapter 6

Binary Search Trees.

6.1 Binaries Trees and How to Encode Them.

We have already seen, in heaps, that organizing our data in a graph-like structure can offer a speed advantage. For future applications, and in particular for maintaining data in sorted order, we will have to encode our data using binary trees. These trees may not be almost complete and also have to support pointer manipulations, specifically placing a binary tree as a left or right subtree of a given node. To enable this, we will have to treat the **right**, **left**, and **parent** as variables, in contrast to heaps where they are determined completely by the node index. We begin this section by stating definitions.

- Definition 6.1.1.**
1. *Binary Tree: A tree in which any vertex has at most two children.*
 2. *A descendant of vertex x is a vertex in the subtree whose root is x .*
 3. *An ancestor of x is a vertex to which x belongs as a descendant.*
 4. *A leaf is a vertex without children.*
 5. *Height of vertex x is the length of the longest simple path (without cycles) between x and one of the leaves.*
 6. *Height of the tree is the height of its root, which is usually denoted by h .*