# Lecture Notes, Week 5

## Textbook Definition of a Tree

A **tree T** is a set of nodes, using storing values, in a parent-child relationship with the following properties:

* T could be empty, i.e. it has no nodes. This is called the **empty tree**.
* If T is non-empty, it has a special node called the **root of T** that has no parent.
* Each node v of T *different* from the root has a *unique parent* node w. We say v is a **child** of w.

Notes:

* Every non-empty tree has exactly one root. When drawing a tree, we typically put the root at the top.
* All nodes in a tree can have 0 or more child nodes, called **children**.
* A node with 0

Formally, we define tree T to be a set of nodes storing elements in a parent-child relationship with the following properties: • If T is nonempty, it has a special node, called the root of T, that has no parent. • Each node v of T different from the root has a unique parent node w; every node with parent w is a child of w. Note that according to our definition, a tree can be empty, meaning that it doesn’t have any nodes. This convention also allows us to define a tree recursively, such that a tree T is either empty or consists of a node r, called the root of T, and a (possibly empty) set of trees whose roots are the children of r.