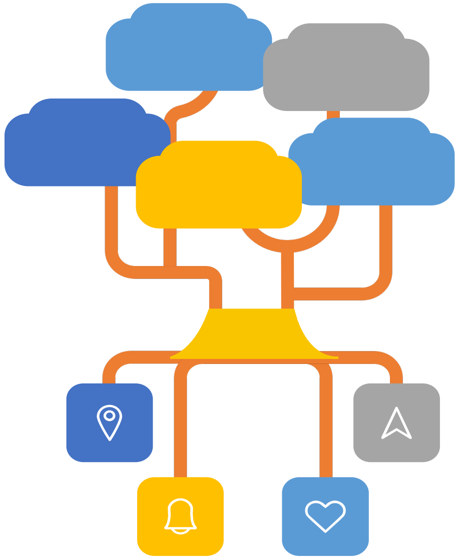
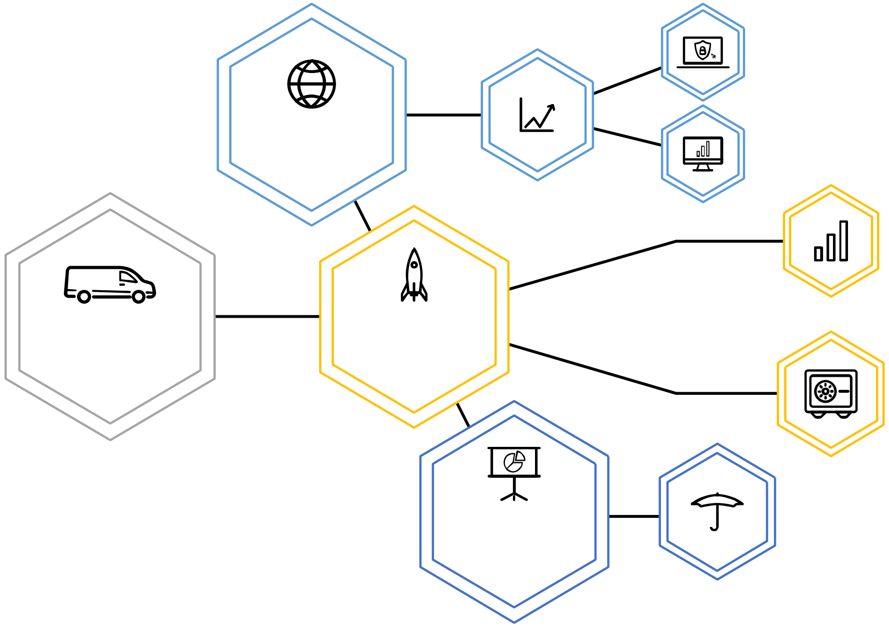




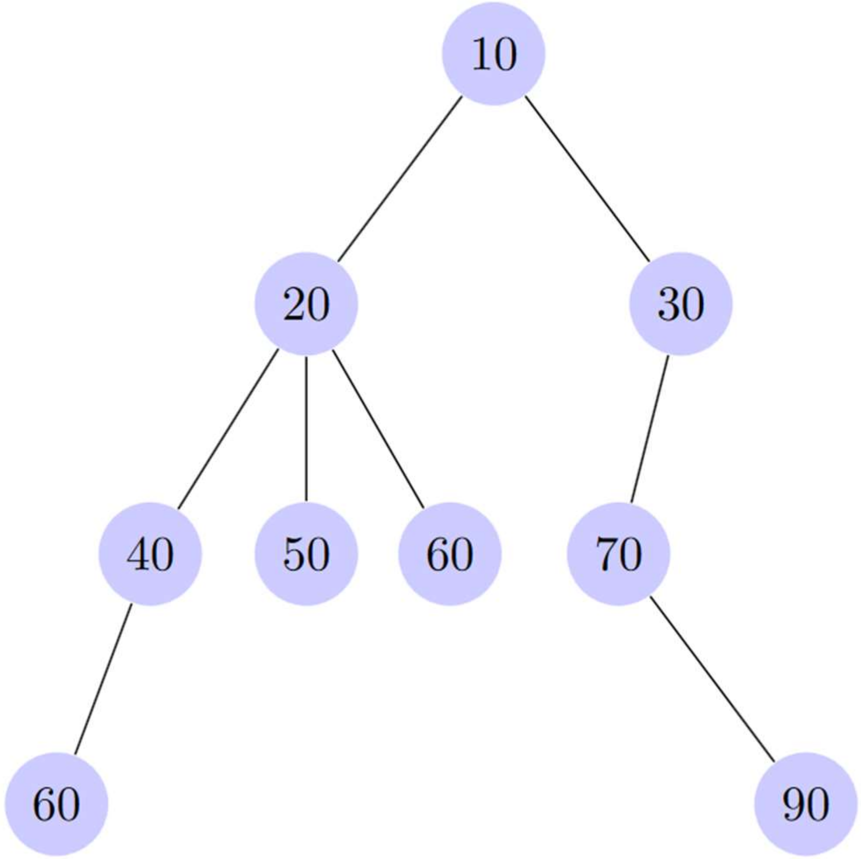
AV121: Data Structures and Algorithms

Tutorial-01

Tutorial 01 – Plan

- Set operations
- Tree Basics
  - → Height, siblings, ancestor, depth
- Tree Traversals
  - → Inorder, Postorder, Preorder

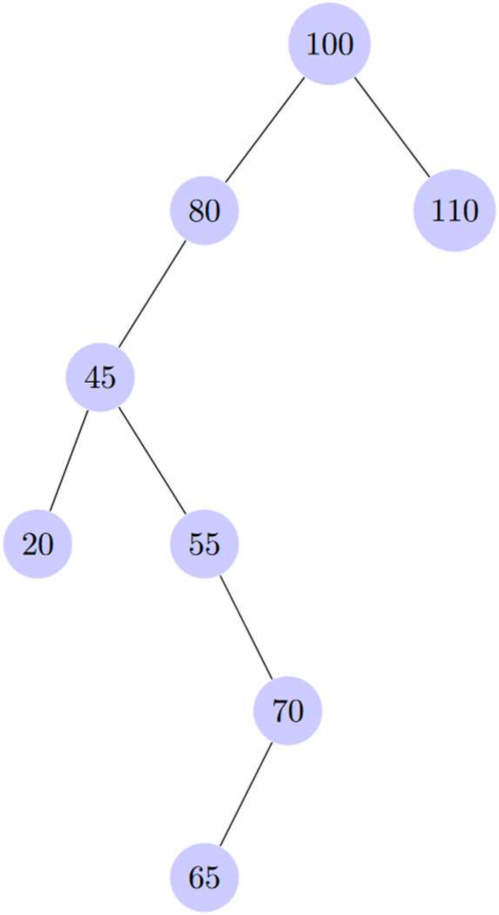
Tree Terminologies



Root	10
Parent of Node 50	20
Siblings	{40,50,60} and {20,30}
Leaf nodes	60,90
Degree of Node 20	3
Internal Nodes	10,20,30,40,70
Depth of Node 60	2
Descendants of 70	70,90
Ancestors of 70	70,30,10
Height of tree	3
Degree of tree	3

Array representation of binary tree

- Show the array representation of following binary tree



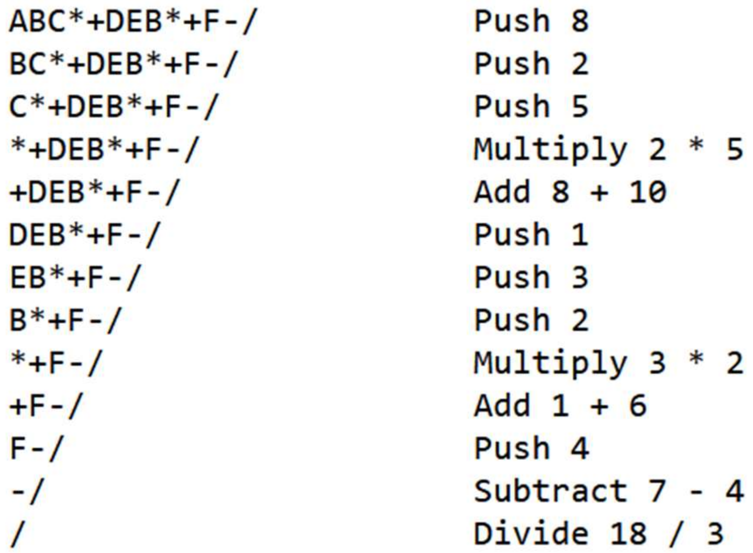
Index	Value
0	100
1	80
2	110
3	45
.	.
7	20
8	55
.	.
18	70
.	.
37	65

Left(n)=2*n+1
Right(n)=2*n+2
Parent(n)=floor((n-1)/2)

Infix to Postfix

Infix	Postfix
A+B*C	ABC*+
A*B+C	AB*C+
A*B/C	AB*C/
A*B^C+D	ABC^*D+
((A+B)*C+D)/(E+F+G)	AB+C*D+EF+G+/

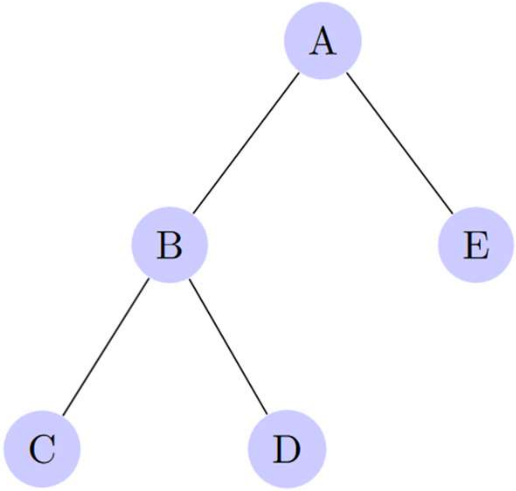
- Let A,B,C,D,E,F=8,2,5,1,3,4. Evaluate ABC*+DEB*+F-/ 6



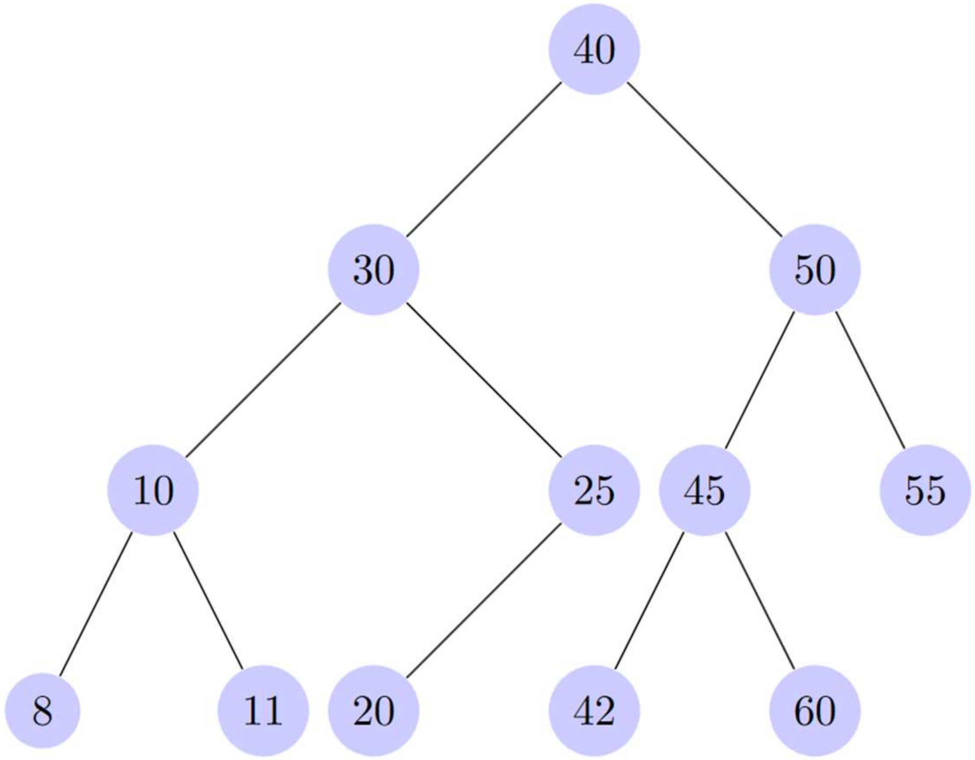
```
ABC*+DEB*+F-/          Push 8
BC*+DEB*+F-/           Push 2
C*+DEB*+F-/            Push 5
*+DEB*+F-/             Multiply 2 * 5
+DEB*+F-/              Add 8 + 10
DEB*+F-/               Push 1
EB*+F-/                Push 3
B*+F-/                 Push 2
*+F-/                  Multiply 3 * 2
+F-/                   Add 1 + 6
F-/                    Push 4
-/                     Subtract 7 - 4
/                      Divide 18 / 3
```

Tree Traversals

- Write inorder, postorder and preorder traversal of following trees



Inorder : CBDAE
Preorder : ABCDE
Postorder : CDBEA



Inorder : 8 10 11 30 20 25 40 42 45 60 50 55
Preorder : 40 30 10 8 11 25 20 50 45 42 60 55
Postorder : 8 11 10 20 25 30 42 60 45 55 50 40

Tree Traversals

- From which of the following traversals we can construct a binary tree
  - Preorder and postorder
  - Levelorder and preorder
  - Levelorder and postorder
  - Inorder and preorder ⬅
  - Inorder and postorder ⬅
  - Inorder and levelorder ⬅
- Construct tree where inorder = g d h b e i a f j c and preorder = a b d g h e i c f j
- Construct tree where inorder = g d h b e i a f j c and postorder = g h d i e b j f c a
- Construct tree where inorder = g d h b e i a f j c and levelorder = a b c d e f g h i j