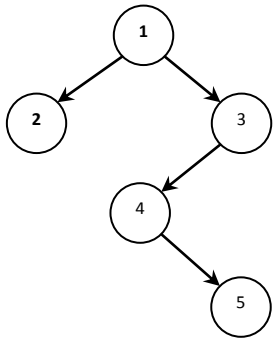
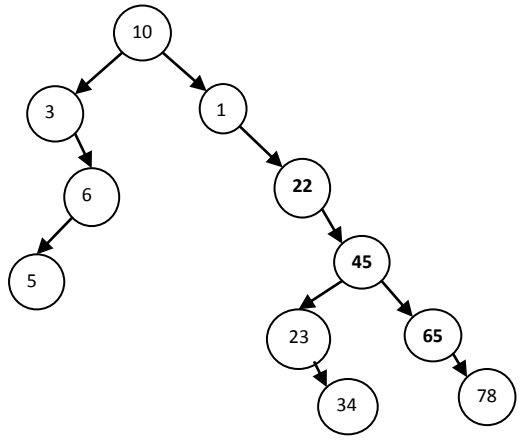
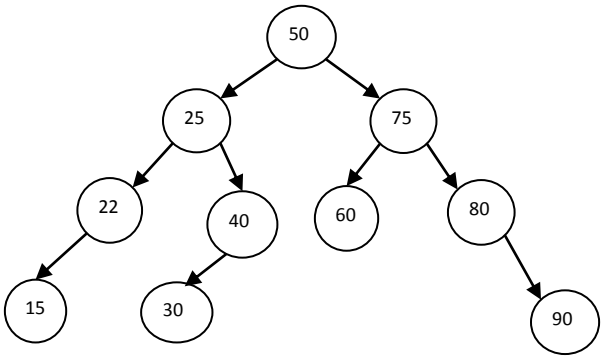
## Traverse given Tree in In-order, Pre-order and Post-order

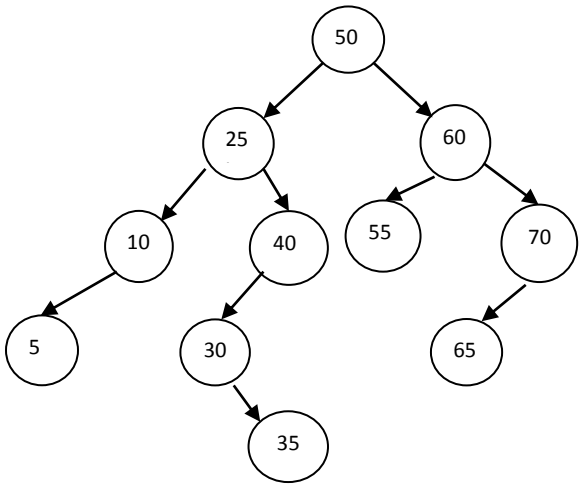


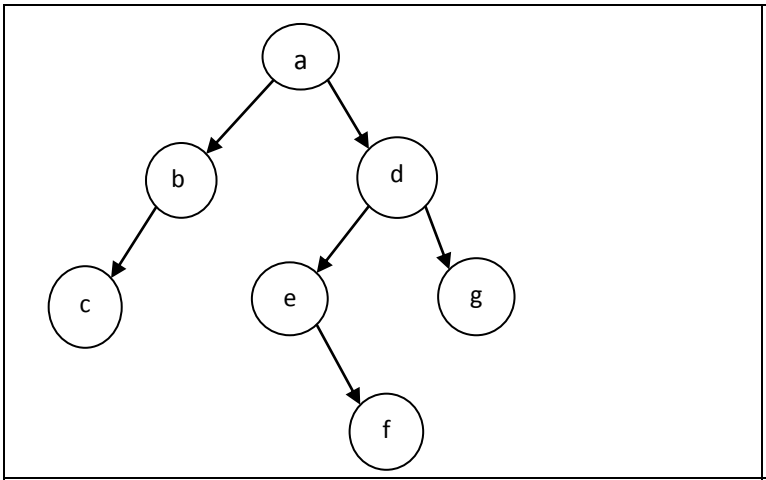

Preorder:
Inorder:
Postorder:

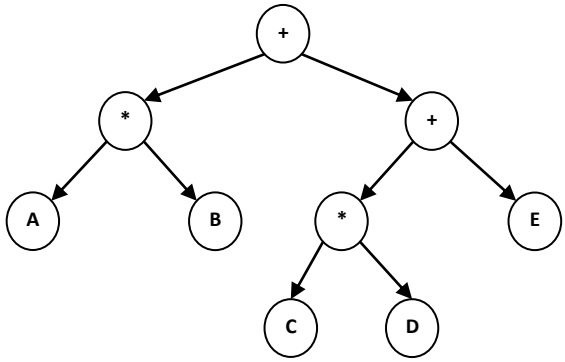


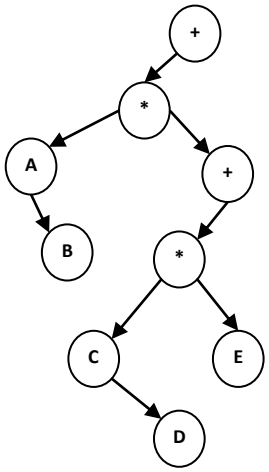

Preorder:
Inorder:
Postorder:

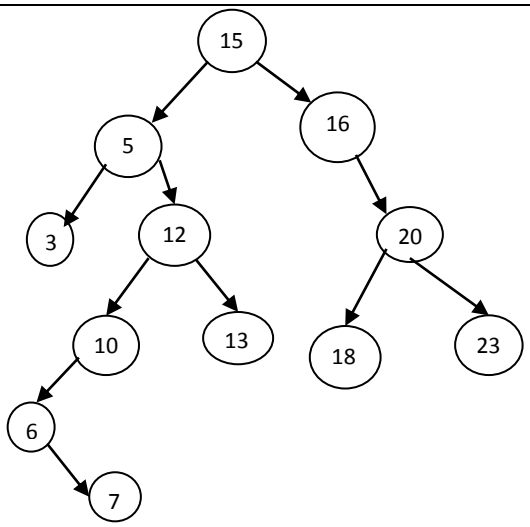


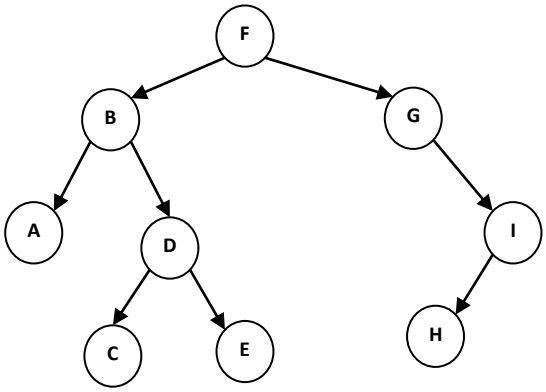



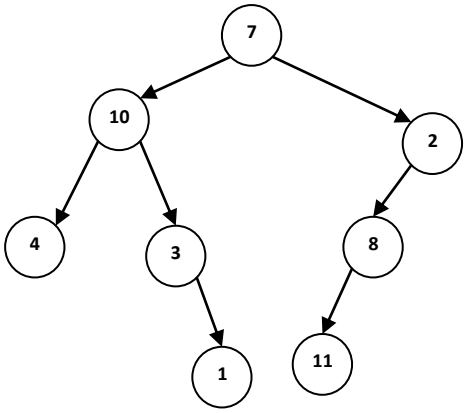

a
b
d
c
e
g
f

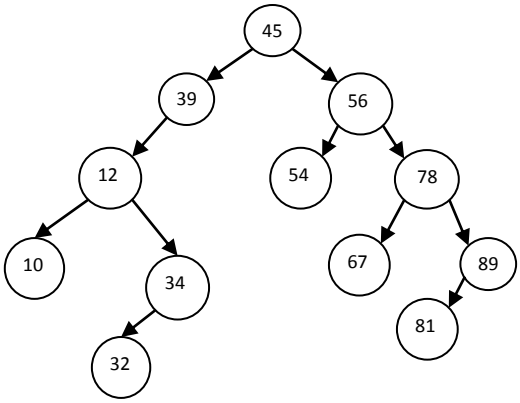

+
*
+
A
B
*
E
C
D

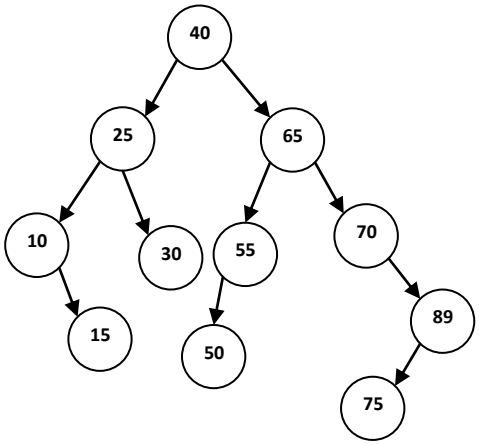

+
*
A
+
B
*
C
E
D

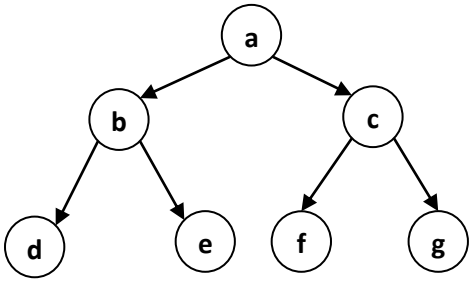

15
5
16
3
12
20
10
13
18
23
6
7

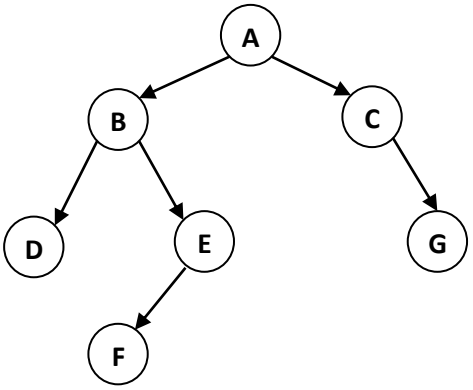

F
B
G
A
D
I
C
E
H

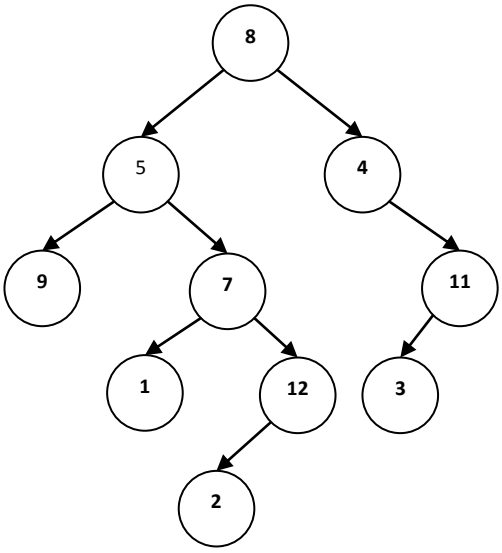

7
10
2
4
3
8
1
11


45
39
56
12
54
78
10
34
67
89
32
81


40
25
65
10
30
55
70
15
50
89
75


a
b
c
d
e
f
g


A
B
C
D
E
G
F


8
5
4
9
7
11
1
12
3
2

## Construct a tree for the given In-order and Pre-order OR In-order Post-order traversals

1. Inorder: DGBAHEICF
   Postorder: GDBHIEFCA
2. Inorder: EACKFHDBG
   Preorder: FAEKCDHGB
3. Inorder: QBKCFAGPEDHR
   Preorder: GBQACKFPDERH
4. Inorder: BIDACGEHF
   Postorder: IDBGCHFEA
5. Inorder: BCAEDGHFI
   Postorder: CBEHGIFDA
6. Inorder: d,b.e,a,f,c,g
   Preorder: a,b,d,e,c,f,g
7. Inorder: 4,10,3,1,7,11,8,2
   Preorder: 7,10,4,3,1,2,8,11
8. Inorder: 1, 10, 11, 12, 13, 14, 15, 17, 18, 21
   Postorder: 1, 11, 12, 10, 14, 18, 21, 17, 15, 13
9. Inorder: 1 3 4 6 7 8 10 13 14
   Preorder: 8 3 1 6 4 7 10 14 13
10. Inorder traversal = {4, 2, 5, 1, 3, 6}
    Preorder traversal = {1, 2, 4, 5, 3, 6}
11. Inorder: 3 4 5 6 7 9 17 20 22
    Preorder: 9 4 3 6 5 7 17 22 20
11. Preorder Traversal: 7, 1, 0, 3, 2, 5, 4, 6, 9, 8, 10
    Inorder Traversal: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10

## Draw a Binary expression tree for the following and perform preorder traversal

1. (A $ B $ C) + (D – E * F)
2. (A + B $ C) + (D + E * F)
3. v1*v2-(v3+v4^v5).

**Draw tree whose postorder traversal is C B F E G D A**