**Chapter 11**

Table of Contents

[11.1 Introduction to Trees 2](#_Toc44627769)

[11.3 Tree Traversal 3](#_Toc44627770)

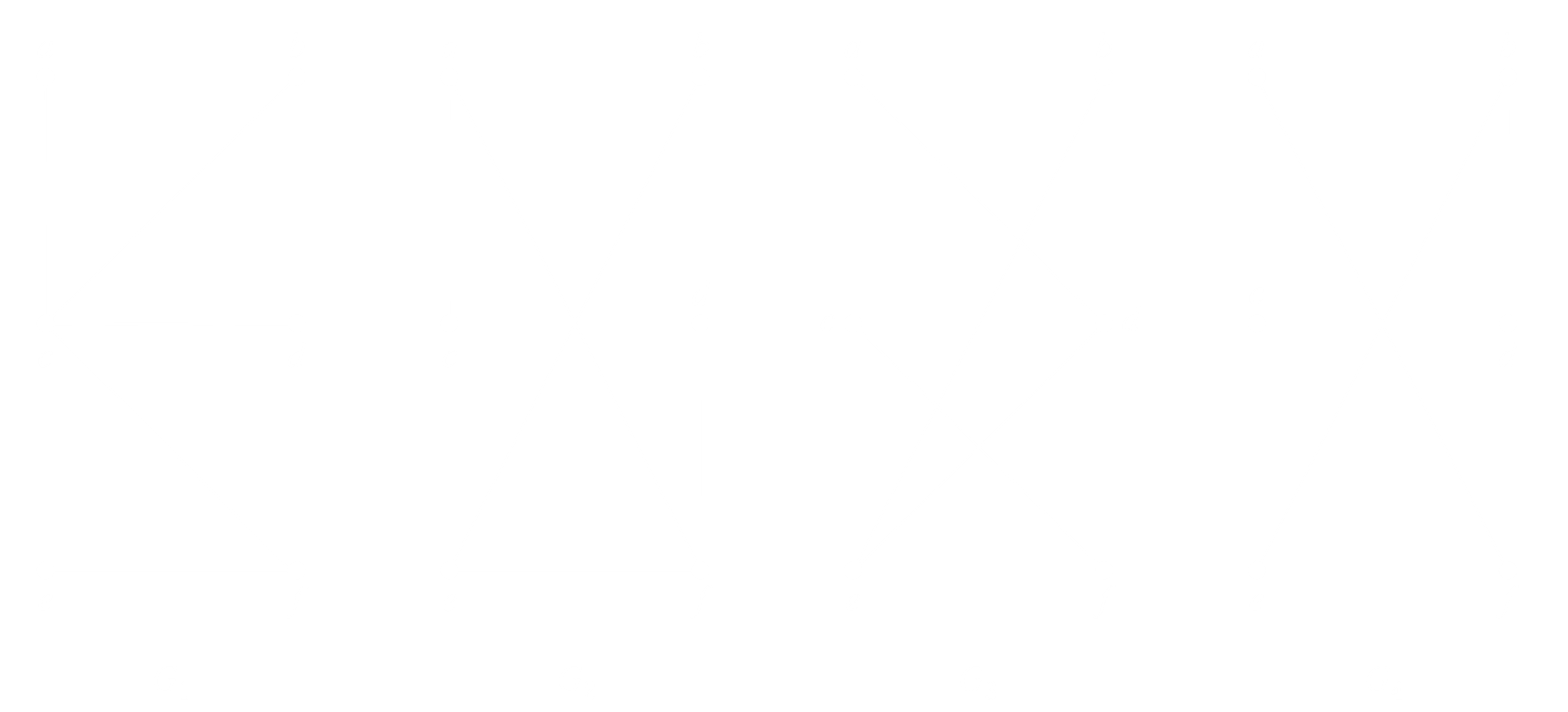
[Preorder Traversal 4](#_Toc44627771)

[Inorder Traversal 4](#_Toc44627772)

[Postorder Traversal 5](#_Toc44627773)

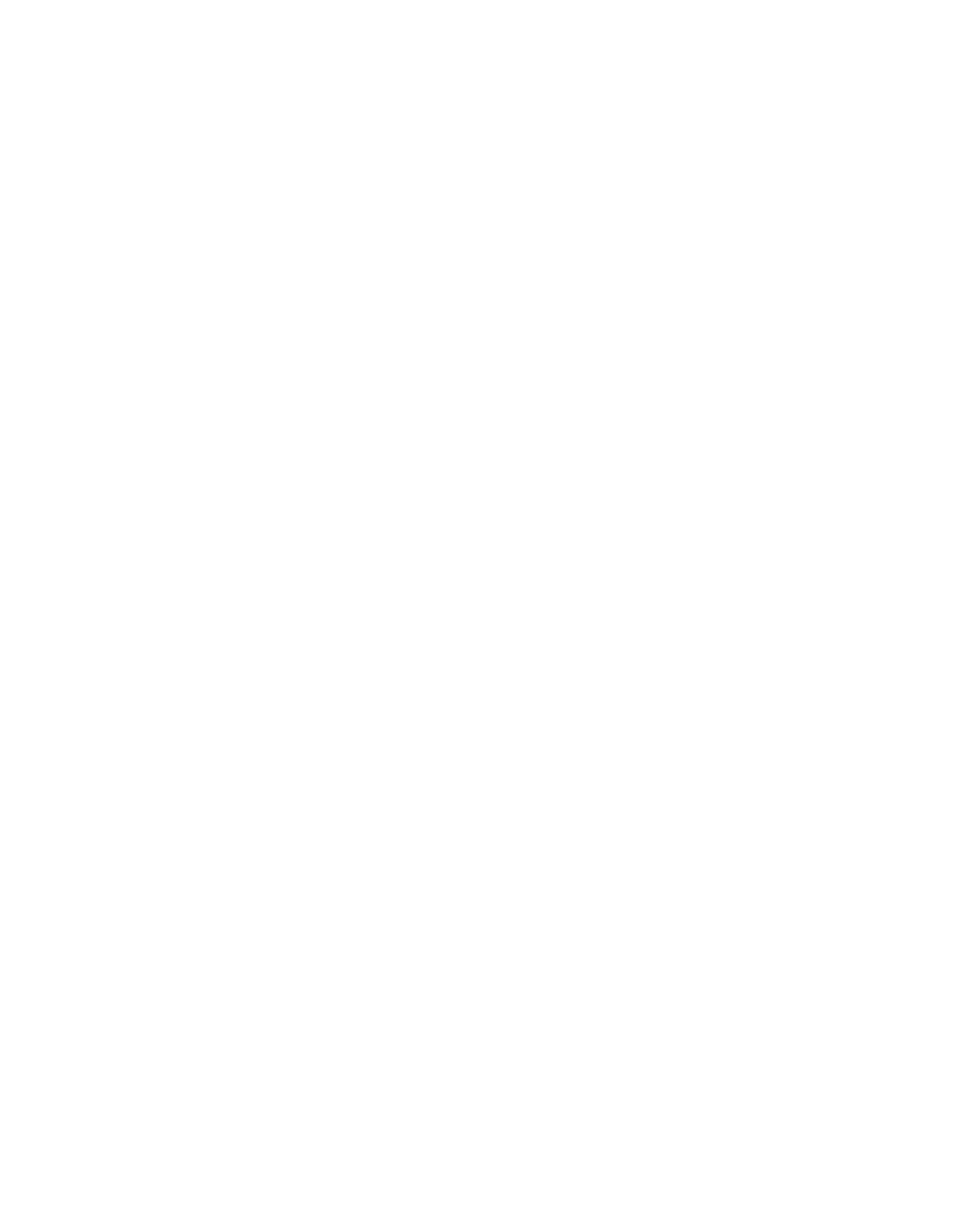
## 11.1 Introduction to Trees

A tree is a connected undirected graph with no simple circuits.



Here, and are trees because both are connected graphs with not simple circuits. is not a tree because is a simple circuit. is not a tree since it is not connected.

A rooted tree is a tree in which one vertex has been designated as the root and every edge is directed away from the root.



Here, the root is .

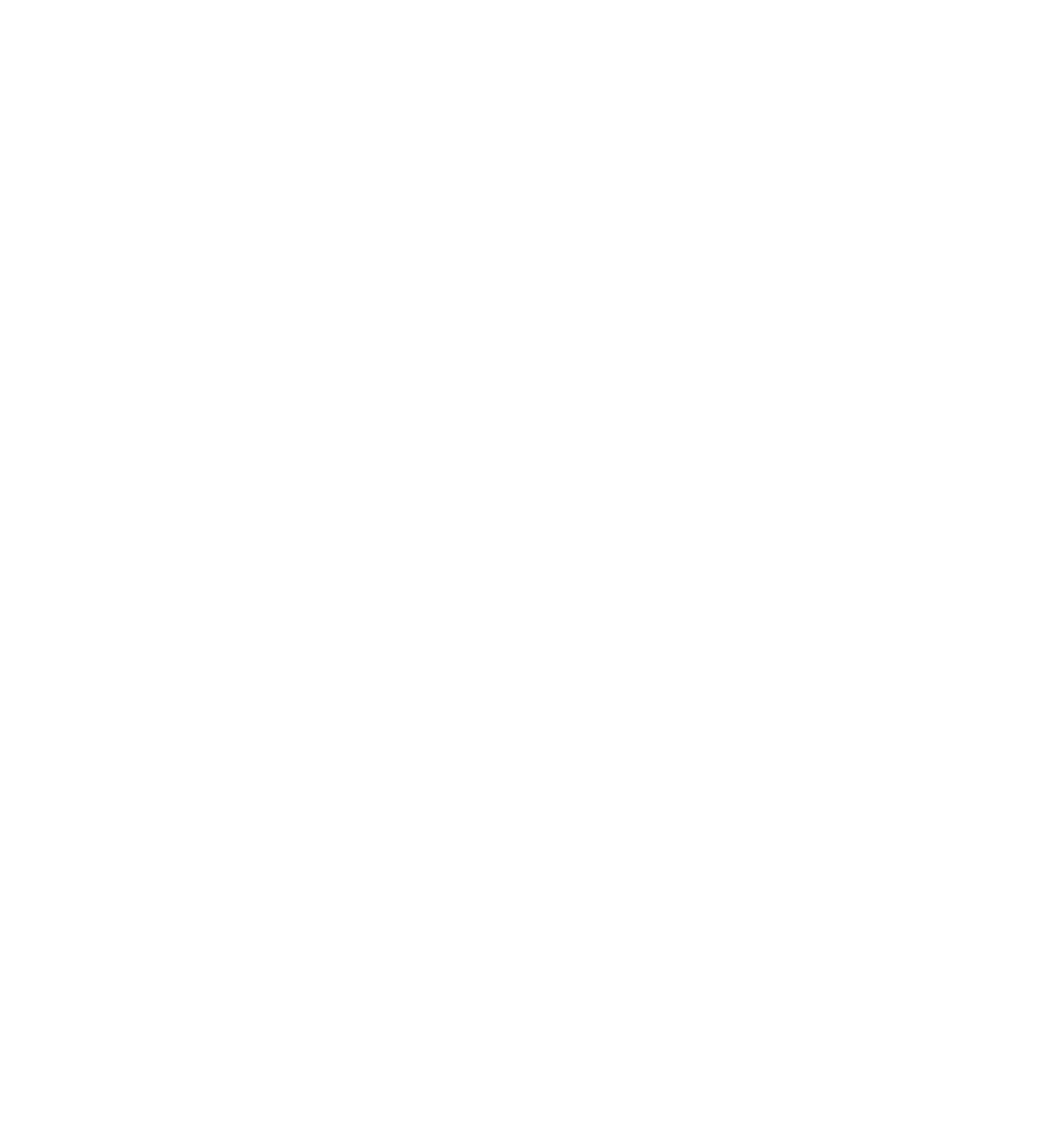
Usually, a rooted tree is shown with the root on top. This allows for some genealogical terminology.

If is a vertex other than the root, the Parent of is the unique vertex such that is above . Here, is the parent of , is the parent of and so on. is in turn known as the Child of . Vertices with the same parent are known as siblings, such as and , which both have the parent .

The ancestors of a vertex , other than the root, are the vertices in the path from the root to that vertex. Here, the ancestors of are and . In turn, the descendants of a vertex are the vertices that have as an ancestor.

## 11.3 Tree Traversal

An ordered rooted tree is a tree in which the order of the vertices is specified.



A traversal algorithm is a procedure for systematically visiting every vertex in an ordered rooter tree.

### Preorder Traversal

For an ordered root tree , with root , if consists only of , then is the preorder traversal of . Otherwise, if , , … are the subtrees of from left to right, then the preorder traversal begins at , and continues traversing in preorder, then in preorder and so on until has been traversed in preorder.

In the above diagram, the traversal would go:

Essentially, we visit the root, and then each subtree from left to right. If the subtrees have subtrees of their own, we visit those subtrees left to right first.

### Inorder Traversal

For an ordered tree with root , if consists only of , then is the inorder traversal of . Otherwise, if , , … are the subtrees of from left to right, the inorder traversal begins by traversing in order, then visiting , and then traversing in order, then in order and so on until is traversed in order.

In the above diagram, the traversal would go:

Essentially, we visit the left-bottom-most vertex, and then travel up until we find a vertex than has other children. We then visit the left-bottom-most descendant of those children and travel up again in the same fashion.

### Postorder Traversal

For an ordered rooted tree , with root , if consists only of , then is the postorder traversal of . Otherwise, if , , … are the subtrees of from left to right, the postorder traversal begins by traversing in postorder, then in postorder and so on until is traversed in postorder, and finally ends by visiting .

In the above diagram, the traversal goes:

Essentially, we visit the left-bottom-most child, and then travel up until we find a vertex that has other children. Before visiting this vertex, we visit the other children from left to right in the same manner first, and then visit that vertex.