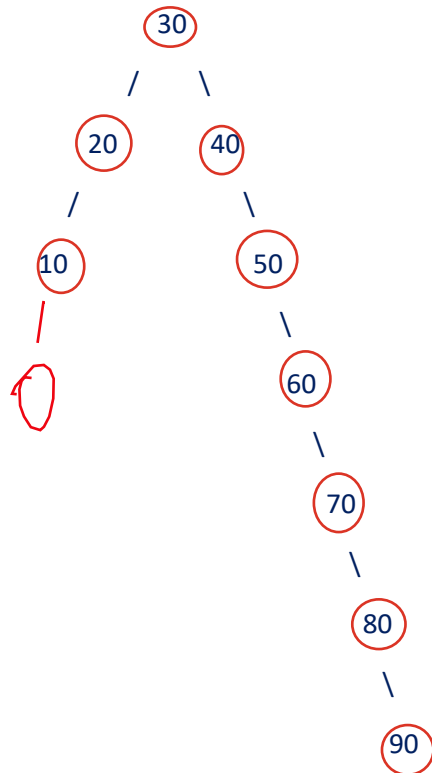## Tutorial 6

Binary Tree

1. **Draw a Binary Tree stored integer number having root node = 30, there are 3 nodes on the left of the Tree and the height of this tree is 6**
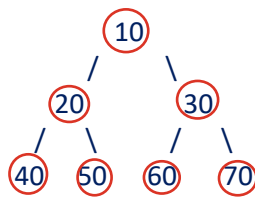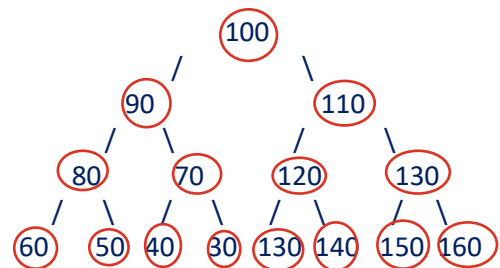
```
          30
         /  \
       20    40
       /       \
     10         50
                  \
                   60
                     \
                      70
                        \
                         80
                           \
                            90
```

left nodes = 3

– 0.5

2. Give 2 examples of Full Binary Tree

1.
```
            10
           /  \
         20    30
        / \    / \
      40  50 60  70
```

2.
```
                100
               /    \
             90      110
            / \      / \
          80   70  120  130
         / \   / \  / \  / \
       60  50 40 30 130 140 150 160
```
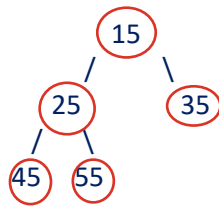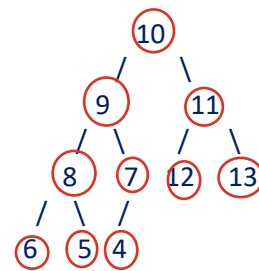
3. Give 2 examples of complete binary tree which is not a full binary tree

1.

```
        15
       /    \
      25     35
     /  \
    45  55
```

2.

```
            10
           /    \
          9      11
         / \    /  \
        8   7  12  13
       / \  /
      6  5 4
```
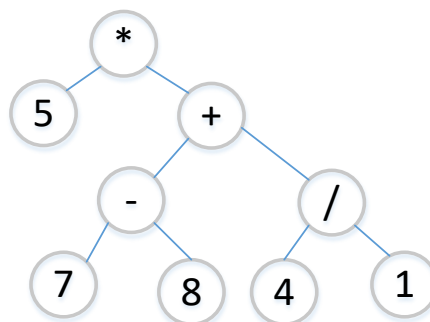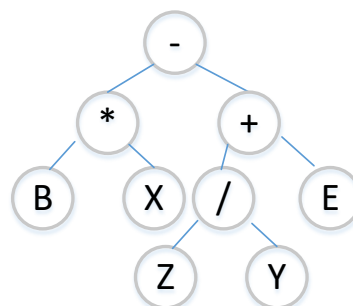
4. Binary Tree can be used to represented mathematical expression. We usually use "Infix" expression in everyday life i.e. a+b to say "a plus b", but there are 2 more possible formats which are "prefix and postfix" i.e. "+ab and ab+" respectively. Use the inorder, preorder, and postorder traversals to determine the infix, prefix, and postfix expression from the following tree
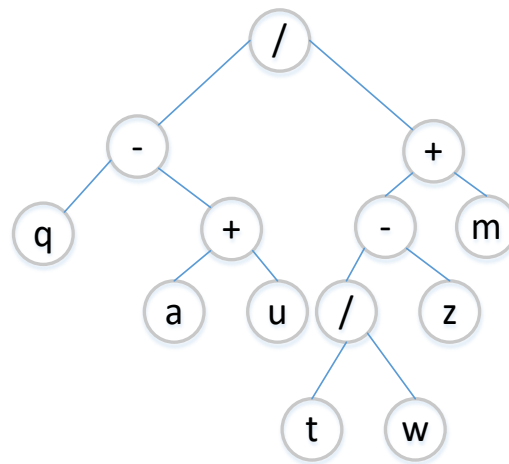
4.1

```
            *
          /   \
         5     +
              / \
             -   /
            / \ / \
           7  8 4  1
```

Infix: 5*((7-8)+(4/1))

4.2

```
            -
          /   \
         *     +
        / \   / \
       B   X /   E
            / \
           Z   Y
```
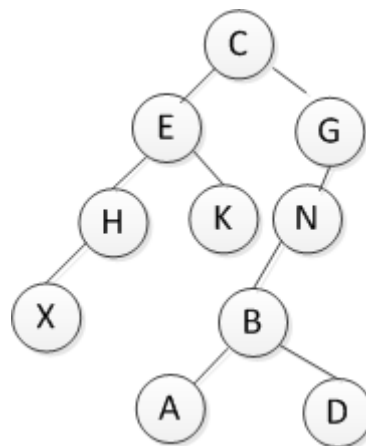
Prefix: -*BX+/ZYE

2

4.3

Postfix: qau+-tw/z-m+/

5. From the following tree:

In order Traversal : XHEKCABDNG

Pre-order Traversal: CEHXKGNBAD

Post order Traversal: XHKEADBNGC

6. Infix to prefix expression conversion using stack

(2+45^6/(7+8))

6.1 Reverse the infix expression

))8+7(/6^45+2(

6.2 Make every '(' as ')' and every ')' as '('

((8+7)/6^45+2)

6.3 Convert expression to prefix form

Expression	Stack	Output	Comment
((8+7)/6^45+2)	Empty		Initial
(8+7)/6^45+2)	(		Push
8+7)/6^45+2)	((		Push
+7)/6^45+2)	((	8	Print
7)/6^45+2)	((+	8	Push
)/6^45+2)	((+	8 7	Print
/6^45+2)	(	8 7 +	Pop until (
6^45+2)	(/	8 7 +	Push
^45+2)	(/	8 7 + 6	Print
45+2)	(/^	8 7 + 6	Push
+2)	(/^	8 7 + 6 45	Print
2)	(+	8 7 + 6 45 ^ /	Pop Pop and Push
)	(+	8 7 + 6 45 ^ / 2	Print
		8 7 + 6 45 ^ / 2 +	Pop until (

6.4 Reverse the expression and show the result

+ 2 / ^ 45 6 + 7 8