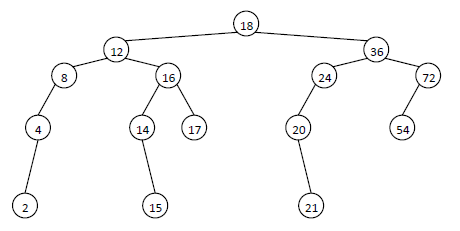
**Binary Search Trees**

This project is a programming assignment in C which aims at developing an algorithm that

builds a binary search tree (BST) from a given sequence and attempts to find other sequences that yield the same BST topology.



**Algorithm For Finding All Possible BST Topologies**

First of all different all possible sequences requires permutation algorithm. I used Heap’s Algorithm for generating permutations. Because it is the best way of generate those numbers. But in given sequence there are 15 numbers which is too high (15! = 1307674368000) I decided to make a little change in algorithm. Since root is same for all sequences, i neglected root which means that execution time decreases x15. But it still runs heavy. Roots childs are 12 and 36, they could neglected as root either then we can make 2! permutations between them. But it is valid for the case that roots both childs are not null. When one single permutation is created by algorithm, a temporary tree is constructed by that permutation sequence. If that tree has same topology with main tree, then count is increased by one. And then temporary tree is removed from heap temporarily. There are total of **3153150** different sequences.