**Date :** 9th - 10 - 2020
Morning Session : 9am – 11.00 PM
By ~ Rohan Kumar

Topics: Binary Tree

Trees: Unlike Arrays, Linked Lists, Stack and queues, which are linear data structures, trees are hierarchical data structures.
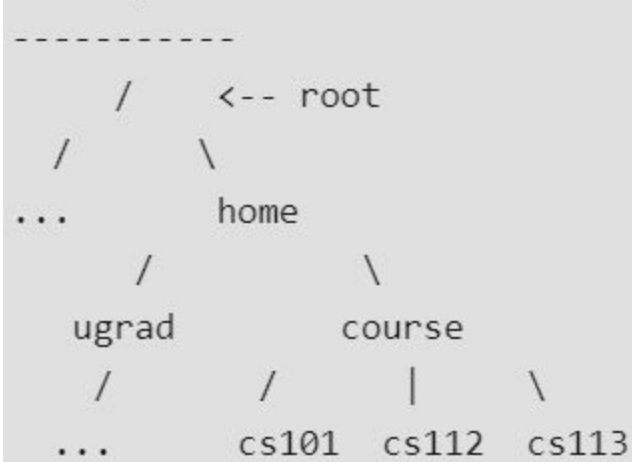
Tree Vocabulary: The topmost node is called the root of the tree. The elements that are directly under an element are called its children. The element directly above something is called its parent. For example, 'a' is a child of 'f', and 'f' is the parent of 'a'. Finally, elements with no children are called leaves.

```
      tree
      ----
       j      <-- root
      / \
    f      k
   / \      \
  a   h      z      <-- leaves
```
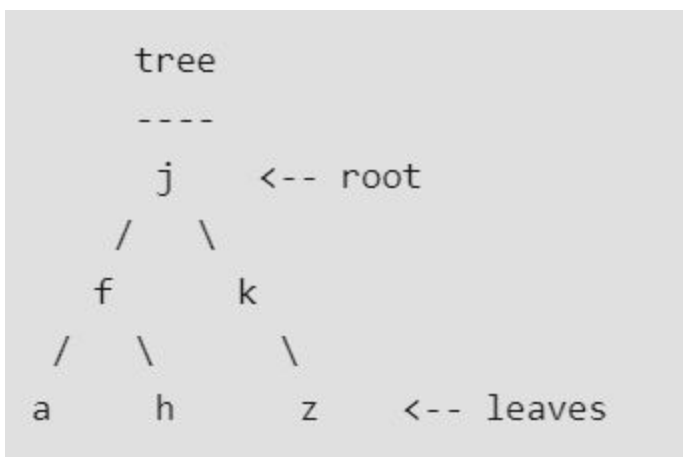
Why Trees?

1. One reason to use trees might be because you want to store information that naturally forms a hierarchy. For example, the file system on a computer:

```
file system
-----------
     /     <-- root
    /    \
  ...       home
    /          \
  ugrad        course
   /     /    |    \
  ...  cs101 cs112 cs113
```

2. Trees (with some ordering e.g., BST) provide moderate access/search (quicker than Linked List and slower than arrays).

3. Trees provide moderate insertion/deletion (quicker than Arrays and slower than Unordered Linked Lists).

4. Like Linked Lists and unlike Arrays, Trees don't have an upper limit on number of nodes as nodes are linked using pointers.

Main applications of trees include:

1. Manipulate hierarchical data.

2. Make information easy to search (see tree traversal).

3. Manipulate sorted lists of data.

4. As a workflow for compositing digital images for visual effects.

5. Router algorithms

6. Form of a multi-stage decision-making (see business chess).

Binary Tree: A tree whose elements have at most 2 children is called a binary tree. Since each element in a binary tree can have only 2 children, we typically name them the left and right child.

Binary Tree Representation in C: A tree is represented by a pointer to the topmost node in a tree. If the tree is empty, then the value of the root is NULL.

A Tree node contains the following parts.

1. Data

2. Pointer to left child

3. Pointer to right child

Summary: Tree is a hierarchical data structure. Main uses of trees include maintaining hierarchical data, providing moderate access and insert/delete operations. Binary trees are special cases of trees where every node has at most two children.

To see linked list implementation in please watch Recorded Lecture

Recorded Lecture

Instructor Provided Resource Link:

https://www.geeksforgeeks.org/binary-tree-set-1-introduction/

MCQ 1:

1. The number of edges from the root to the node is called _____ of the tree.
a) Height
b) Depth
c) Length
d) Width

Answer: B

MCQ 2:

2. The number of edges from the node to the deepest leaf is called _____ of the tree.
a) Height
b) Depth
c) Length
d) Width

Answer: A

MCQ 3:

3. What is a full binary tree?
a) Each node has exactly zero or two children
b) Each node has exactly two children
c) All the leaves are at the same level
d) Each node has exactly one or two children

Answer: A

MCQ 4:

4. What is a complete binary tree?
a) Each node has exactly zero or two children
b) A binary tree, which is completely filled, with the possible exception of the bottom level, which is filled from right to left
c) A binary tree, which is completely filled, with the possible exception of the bottom level, which is filled from left to right
d) A tree In which all nodes have degree 2

Answer: C.