Permutation

3 4 5 6

4 — 3
3 3 3
4 5 6
5 6
6

4 5 6 3

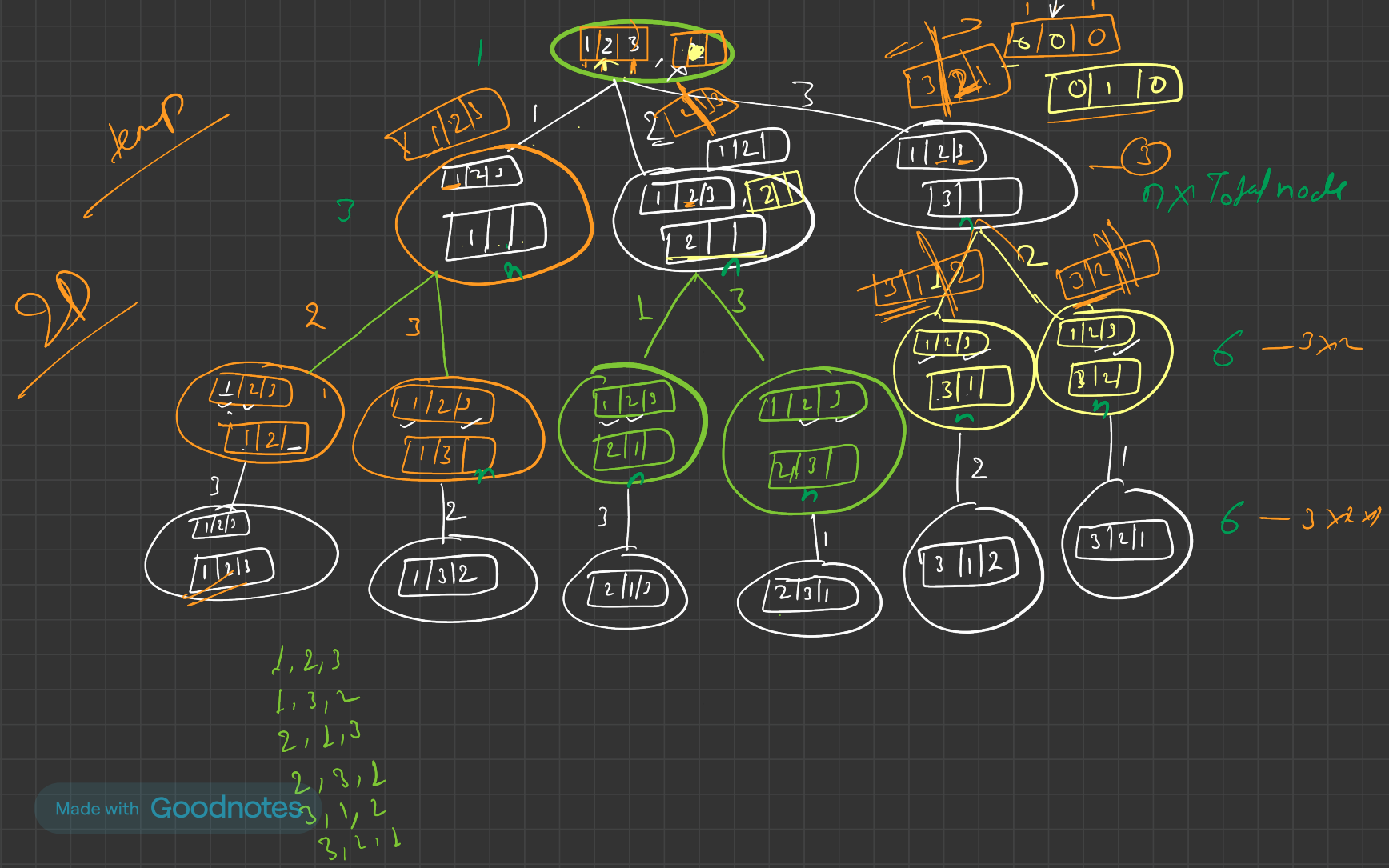
1 2 3
1 3 2
2 3 1
2 1 3
3 1 2
3 2 1

1! = 1
3! = 6
4! = 24
2! = 2
3! = 6

4

3 1 4

3 4
4 3

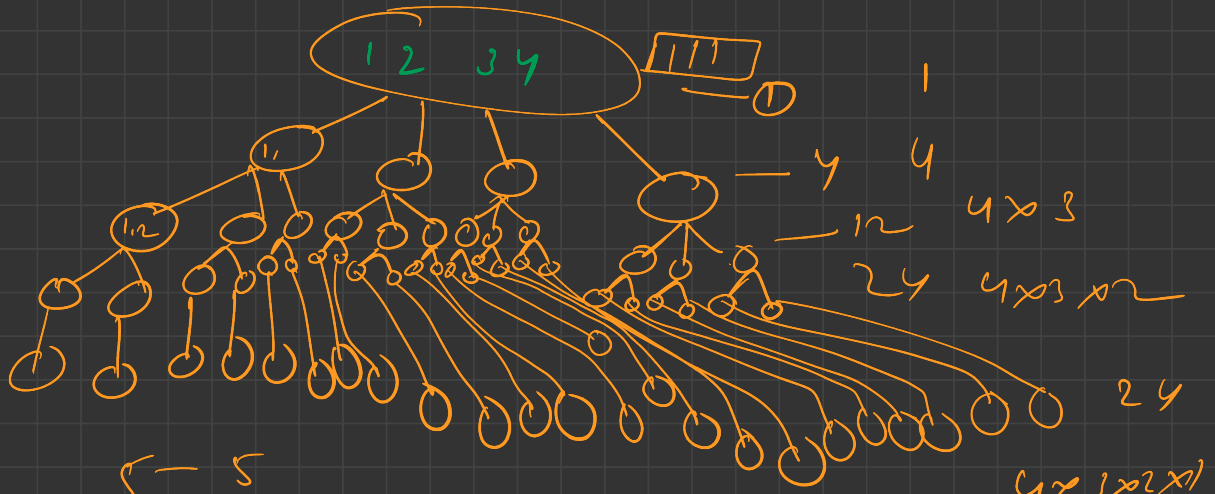


```
permut( int arr[], vector <vector< int >> & ans,
        vector < int > temp , vector < bool > & visited ) {
if ( visited.size() == temp.size () ){
    ans.push_back ( temp);
    return
}
for( i=0; i< visited.size(), i++){
    if( visited [i] == 0){
        visited [i] = 1;
        temp.push_back( arr [i] );
        Permut(arr , ans , temp , visited);
        visited [i] = 0;
        temp.pop_back();
    }
}
```

T.C = O(n * n * n)

S.C = O(n)

n + n + n



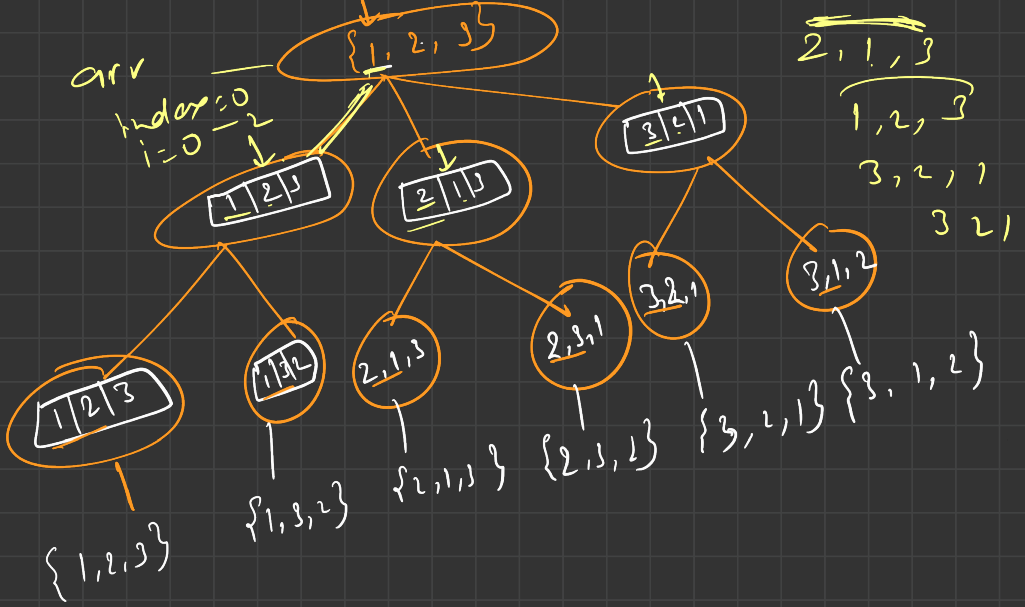
5 — 5

5×4 — 20

5×4×3 — 60

5×4×3×2 — 120

5×4×3×2×1 = 120



```
void permut( vector<int> &arr, vector<vector<int>> &ans,
             int index){
    if (index == arr.size()) {
        ans.push_back(arr)
        return;
    }
    for (i=index, i<arr.size(); i++){
        swap(arr[i], arr[index]);
        permut(arr, ans, index+1);
        swap(arr[i], arr[index]);
    }
}
```