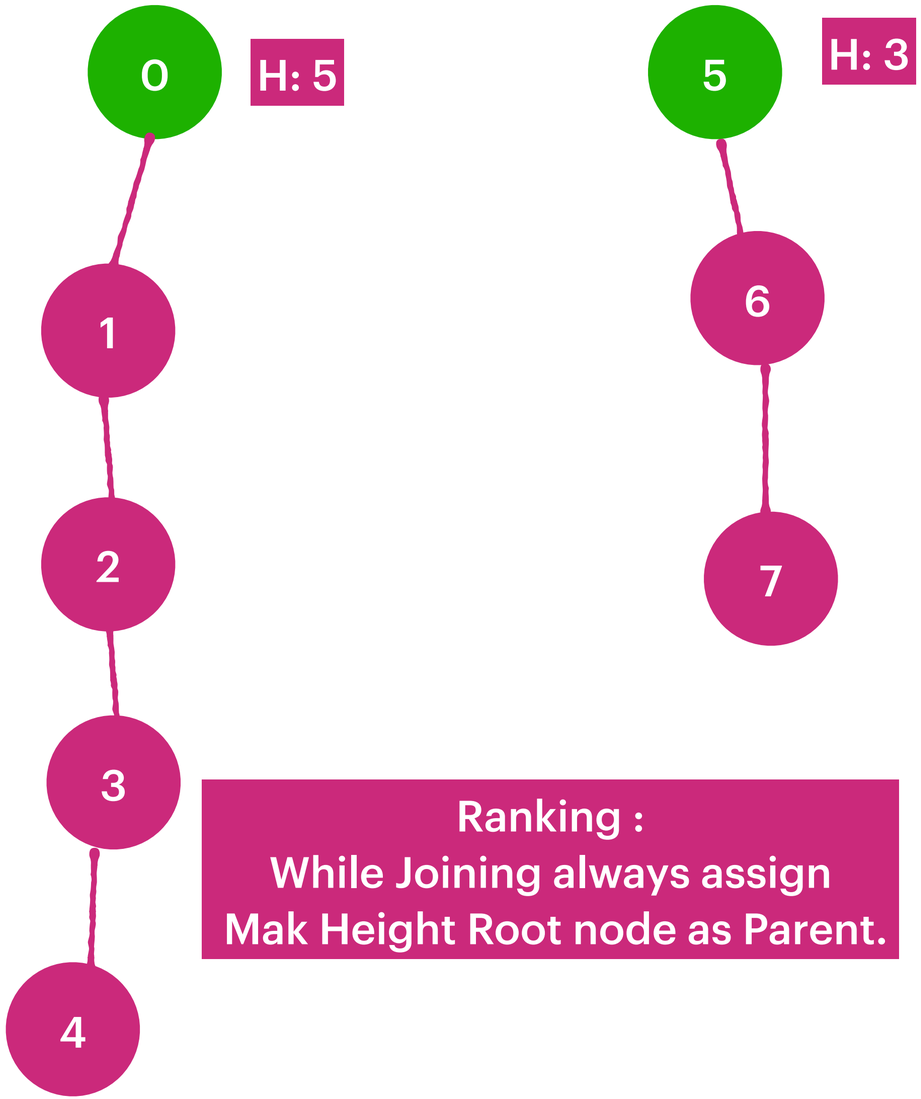
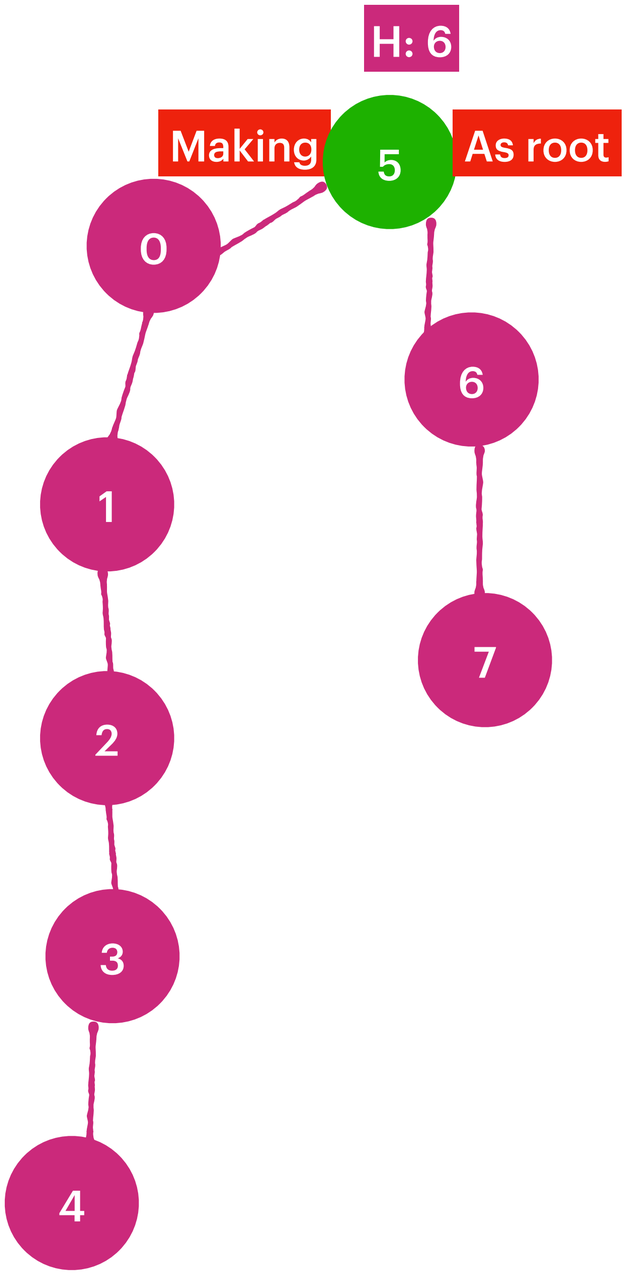
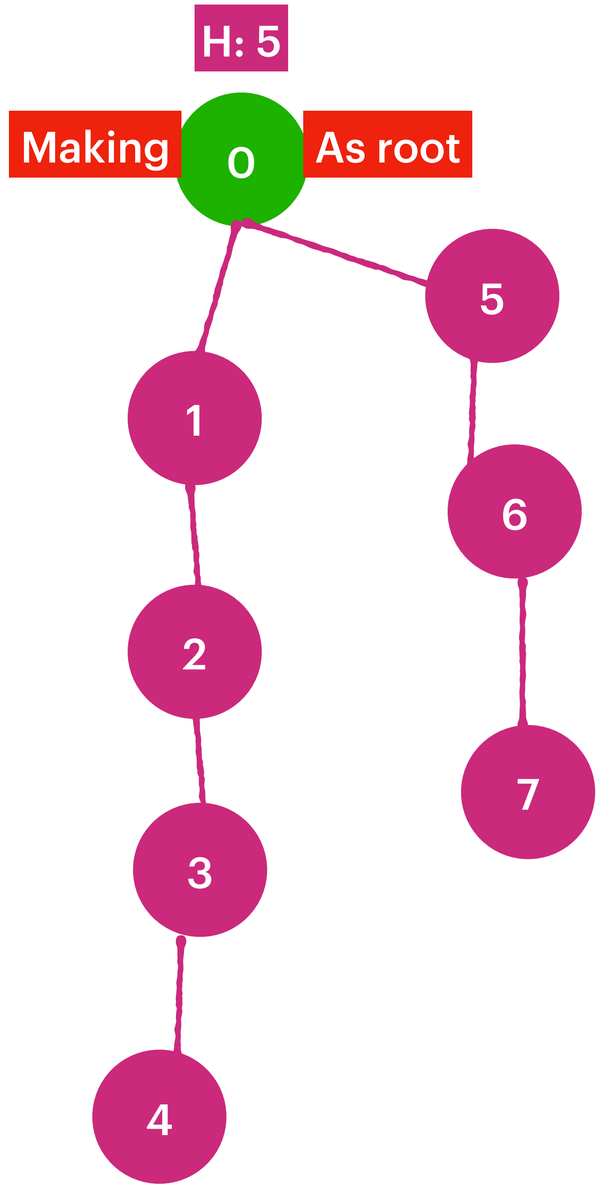
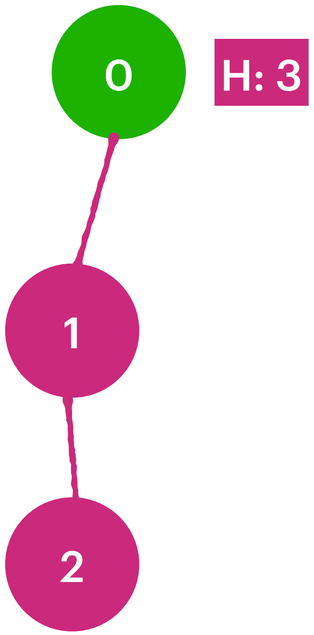
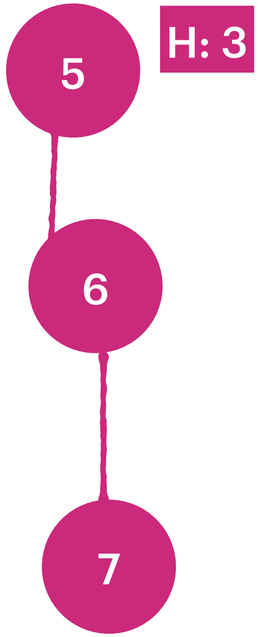
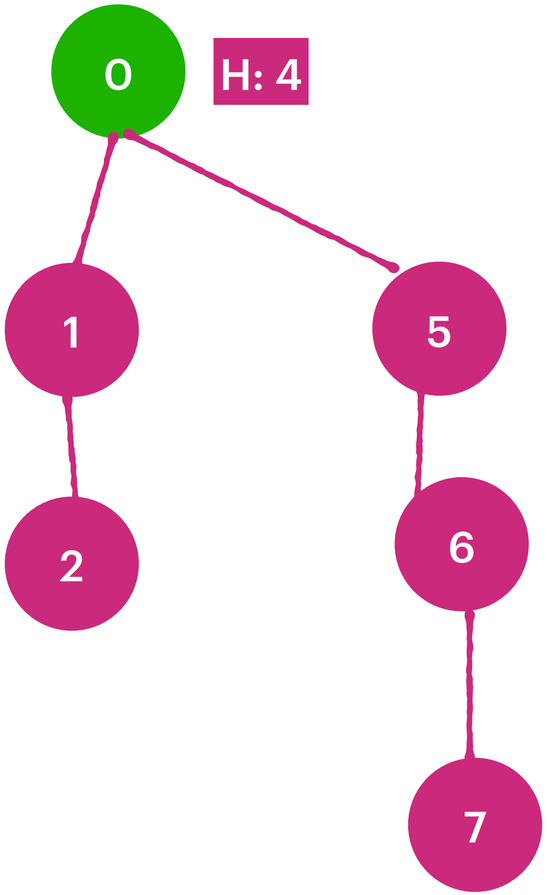

0
H: 5
1
2
3
4
5
H: 3
6
7
Ranking :
While Joining always assign
Mak Height Root node as Parent.

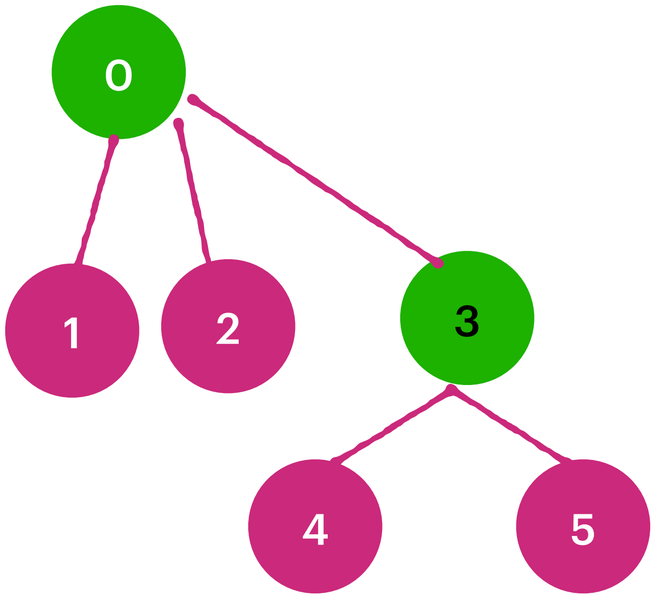

H: 6
Making
5
As root
0
6
1
7
2
3
4

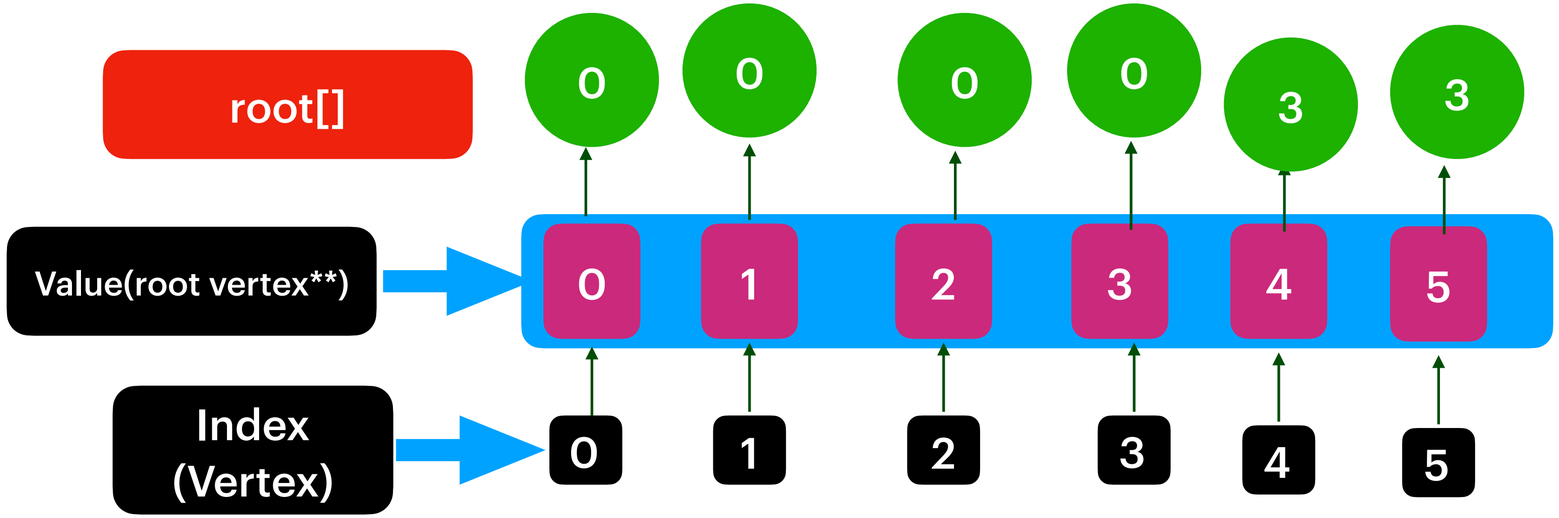

H: 5
Making
0
As root
5
1
6
2
7
3
4

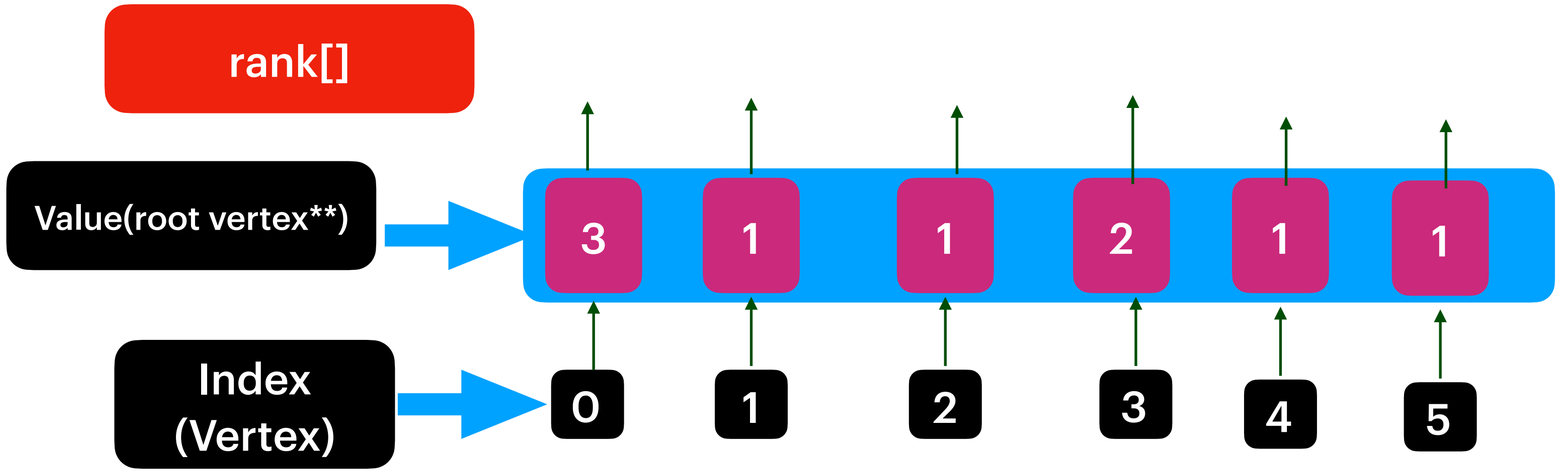

0
H: 3
1
2

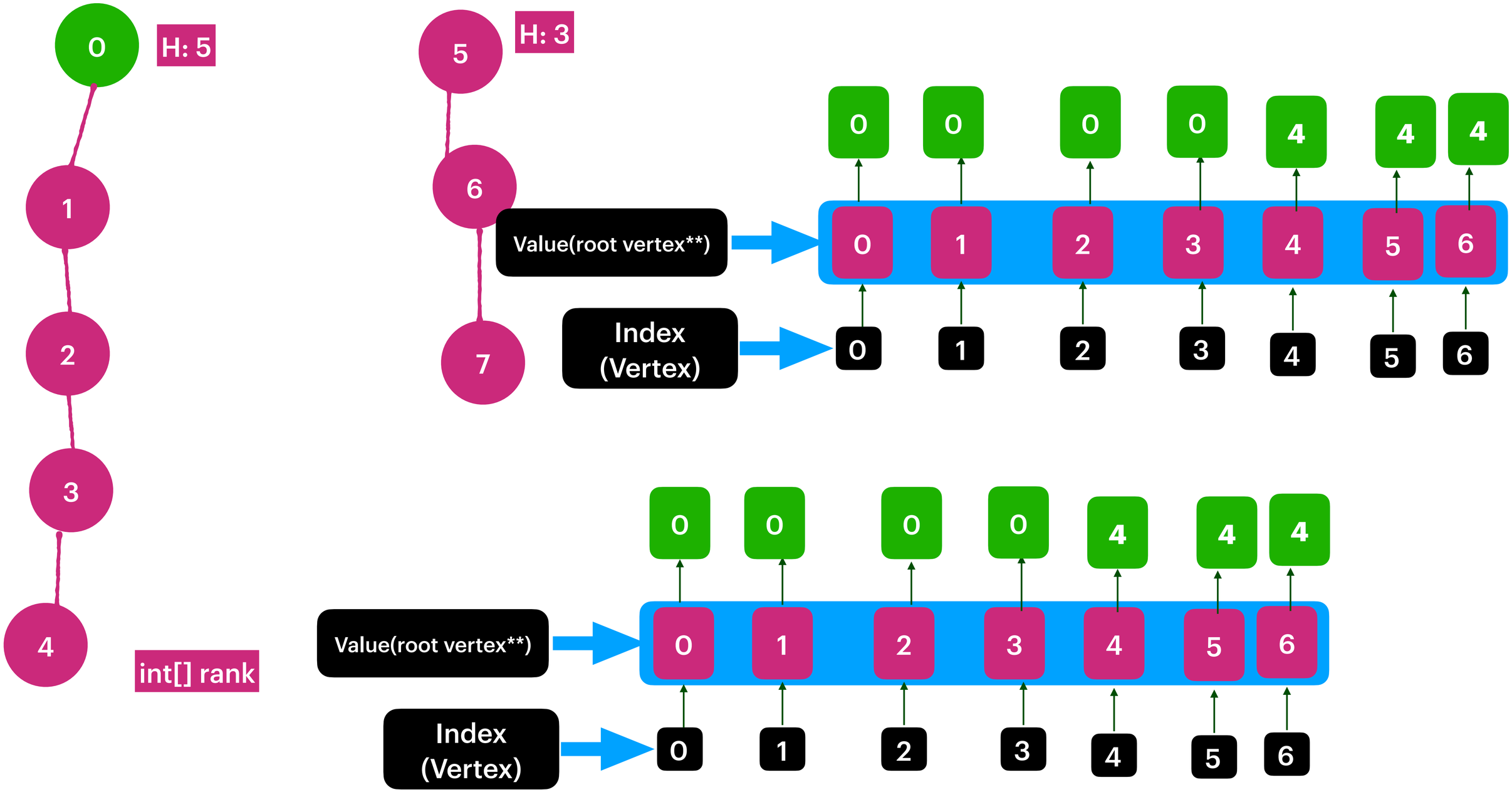

5
H: 3
6
7

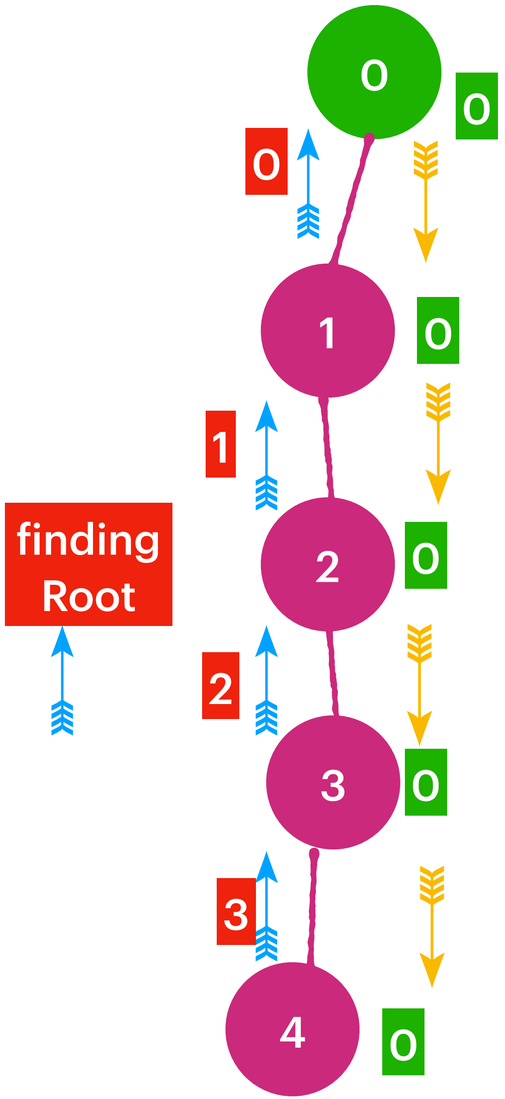

0
H: 4
1
5
2
6
7


0
1
2
3
4
5


root[]
Value(root vertex**)
Index
(Vertex)
0
0
0
0
3
3
0
1
2
3
4
5
0
1
2
3
4
5


rank[]
Value(root vertex**)
Index
(Vertex)
3
1
1
2
1
1
0
1
2
3
4
5


0
H: 5
1
2
3
4
int[] rank
5
H: 3
6
7
Value(root vertex**)
Index
(Vertex)
0 0 0 0 4 4 4
0 1 2 3 4 5 6
0 1 2 3 4 5 6
Value(root vertex**)
Index
(Vertex)
0 0 0 0 4 4 4
0 1 2 3 4 5 6
0 1 2 3 4 5 6



O(H)

Path Optimization

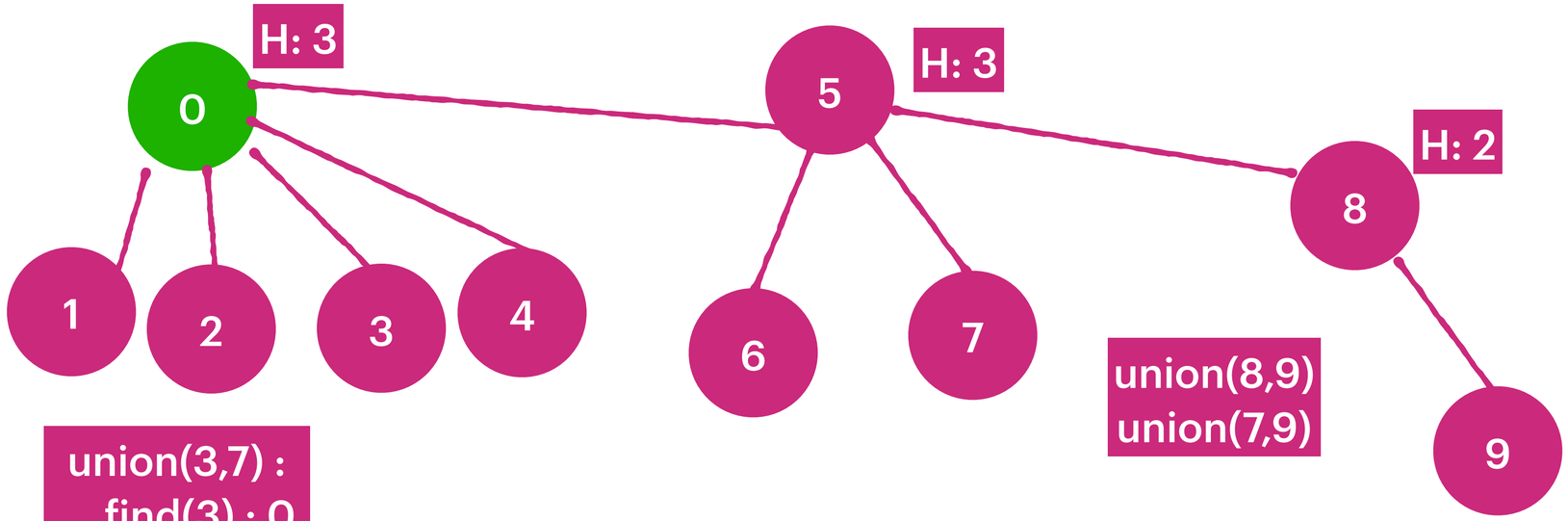
While traversing back, update the child vertex root.

Update the vertex value root

Connect(0,4)

Find(0) : 0

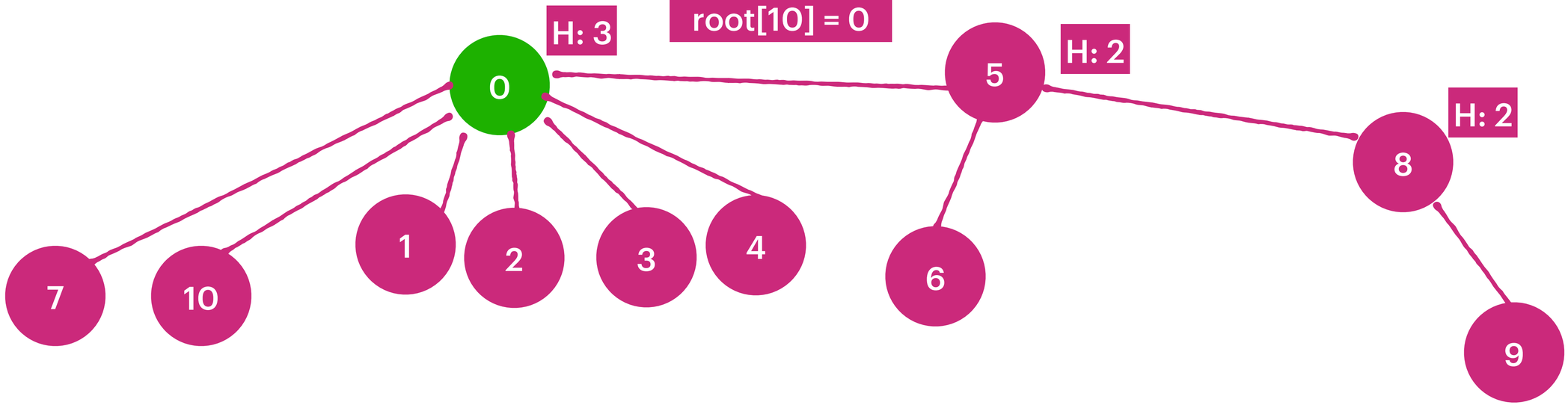
find(4) : 4 -> 3 -> 2 -> 1 -> 0 : 0



union(3,7) :
find(3) : 0
find(7) : 5
root[5] = 0

union(8,9)
union(7,9)

isConnected(1,9) :

find(1) : 0 // 1step
find(9): 8->5->0 : 0 // 3steps
returns true.

union(7,10)
find(7) : 5 -> 0
find(10) : -> 10
root[10] = 0