We have covered what is a tree. We know that a tree is a connected acyclic graph.

1 Trees

How could we represent trees? It is fine for us to represent trees the same as graphs: using either matrix and adjacency list. But tree is a special type of graph, so a new representation will be needed.

Recall a tree is a connected acyclic graph. If a tree has $n$ vertices, then it has $n - 1$ edges. It is inefficient to use the matrix representation to represent a graph. It is also very important to find the siblings of one vertex in a tree.

We have used binary trees before. Each node in a binary tree will have either 0, 1, or two children. But in general, we know that a tree may have arbitrary number of children, so the representation for binary graph is not enough.

```java
class TreeNode {
    Object data; // we could have anything here
    TreeNode child; // link to the first child
    TreeNode nextSibling; // link to the next sibling
}
```

Based on the above representation, how could we represent the following graph?

```
        a
       / \  
      b   c
     / \  /  
    d   e    g
   /   /   /   /
  f
```

Figure 1: Sample tree

We have covered depth-first traversal and breadth-first traversal of graph, could we construct trees based on either depth first traversal or breadth first traversal?