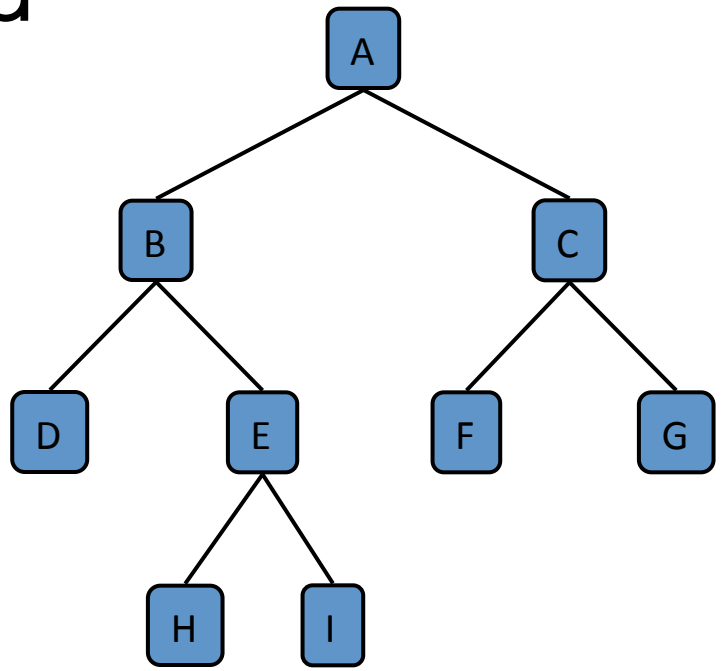
Lecture 11
Binary Trees

Binary Tree Definition

- Each internal node has at most two children
- The children of a node are ordered as left child and right child



Binary Tree Implementation

Node

```
struct Node{
Elem elt;
Node* par;
Node* left;
Node* right;
}
```

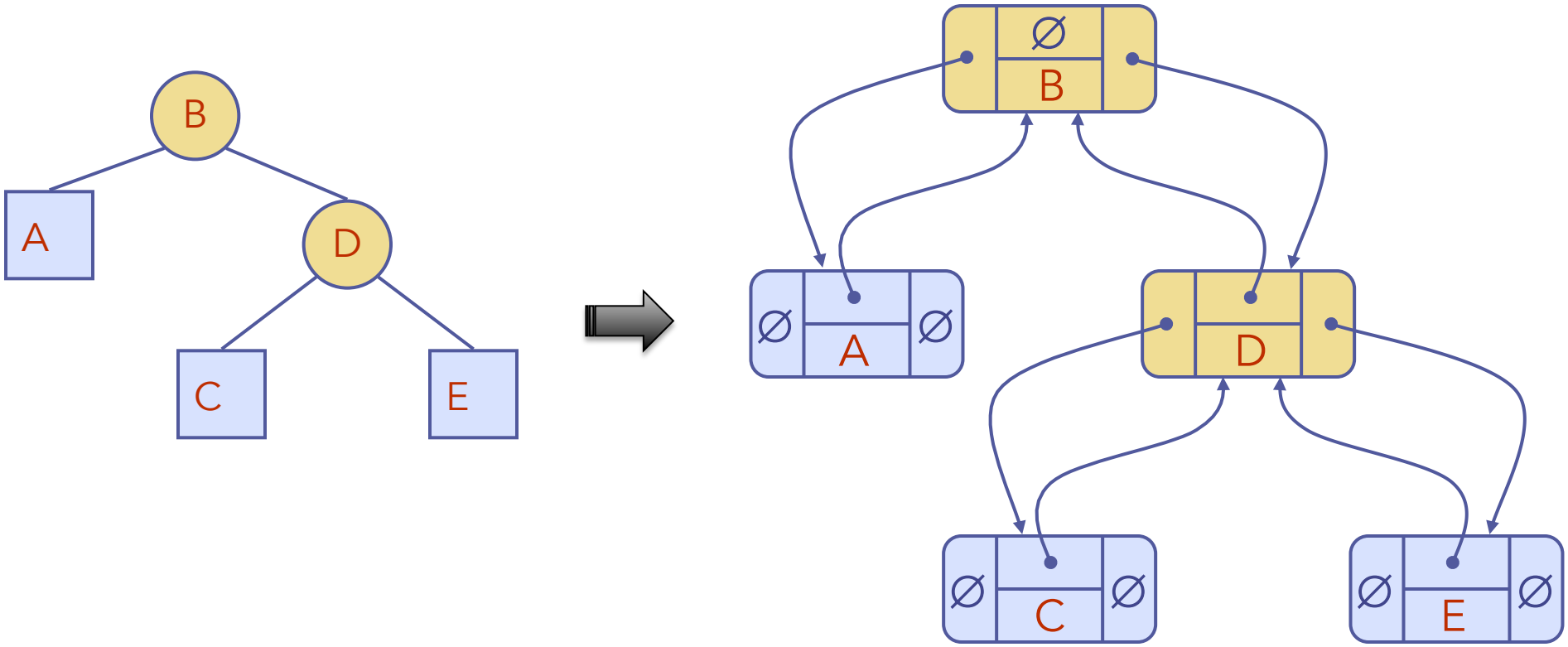
Position for the Node

```
template <typename E>
class Position<E> {
    private:
        Node* v;
    public:
        E& operator*();
        Position left();
        Position right();
        Position parent();
        bool isRoot();
        bool isExternal();
friend class Tree;
};
```

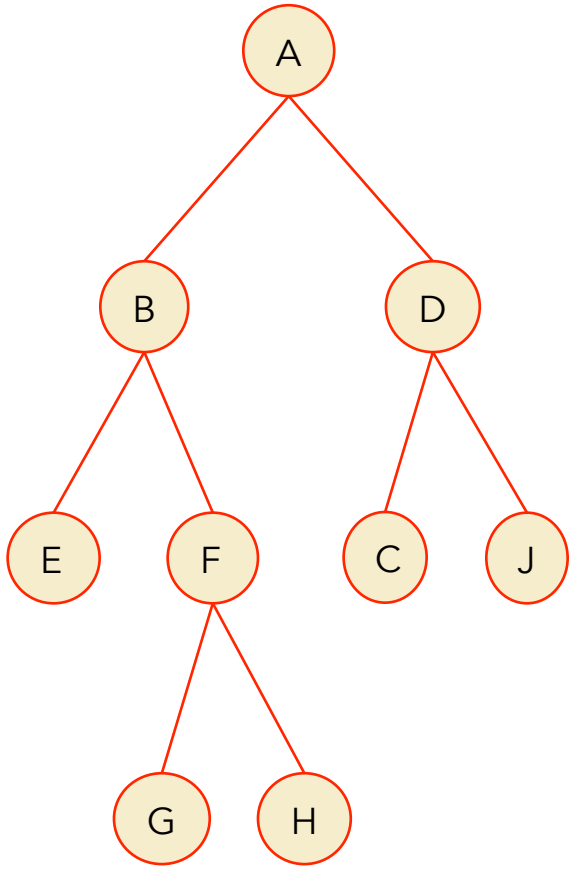
Binary Tree

```
template <typename E>
class Tree<E> {
    private:
        Node* _root;
    public:
        int size();
        bool empty();
        Position root();
        PositionList positions();
};
```

Linked Structure for Binary Trees

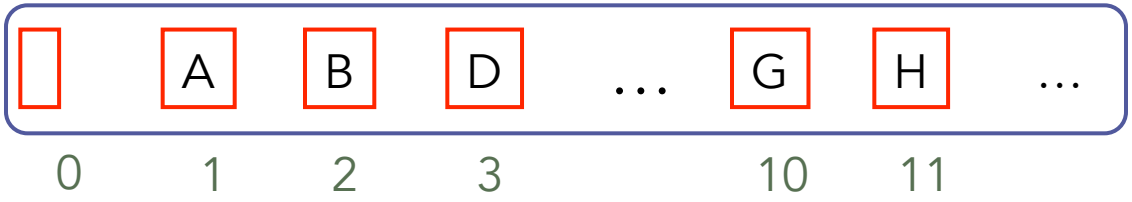


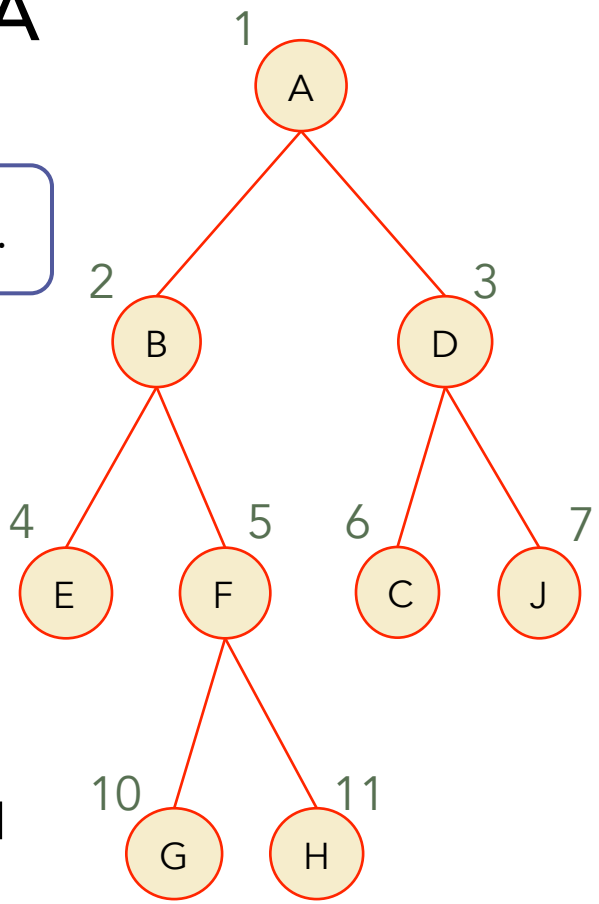
Can we implement a Binary Tree with a Vector?



Array-Based Representation of Binary Trees

- Nodes are stored in an array A





- Node v is stored at A[rank(v)]
  - rank(root) = 1
  - if node is the left child of parent(node),
    rank(node) = 2 · rank(parent(node))
  - if node is the right child of parent(node),
    rank(node) = 2 · rank(parent(node)) + 1

Properties of Binary Trees

What is the maximum tree height for n nodes?

$$h \leq n-1$$

What is the minimum tree height for n nodes?

$$h \geq \log (n+1) - 1$$

$$h+1 \leq n \leq 2^{h+1}-1$$

h = height

n = number of nodes

Minimum number of external nodes in a tree of height h?

1

Maximum number of external nodes in a tree of height h?

2^h

$$1 \leq n_E \leq 2^h$$

n_E=No. of external nodes

Minimum number of internal nodes in a tree of height h?

h

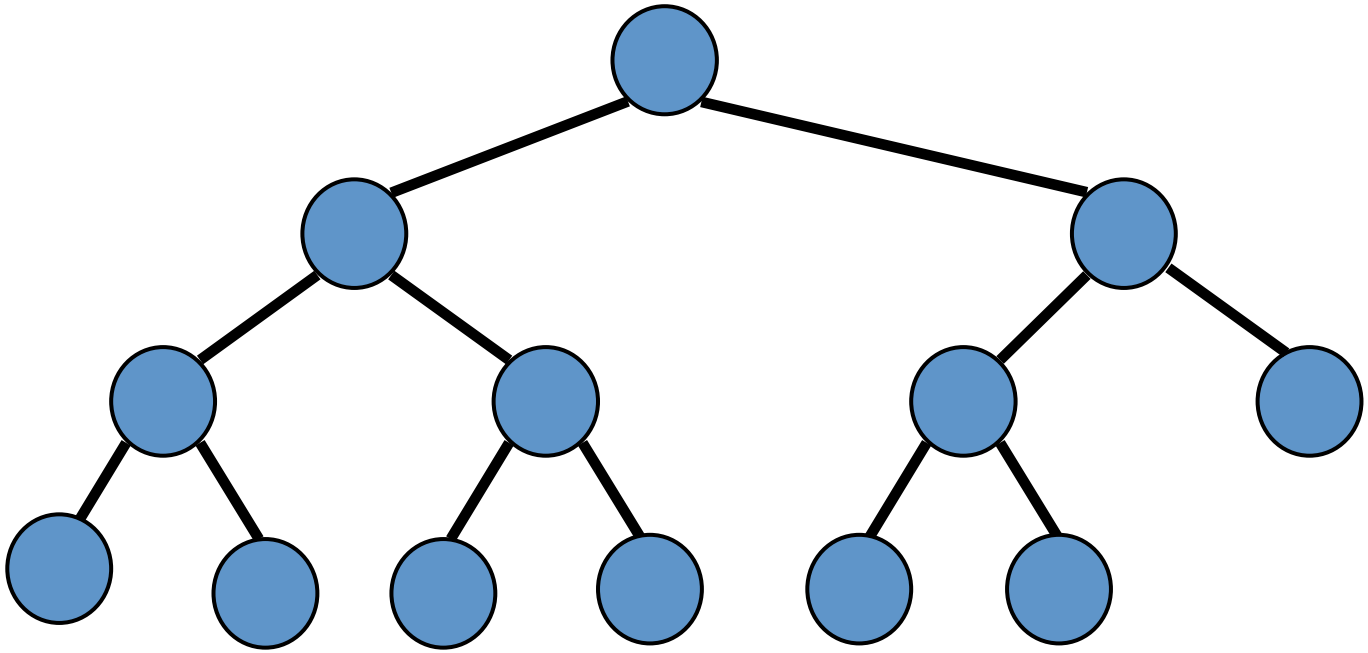
Maximum number of internal nodes in a tree of height h?

2^h-1

$$1 \leq n_l \leq 2^h - 1$$

n_l = No. of internal nodes

Proper Binary Tree :no child or 2 child



Height

$$\log (n+1) - 1 \leq h \leq (n-1)/2$$

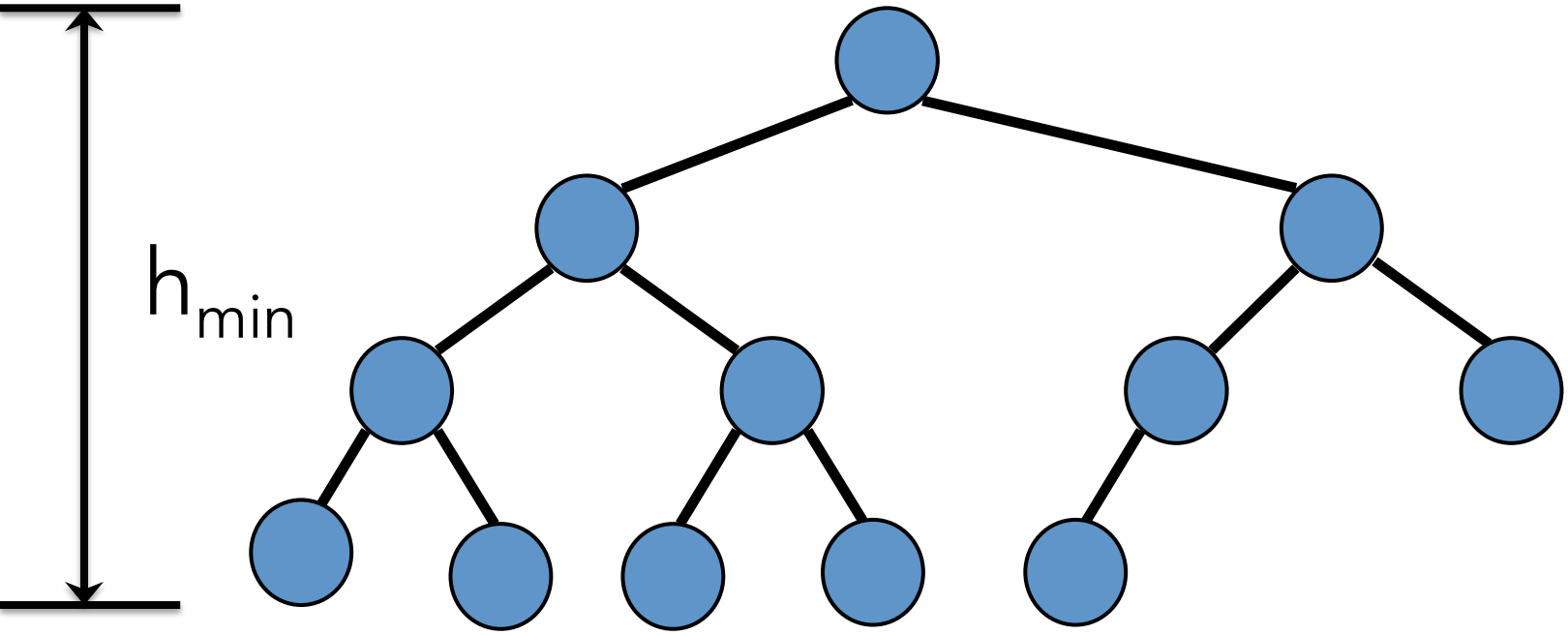
External Nodes

$$h+1 \leq n_E \leq 2^h$$

Internal Nodes

$$h \leq n_I \leq 2^h-1$$

Complete Binary Tree?

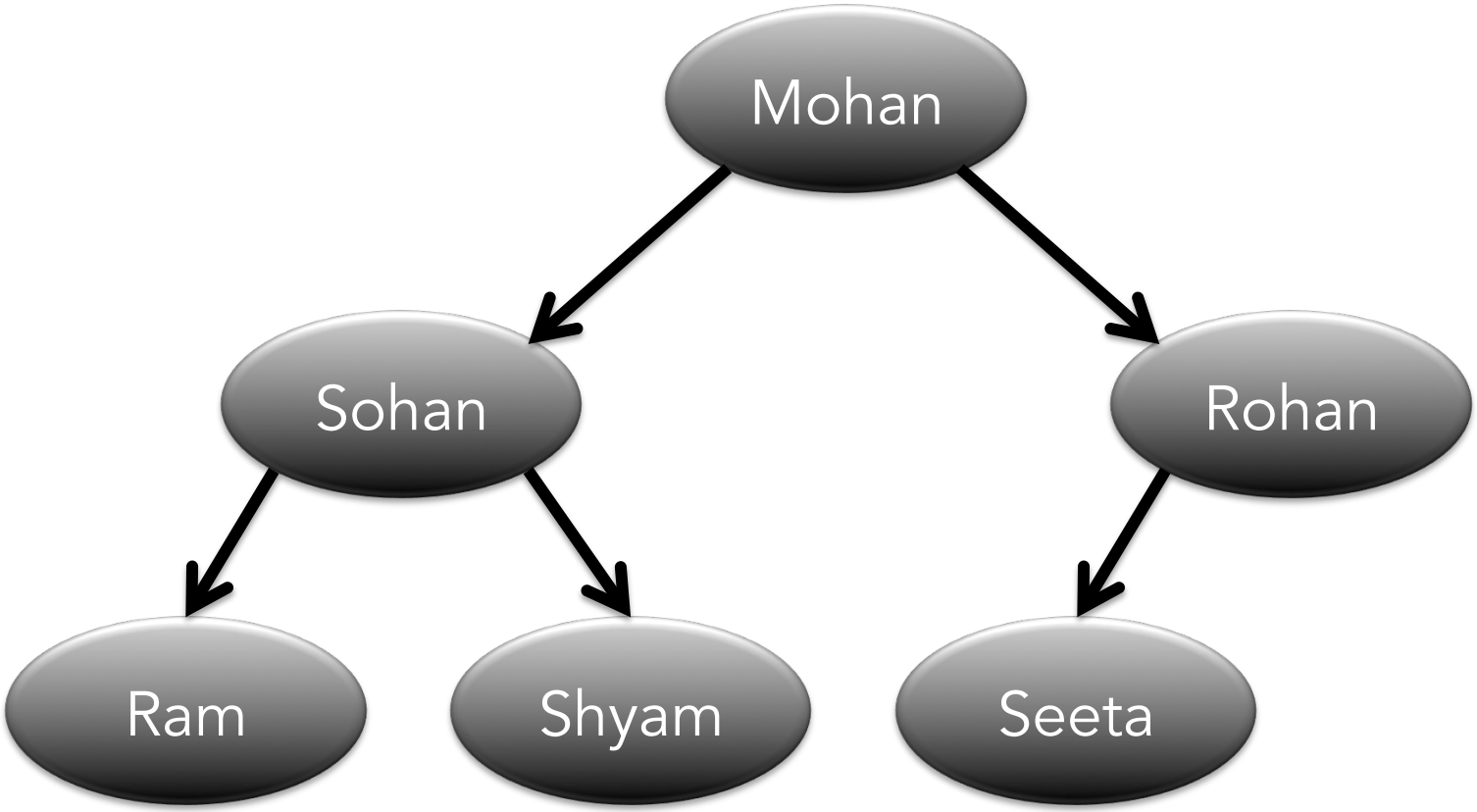


How many Binary Trees are possible given n nodes?

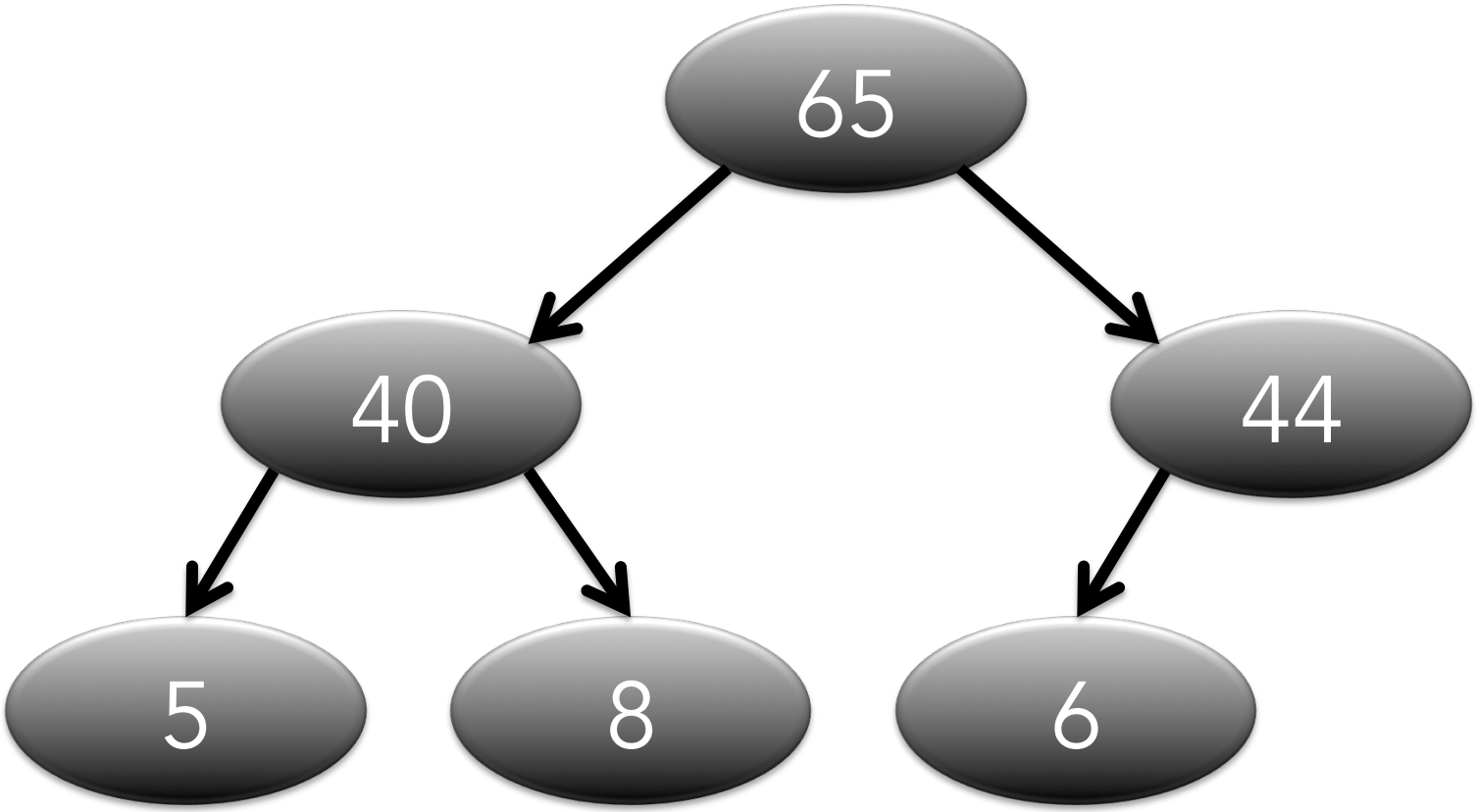
Ordered Tree

- children have certain order as being first, second, etc.
- the leftmost child is called the “first”
- e.g. book

Ordered Family Tree



Numbered Ordered Tree

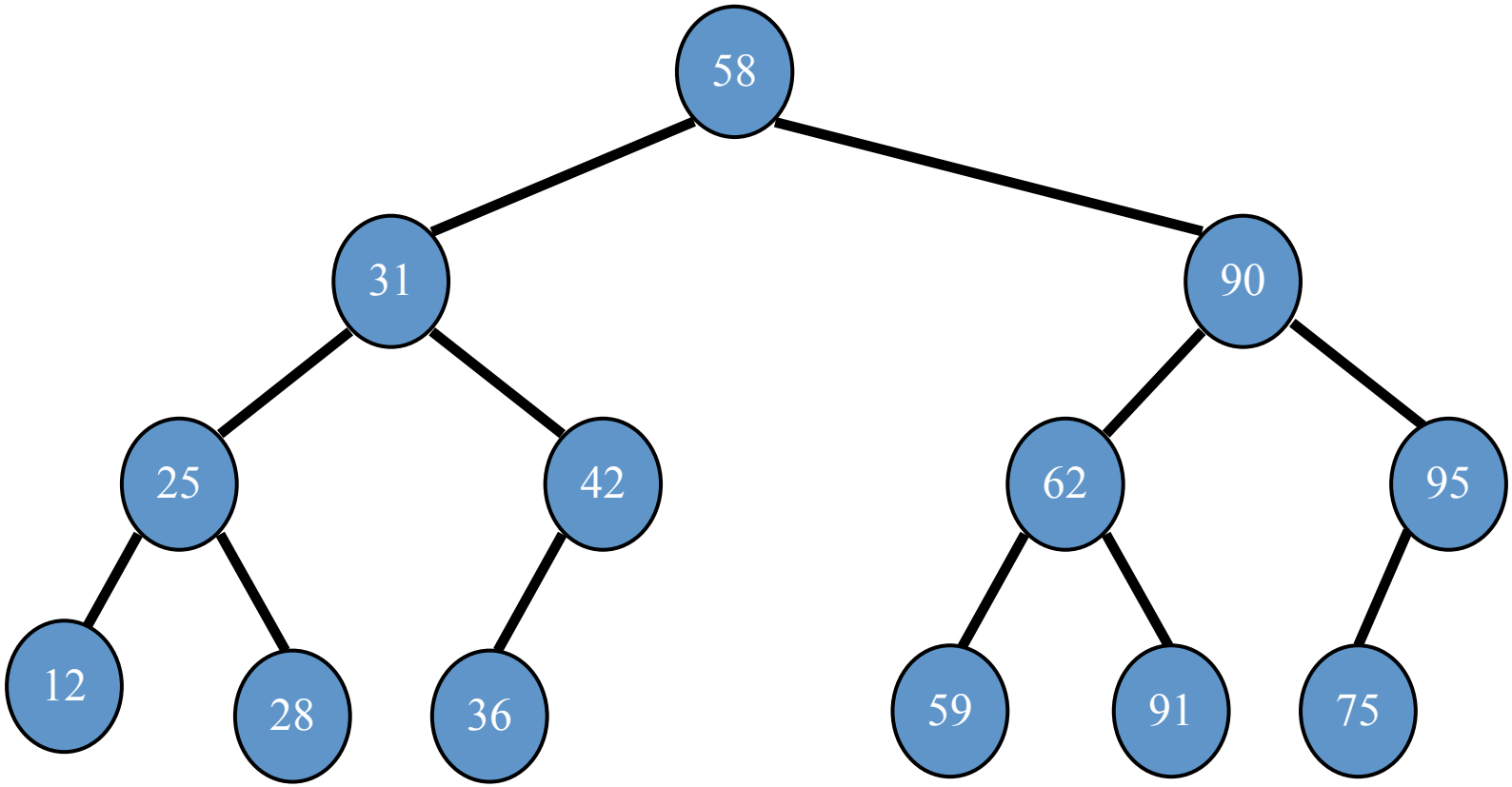


Binary Search Tree

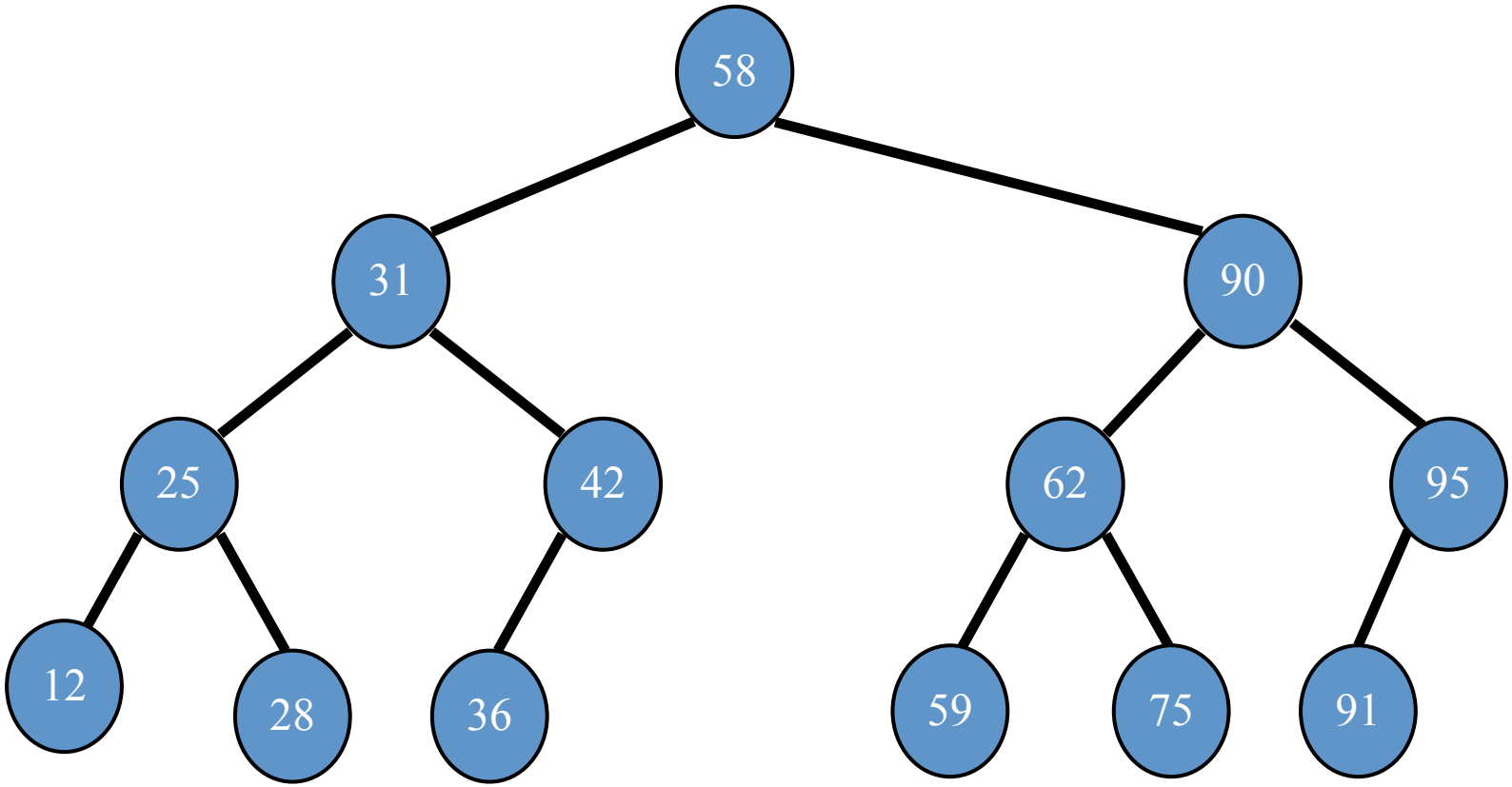
Binary Search Tree

- every node stores a key
- left subtree < node
- right subtree > node

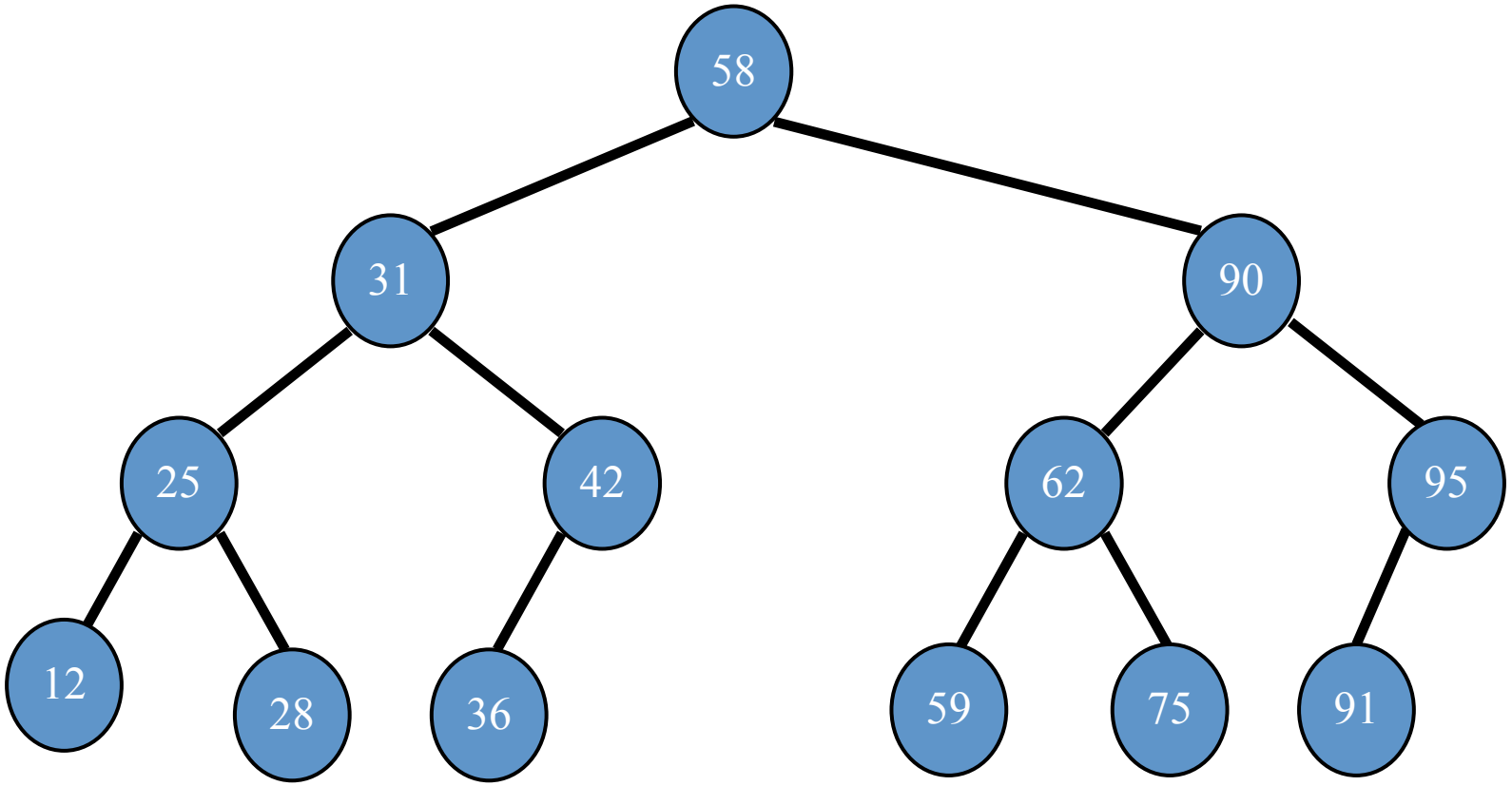
Is this a BST?



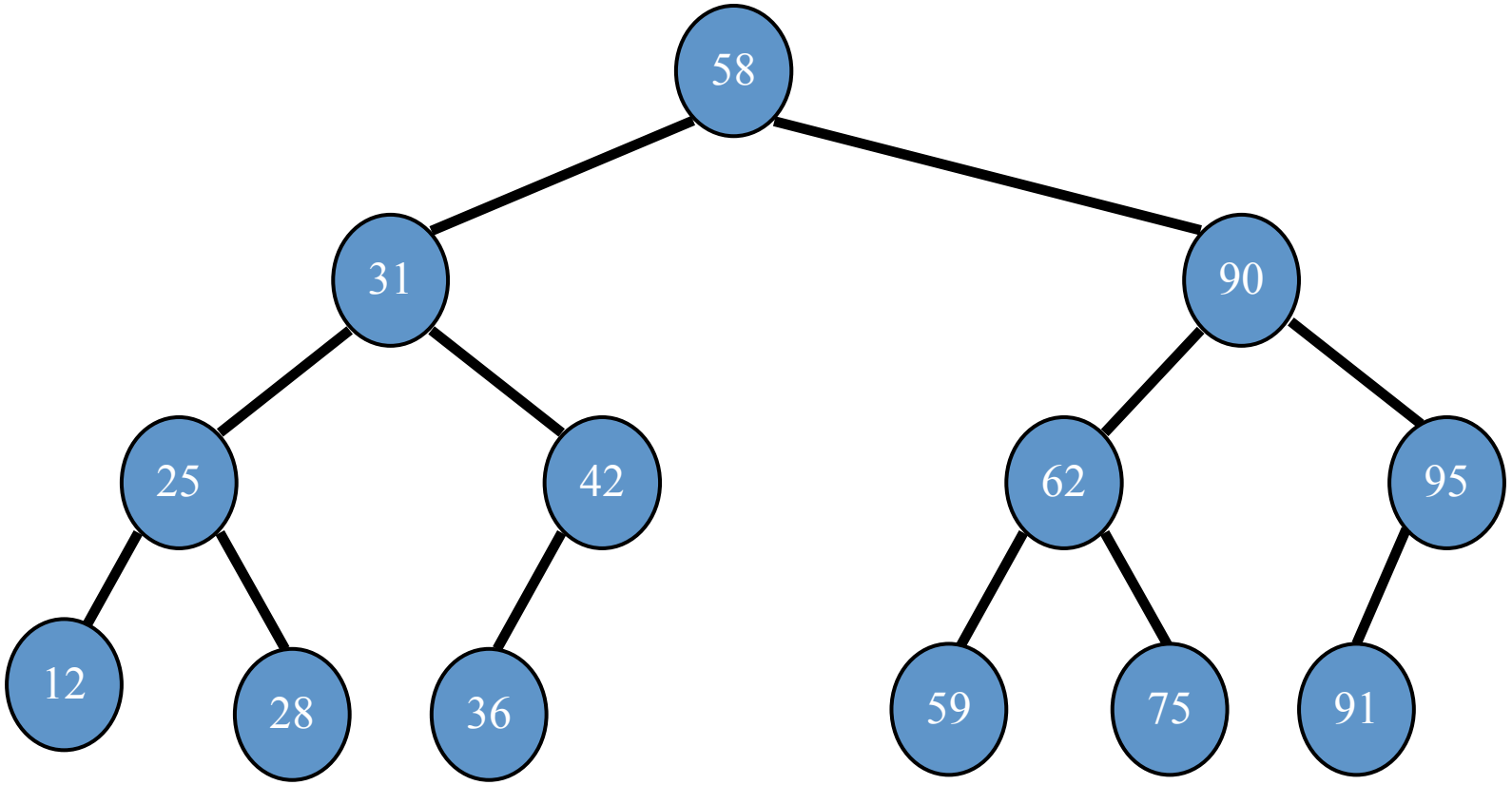
A BST



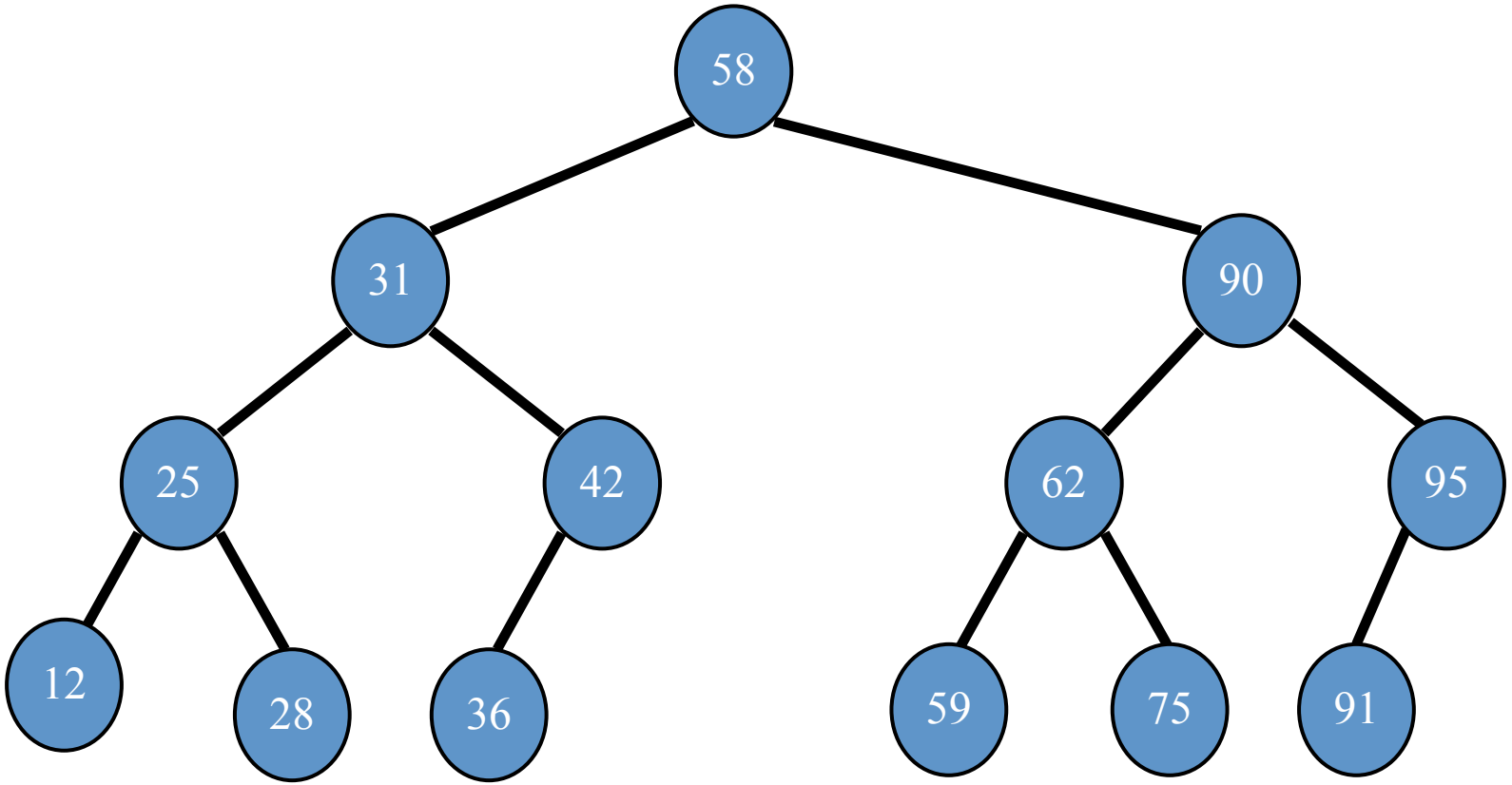
How to search key?
35, 59



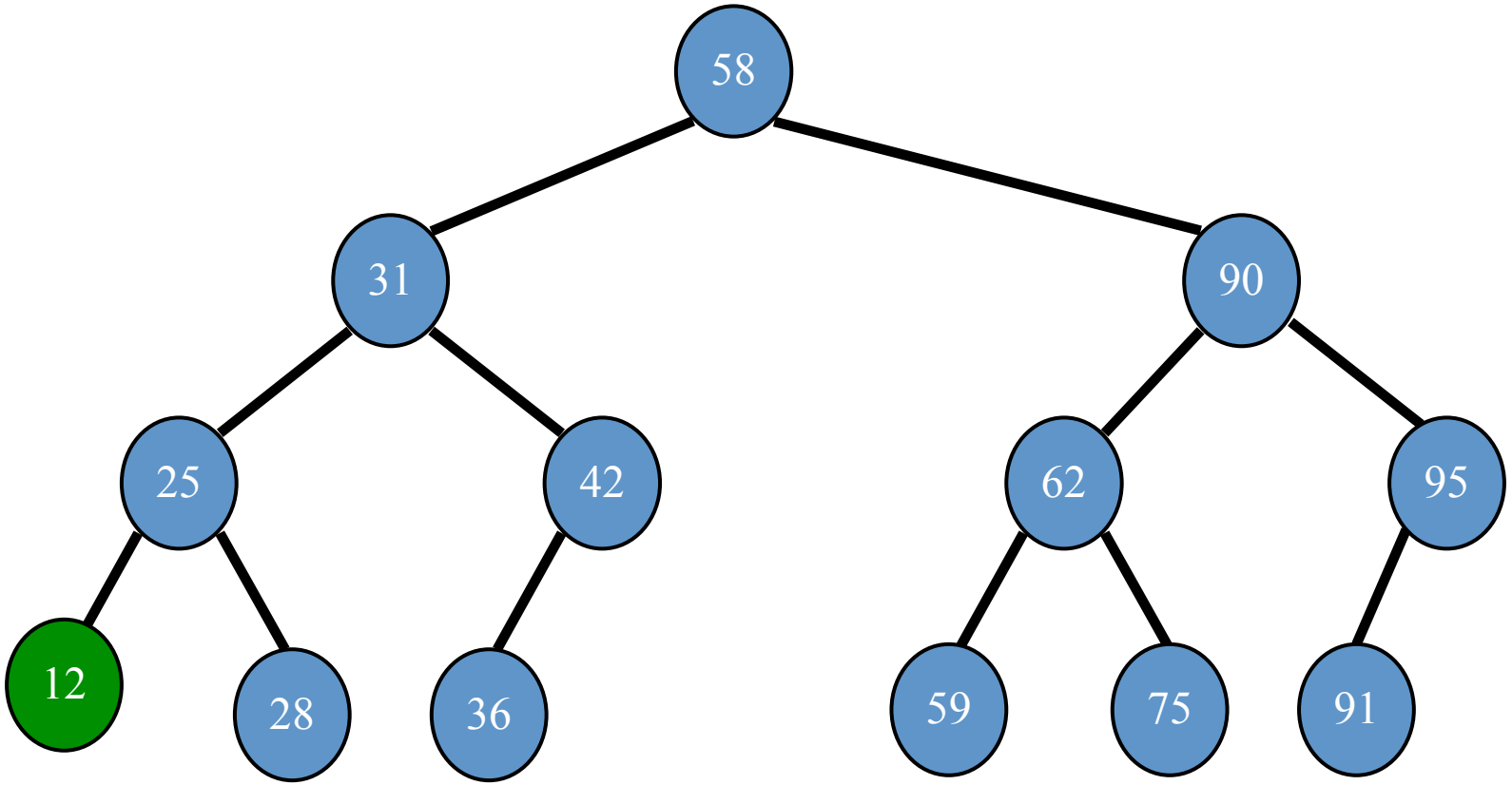
What is the time complexity of search?



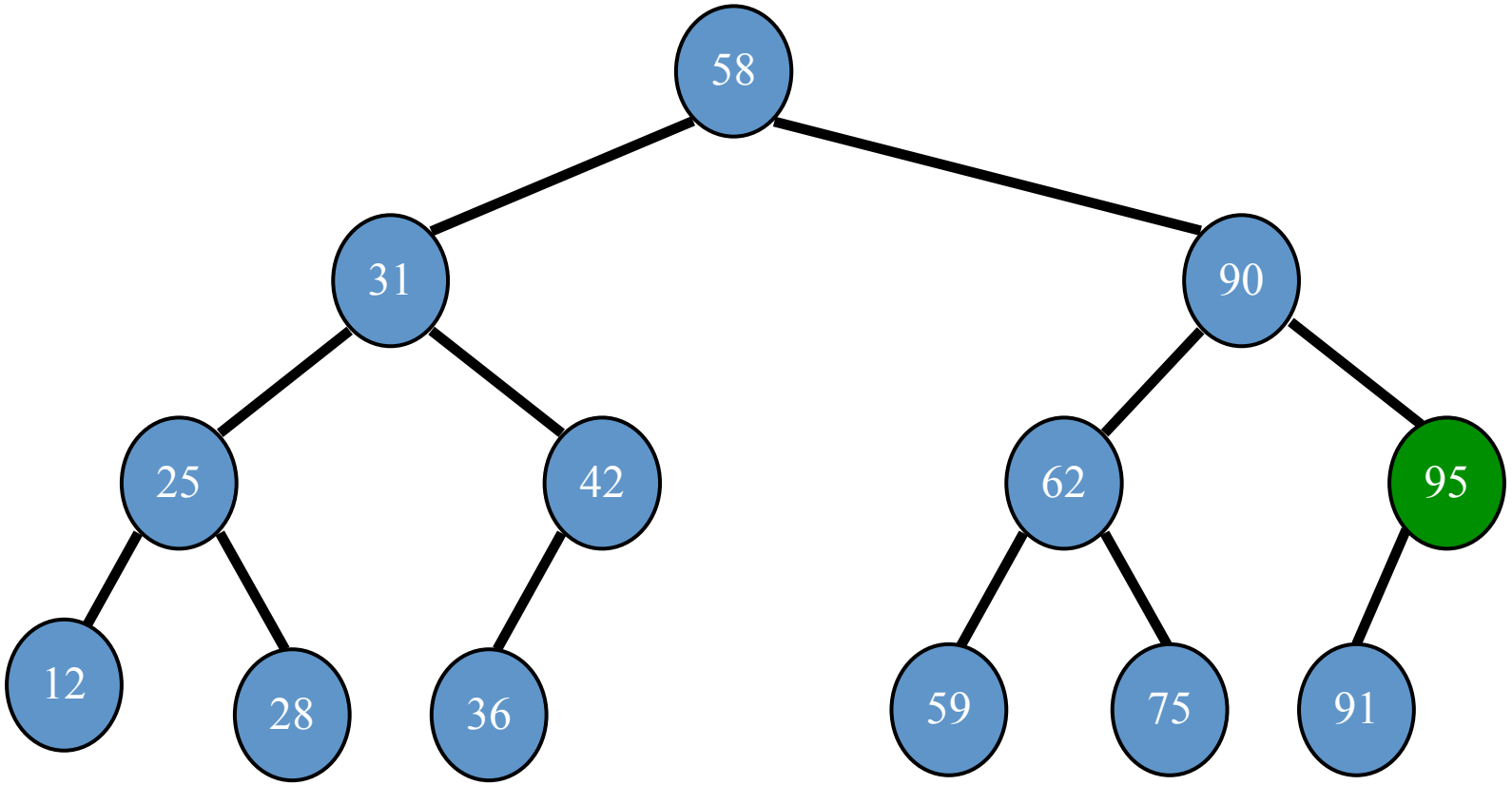
Find node with minimum number?



Find node with maximum number?



How to arrange in increasing order?



How many BSTs are possible given n nodes?