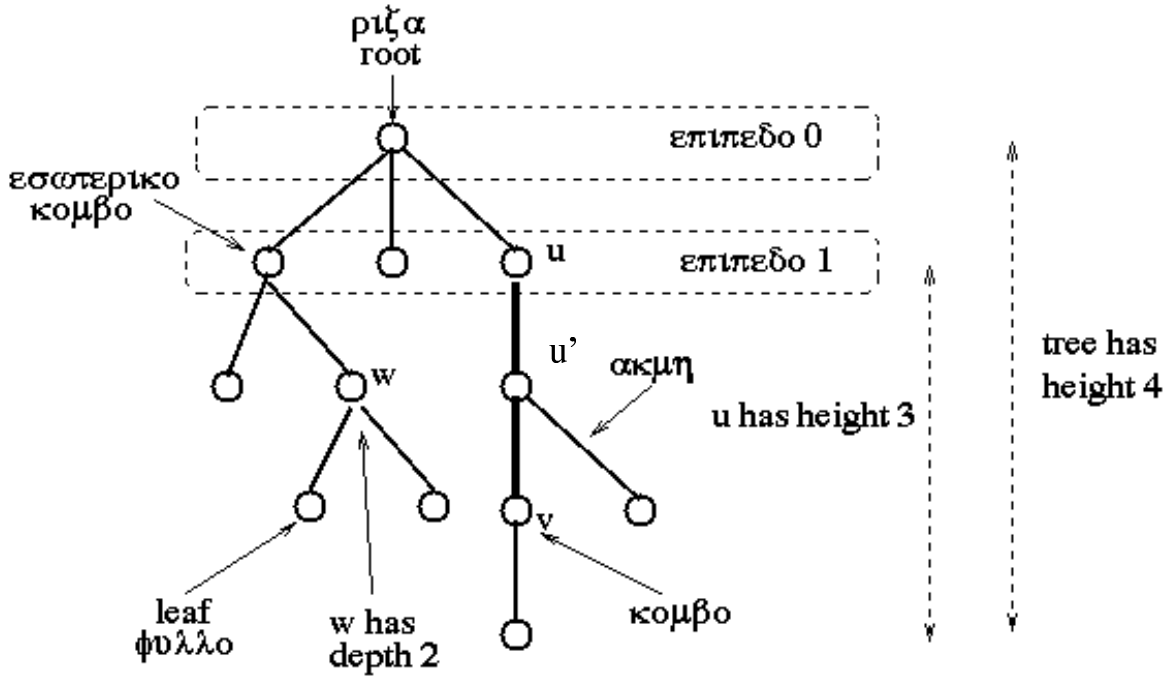
□/□ 4-35, □□□□

□□□□□□

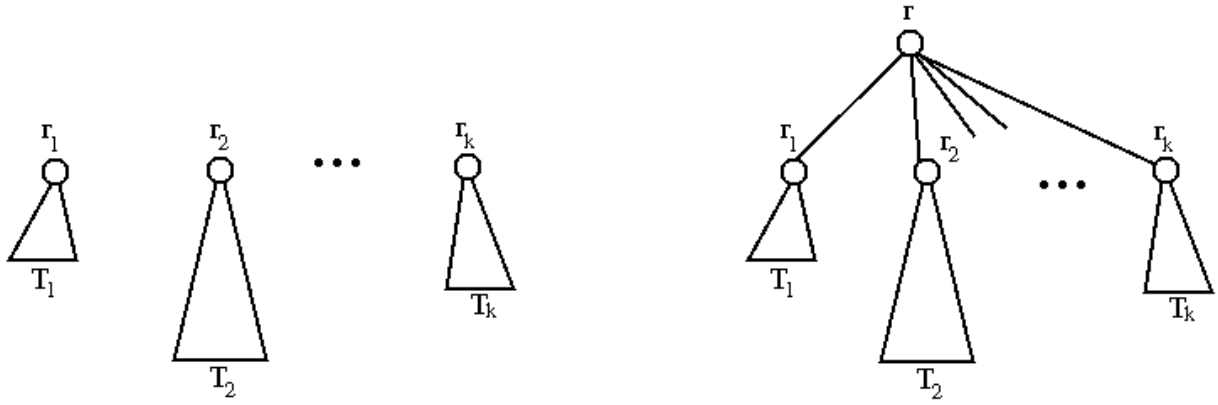

ριζα
root
επιπεδο 0
εσωτερικο
κομβο
επιπεδο 1
u
u'
ακμη
w
u has height 3
tree has
height 4
v
leaf
φυλλο
w has
depth 2
κομβο

- □□□□□□(nodes)
- □□□□□(edges)
- □□□□□□□□□□□□□□□□

□□□□□□□□□□□□(node degree)

□ □□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□□□□□□


r
r1
r2
rk
T1
T2
Tk

□□□□□□□□□□□□□□□□□□□

□□□□□□□□□□

□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□□□□□□□□□□□□□□

□□□□□□□

□□□□□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□

□/□ 4-35, □□□□

□□□□□□□□□□□□□□

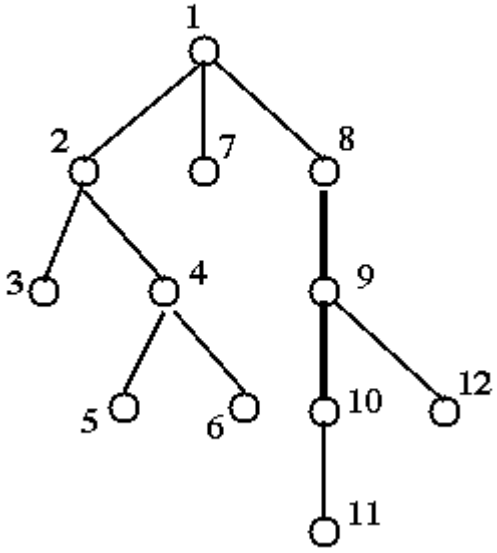
Visit(P): □□□□□□□□□□□□□□□□

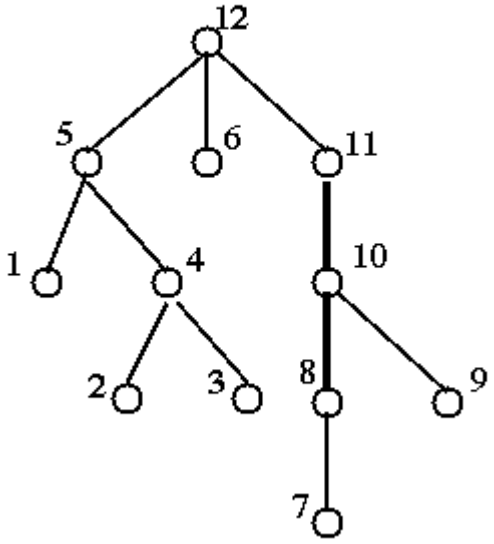
□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□

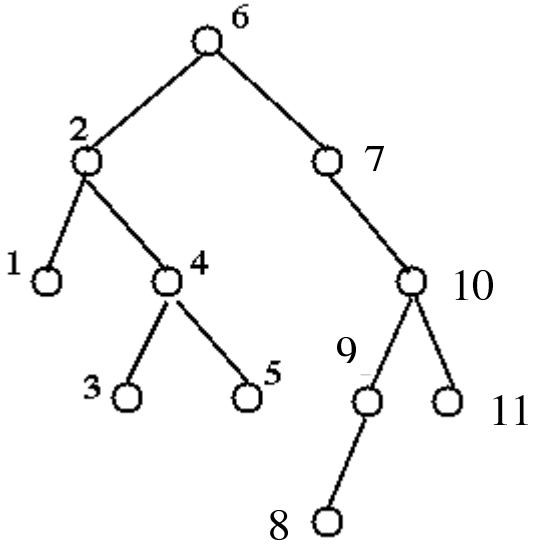
□□□□□□□□□□□□

□□□□□□□□□□□□□□□□□ □□□□□□□□□□□□□□□□□


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


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

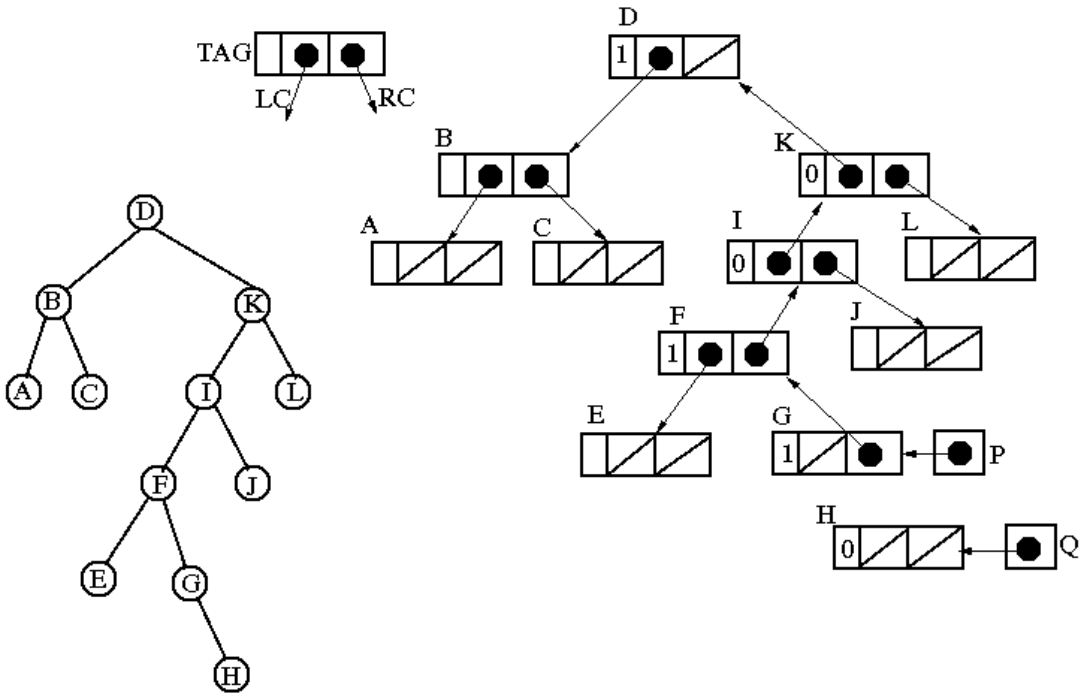
□□□□□□□□□□□□□□□□□


6
2
7
1
4
10
9
3
5
11
8

□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□

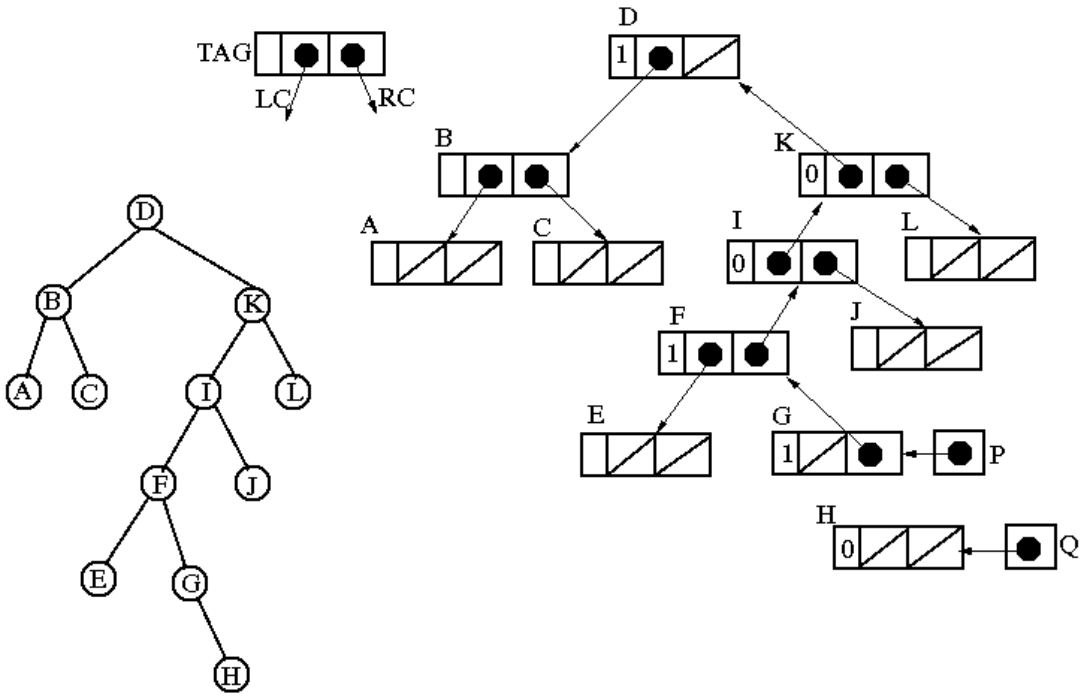
□□□□□□□□□□□□□□□□□□□□



Procedure *LinkInversionTraversal(pointer Q):*
/* Initially Q points to the root of the tree to be traversed */

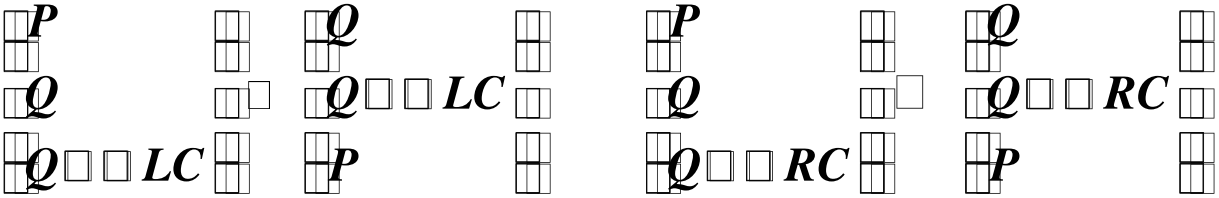
```
P = NULL;
while (1)
        while (Q != NULL) do
                Visit(Q)
                Tag(Q) = 0
                descend to left
        while (P != NULL and Tag(P) = 1) do
                ascend from right
                Visit(Q)
        if (P == NULL) then return
        else
                ascend from left
                Visit(Q)
                Tag(Q) = 1
                descend to right
```

□□□□□□□□□□□□□□□□□□□□



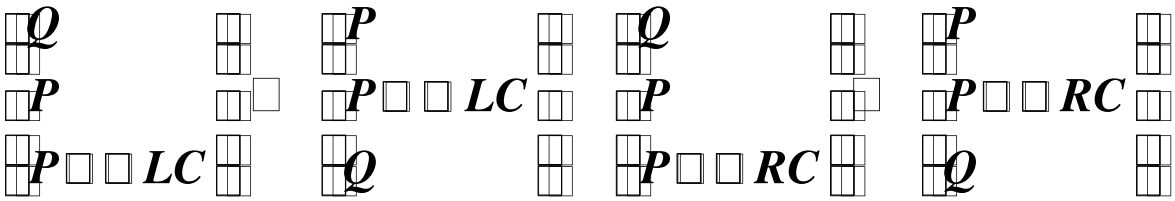
descend to left:

descend to right:



ascend from left:

ascend from right:

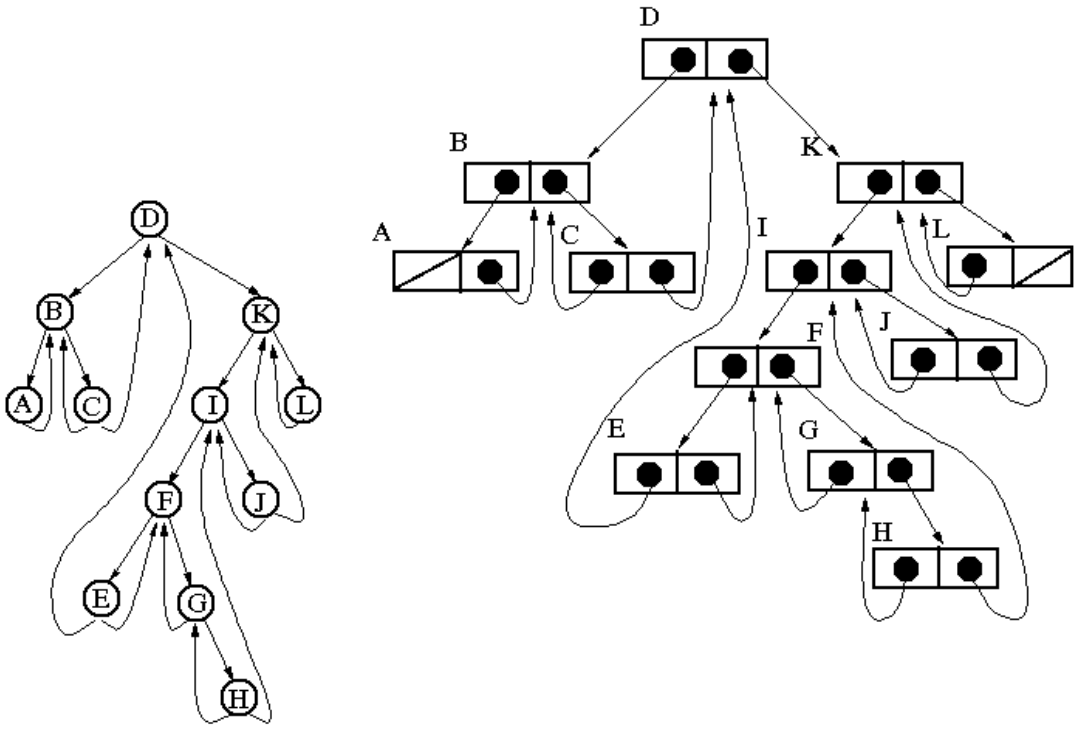


□□□□□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□□□□□□

- □□□□□□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□□□□□□



function *InorderSuccessor(pointer N):* **pointer**
/* returns the inorder successor of node N, or □ if N has none */

```
P = N->RC
if (P = NULL) then return NULL
else if (P is not a thread) then
        while (P->LC is not a thread or NULL) do
              P = P->LC
return P
```

□□□□□□□□□□□□□□□□

□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□

□□□□□□□□□

- □□□□□□□□□□□□□□□□□□□□