1. **Terminology**
2. Root : tree 구조에서 제일 첫번째로 위치해 있는 노드(N)
3. Edge : tree에서 두 노드 사이를 연결시켜주는 선(N-1)
4. Parent : 부모 노드
5. Child : 자식 노드, 자식노드라고 해서 바로 밑의 노드 뿐만이 아닌 Parent의 밑에 있는 노드이면 다 Child라고 할 수 있다.
6. Siblings : 같은 부모노드를 지니고 있고, 같은 Level에 있는 노드들을 말한다.
7. Leaf(External Node) : tree에서 child 노드가 없는 노드들을 지칭한다.
8. Internal Nodes : child 노드가 하나라도 존재하는 노드를 의미한다.
9. Degree : 해당 노드와 바로 연결되어 있는 자식노드의 수를 뜻한다.
10. Level : 노드의 계층을 뜻한다. 다시 말해, 라인의 수를 말한다.?
11. Height : Leaf 노드에서 특정 노드까지의 edge의 수를 말함.
12. **Formulas**
13. 주어진 Node들을 이용한 Binary Tree의 수
14. Shape(Tree가 나올 수 있는 모양의 수) – 한번 그려보면 됨.

* 1개의 노드에서 나올 수 있는 모양은 1개이다.
* 2개의 노드에서 나올 수 있는 모양도 2개이다.
* 3개의 노드에서 나올 수 있는 모양은 5개이다.
* 따라서 이를 공식화 하면 이 도출 된다.

1. Permutation (각 노드에 들어가야할 value의 경우의 수)

* 2개의 노드가 있다면 2개의 value를 넣는 경우의 수는 2\*1개이다.
* 3개의 노드가 있다면, 3\*2\*1 개가 된다.
* 따라서 n개의 노드가 있다면, n! 가 될 것이다.

1. Max, Min Height or Node 구하기
2. Height가 주어졌을 경우

* Min Nodes : n=h+1
* Max Nodes : n=2h+1-1

1. Node가 주어졌을 경우

* Min Height : h=
* Max Height : h=n-1

1. Degree 구하기

* Deg(0) = external node
* Deg(0) = Deg(2)+1

1. Strict Binary Tree
2. Strict Binary Tree : Degree 가 2 or 0만을 가지고 있는 tree.
3. Height 가 주어졌을 때

* Min Nodes : n=2h+1
* Max Nodes : n=2h+1-1

1. Node 가 주어졌을 때

* Min Height : h=
* Max Height :

1. Internal node와 External node의 관계

* E=I+1