

```
class Node {
    int data;
    int left;
    int right;
}
```

created  
Node  
(or a default)

Tzinach thought  
process

① How to align a tree

In such a way to satisfy  
the specific conditions like

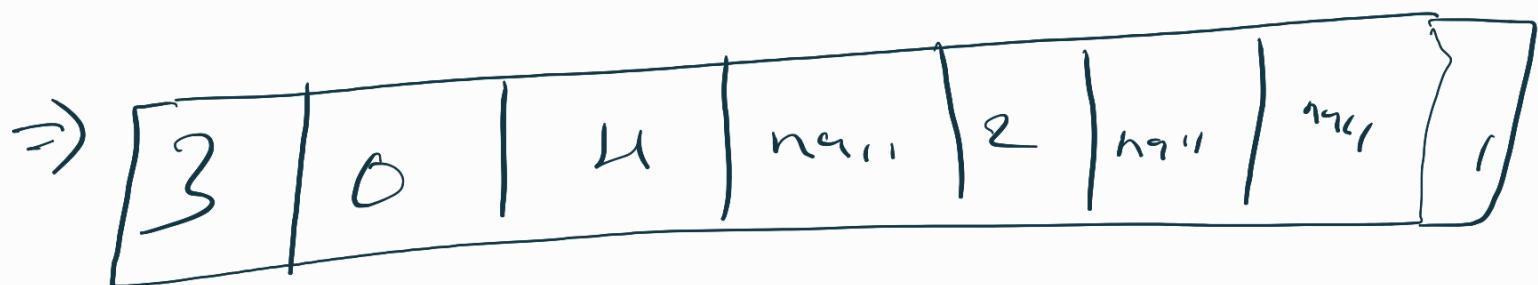
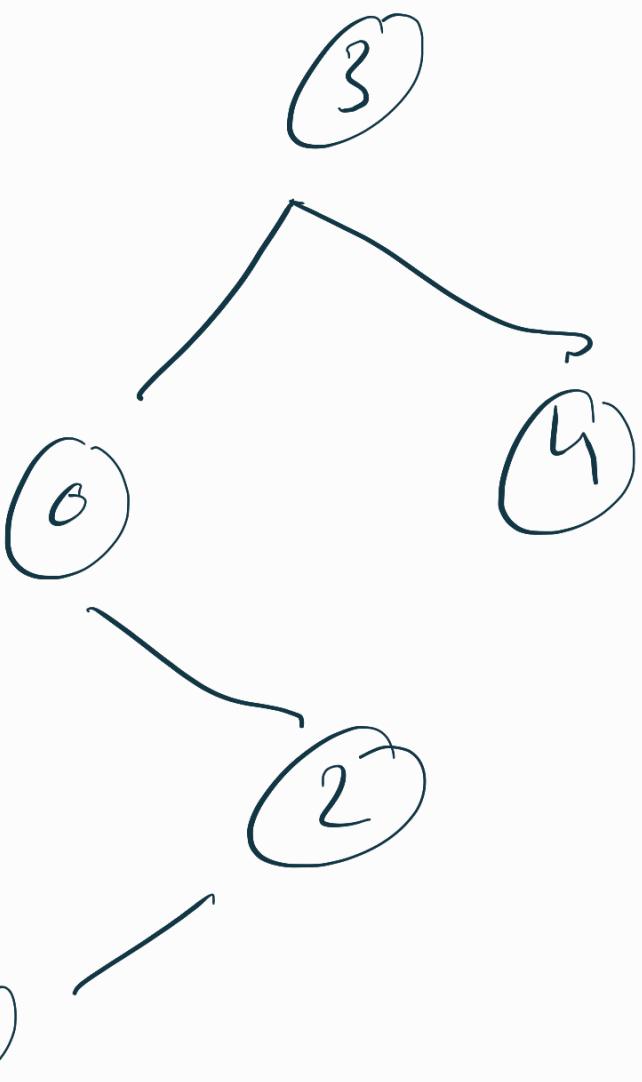
B.T, max, min ...

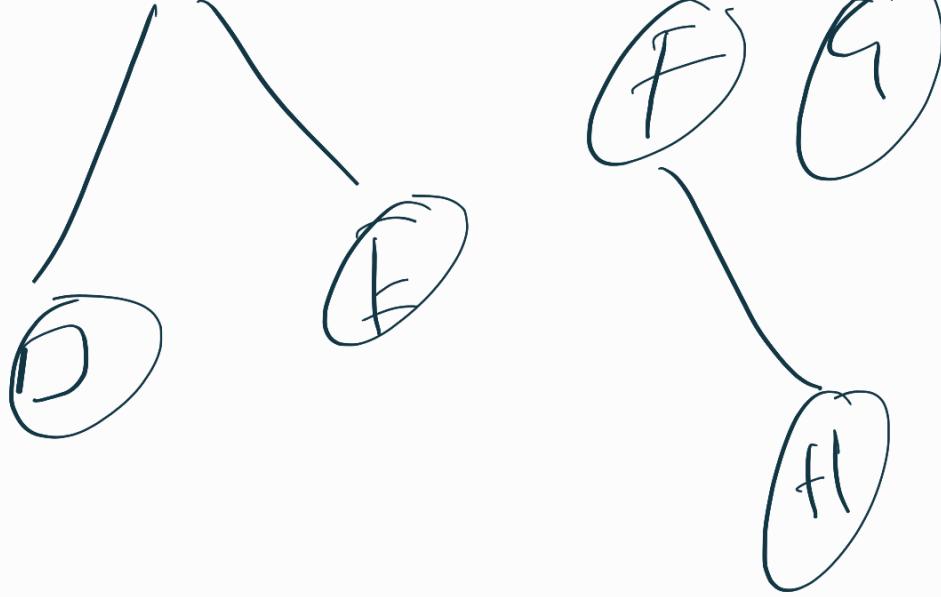
Observations

① Every node has only  
two sub nodes  
one is at left side

& another one at right  
side.

- ② How to compare the nodes  
& How to insert the values  
in proper order?
- ③ How the correlation  
maintaining in-between the  
nodes (descendant).
- ④ How to check whether  
the tree is in proper order  
or not / or whether it is of  
Complete tree.





```

class Node {
    int data;
    Node left;
    Node right;
}
  
```

The code defines a class `Node` with three members: `int data`, `Node left`, and `Node right`. A brace on the right side of the class definition groups the `left` and `right` members, with the word "action" written next to it. Another brace above the class definition groups the entire class body, with the word "Create" written next to it.

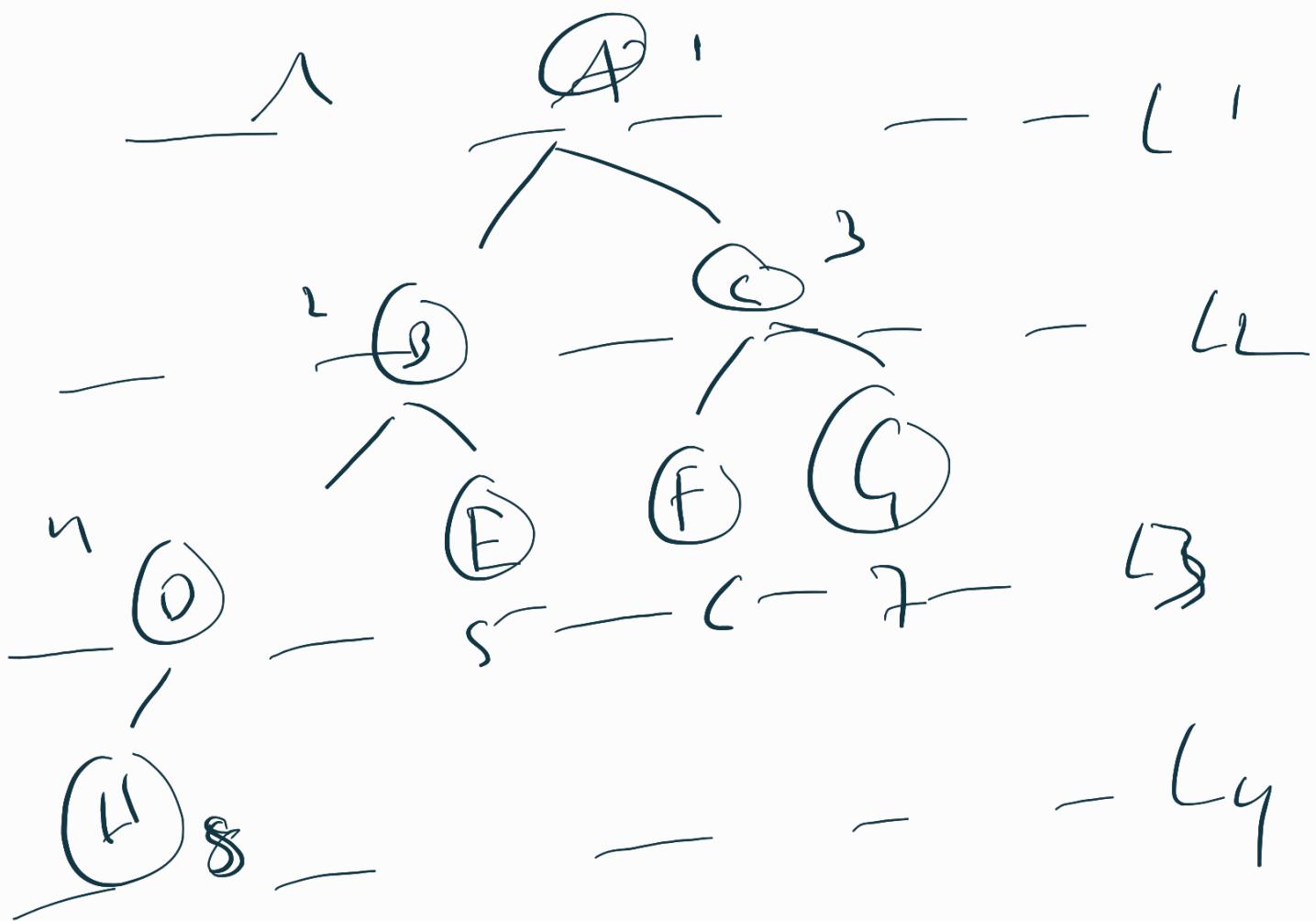
```

root == null;
=> root = Node
  
```

A horizontal line with an asterisk (\*) in the center is followed by the words "Day End".

Firstly, the given input is

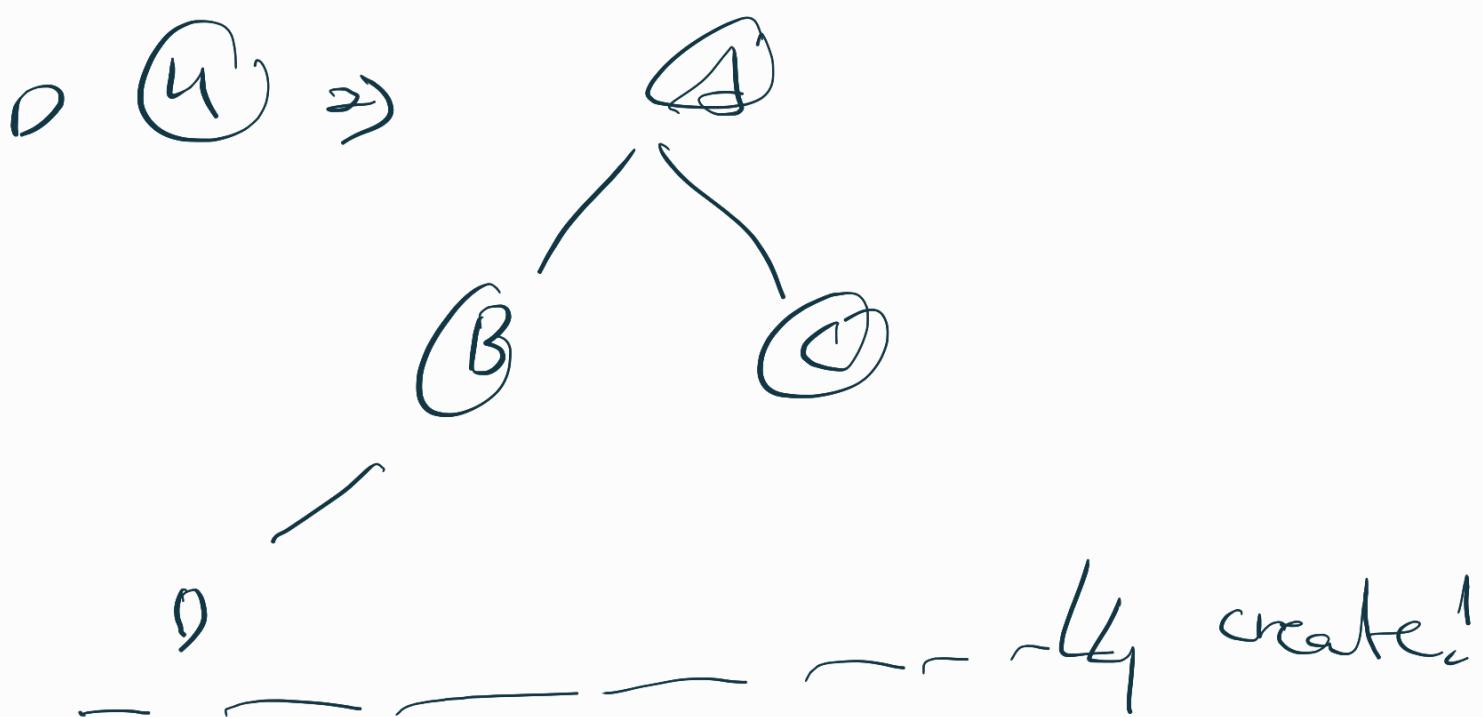
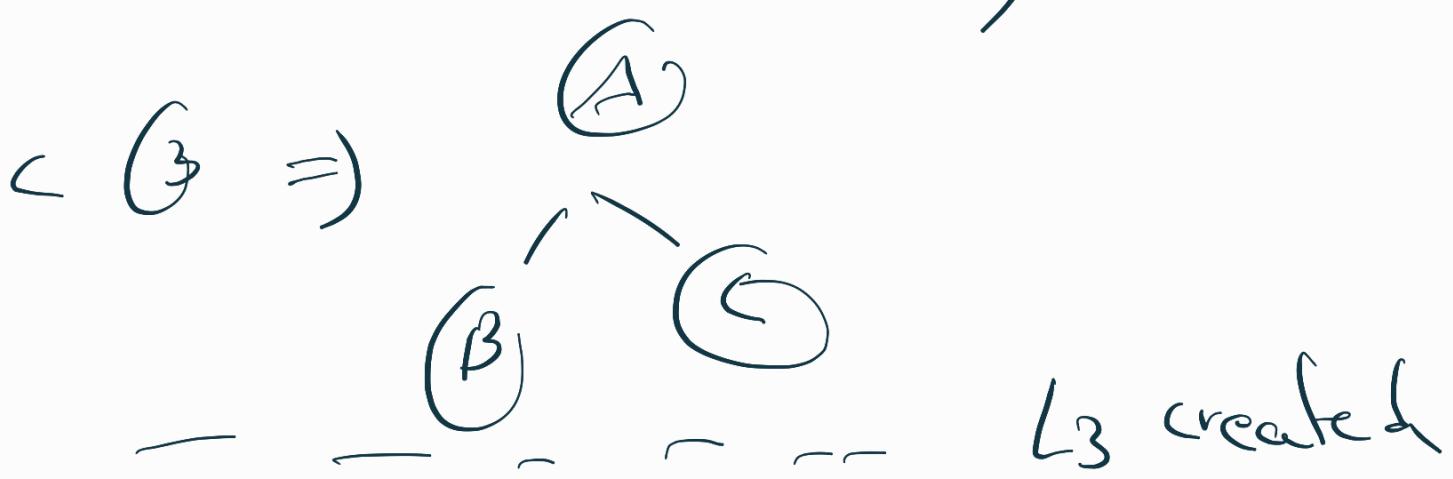
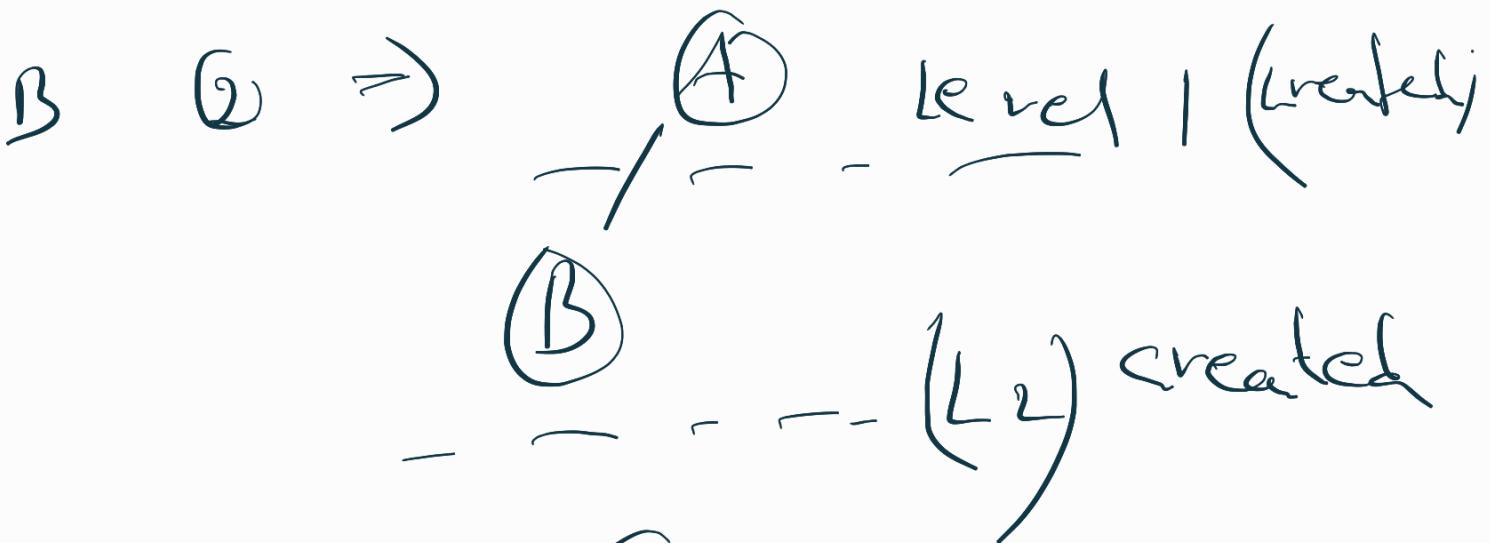
A	B	C	D	E	F	G	H
1	2	3	4	5	6	7	8



create node;

insert left node;

insert right node;



```
class Node {
    int data;
    Node left;
    Node right;
}

void Node(int d) {
    data = d;
}
```

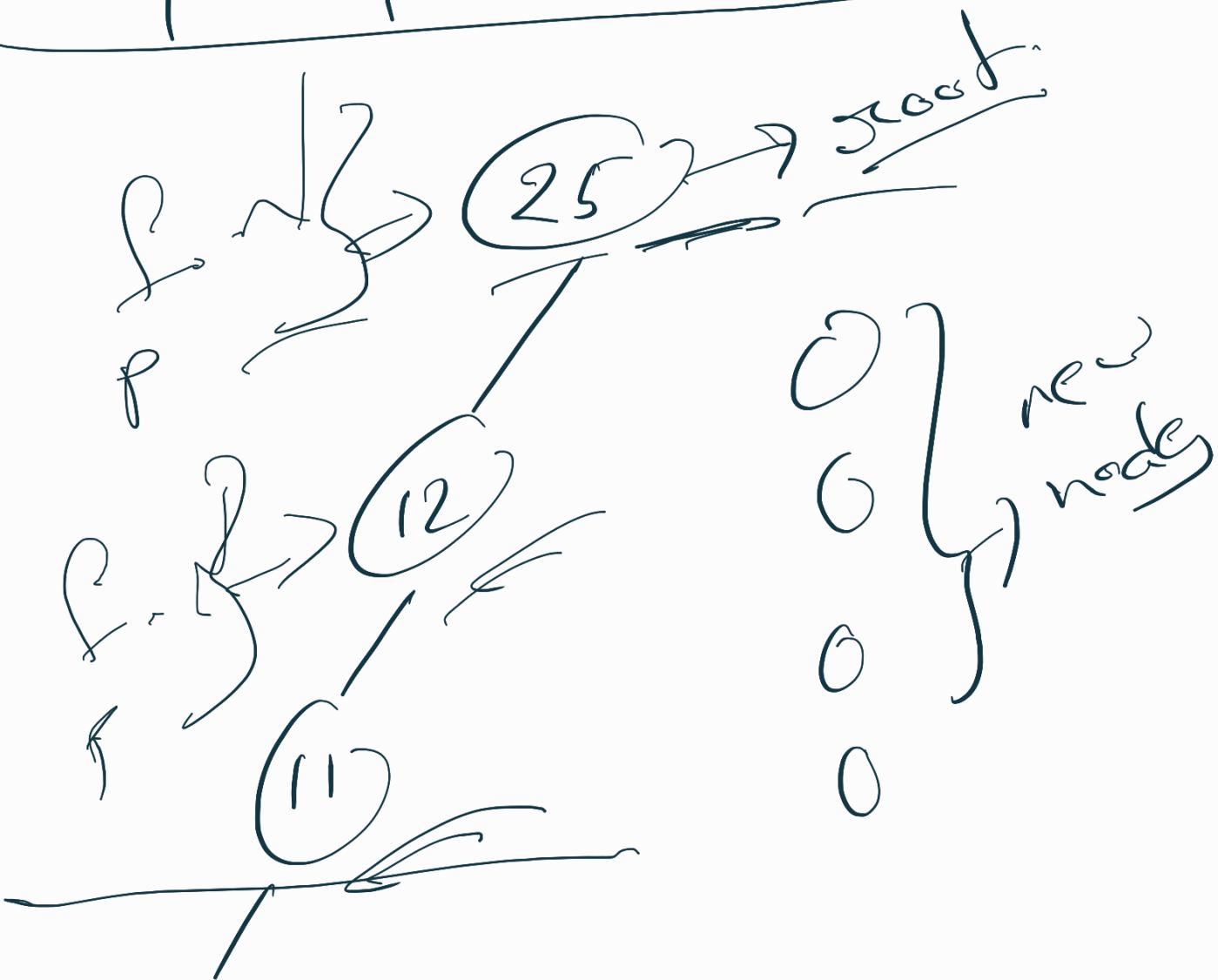
```
main() {
```

~~Node nd = new Node(15);~~

~~Node.left = nd?~~

?

25	12	11	10
----	----	----	----



(10)  
leaf -