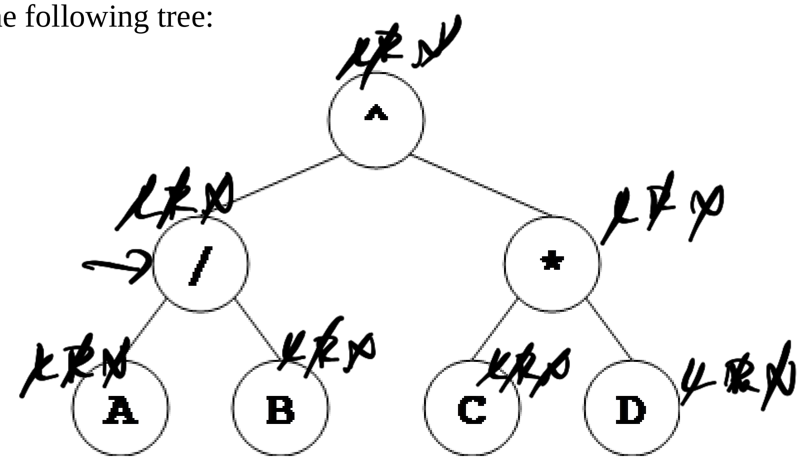
Recitation11

1. Consider the following tree:



LNR
NLR
LRN

a. What is the in-order traversal of this tree? A/B^C*D
b. What is the pre-order traversal of this tree? ^/AB*CD
c. What is the post-order traversal of this tree? AB/CD*^

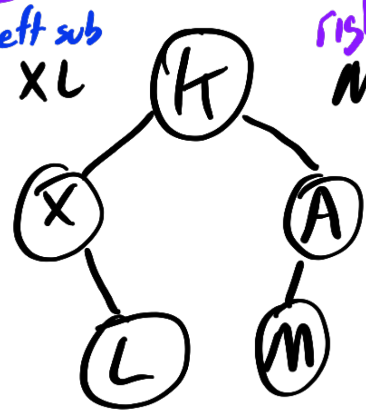
2. Given the following traversals, construct the general binary tree.

Infix
XL K MA
Left Root Right

NLR Pre-order: **KXLAM**
LNR In-order: **XLKMA**
LRN Post-order: **LXMAK**

left sub
XL

rightsub
MA



left sub

Pre	XL
In	XL
Post	LX

right sub

Pre	AM
In	MA
Post	MA

Steps
1) Find Root w/ Preorder or Postorder
2) Look at Inorder to divide into Left and Right subtrees,

3a) $\boxed{2^{h+1}-1}$

3. Binary Trees

3b) Max = Full binary
Min = Linked List

a. The height of a binary tree is the length of the longest root-to-leaf path in it. What is the maximum number of nodes in a binary tree of height h?

Means you have a full Binary tree.

b. The height of a binary tree is the length of the longest root-to-leaf path in it. The maximum and minimum number of nodes in a binary tree of height 5 are?

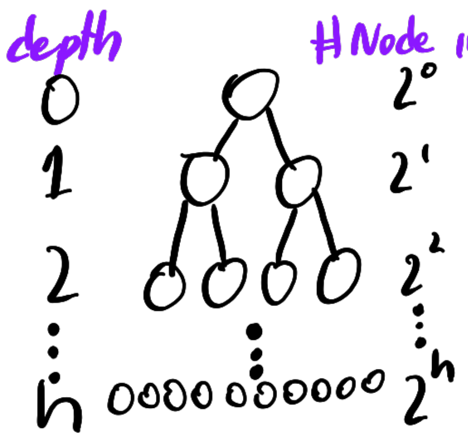
Max: $2^{5+1}-1 = \boxed{63}$ Min: $\boxed{6}$

Nodes in Full Binary

$$\boxed{\sum_{i=0}^{h} 2^i = 2^{h+1}-1}$$

c. A full binary tree with n leaves contains how many nodes?

$\boxed{2n-1}$



3c) Find # Nodes in Full Binary

In a full binary tree of h height.

The last level = the number of leaves

$2^h = n$ multiply by 2 = $2^{h+1} = 2n$

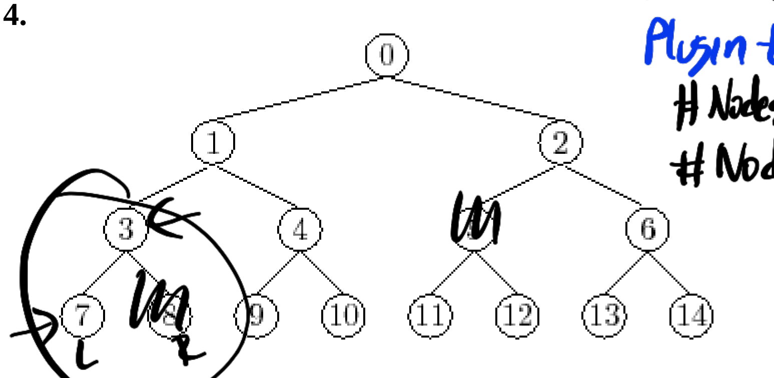
Plugin to formula (3a)

Nodes in Full Binary = $2^{h+1}-1$

Nodes in Full Binary = $2n-1$

$\boxed{2n-1}$

4.



0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Given the above tree and its implementation using an array. Write down the methods to return the left child, the right child and the parent of a particular node with the given index.

```
int left(int i)
    return 2i+1;

int right(int i)
    return 2i+2;

int parent(int i)
    return (i-1)/2;
```