We will primarily talk about **binary trees**:  
- How many parents does each vertex have?  
- Which vertex has the fewest **children**?  
- Which vertex has the most **ancestors**?  
- Which vertex has the most **descendants**?  
- List all the vertices in b’s left **subtree**.  
- List all the **leaves** in the tree.

### Definition: Binary Tree

A *binary tree* $T$ is:

The height of a tree $T$ is:

### Tree Property: Full

### Tree Property: Perfect

### Tree Property: Complete

### Towards a Tree Implementation – Tree ADT:

<table>
<thead>
<tr>
<th>ADT Functionality (English Description)</th>
<th>Function Call</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Tree Class

```cpp
#pragma once

template <typename T>
class BinaryTree {
    public:
        /* ... */
    private:
};
```
Trees are nothing new – they’re fancy linked lists:

Theorem: If there are $n$ data items in our representation of a binary tree, then there are ___________ NULL pointers.

Traversals:

**CS 225 – Things To Be Doing:**

1. Programming Exam A is on-going (ends on Sunday!)
2. MP3 extra credit deadline is Monday!
3. lab_quacks due Sunday
4. Daily POTDs