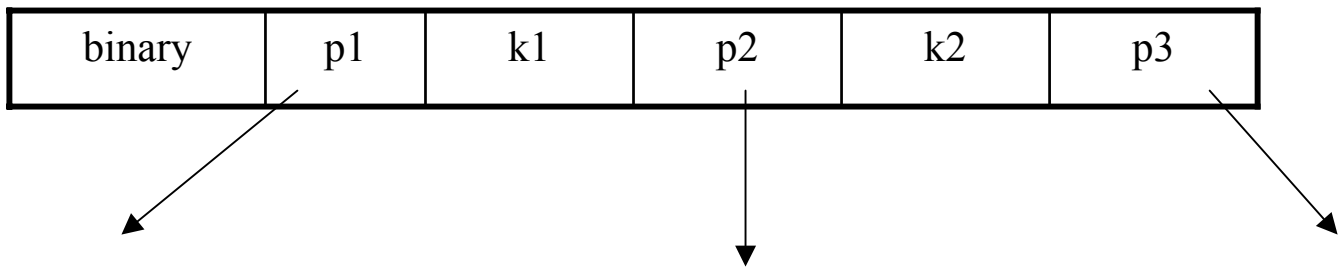
2-3 TREE

2-3 Tree Definition

- All leaf nodes are at the same level.
- All non leaf nodes are either binary or ternary.
- Binary nodes are like BST nodes with single key k.
- Ternary nodes have two keys k1 and k2 and three children pointers p1, p2, p3.
- All keys < k1 reside in sub-tree pointed to by p1. All keys > k2 reside in sub-tree pointed to by p3. Others reside in the sub-tree pointed to by p2. All keys are distinct .

2-3 Tree node

```
typedef struct nt {
        T       k1, k2;
        int     binary;
struct nt       * p1, * p2, * p3;
        }       23treenode;
```



Operations

- If h is the height of a 2-3 tree having n nodes,

then $2^h-1 <= n <= 3^h-1$

Thus, $\log_3(n+1) <= h <= \log_2(n+1)$

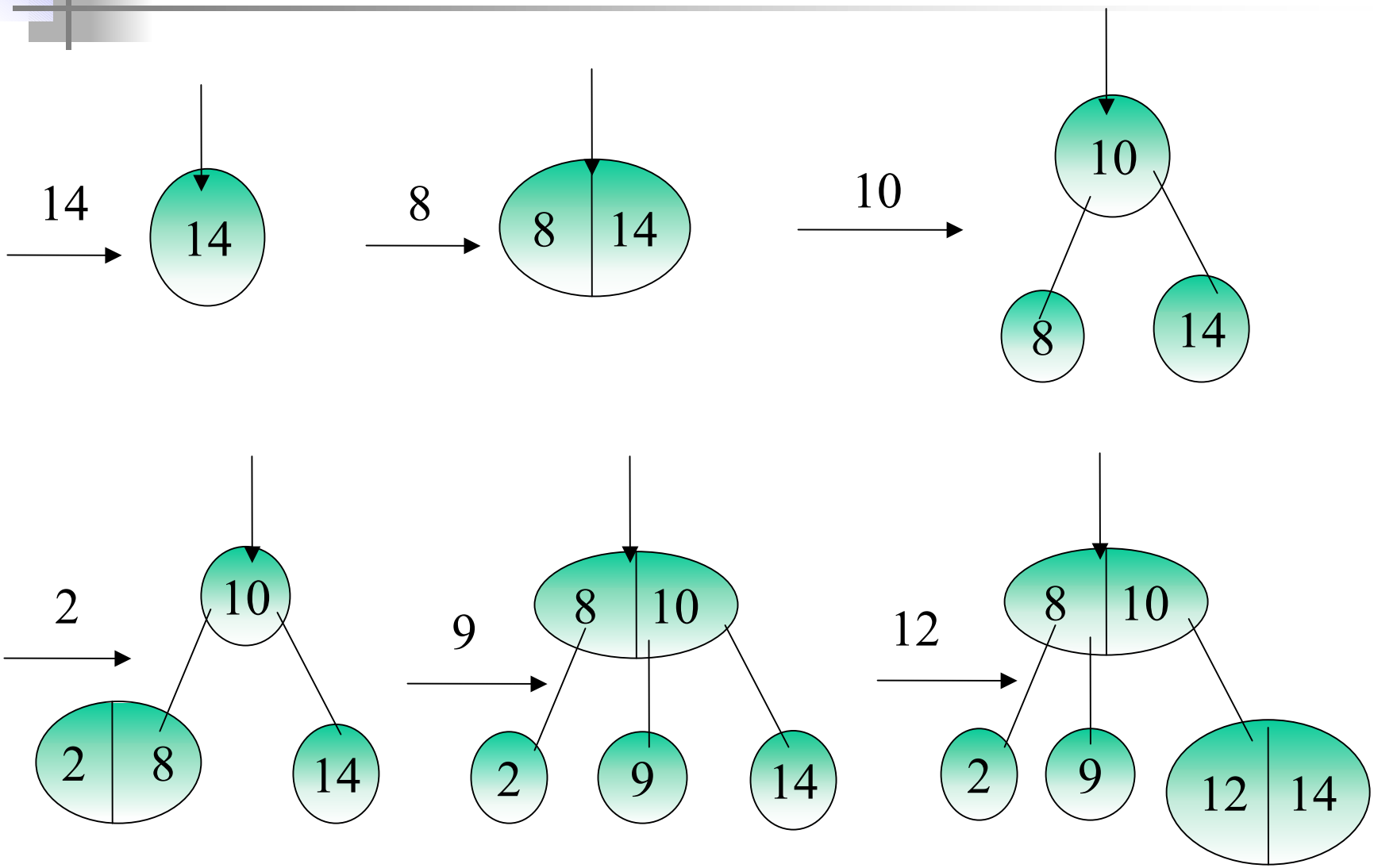
Operations:

Insert node

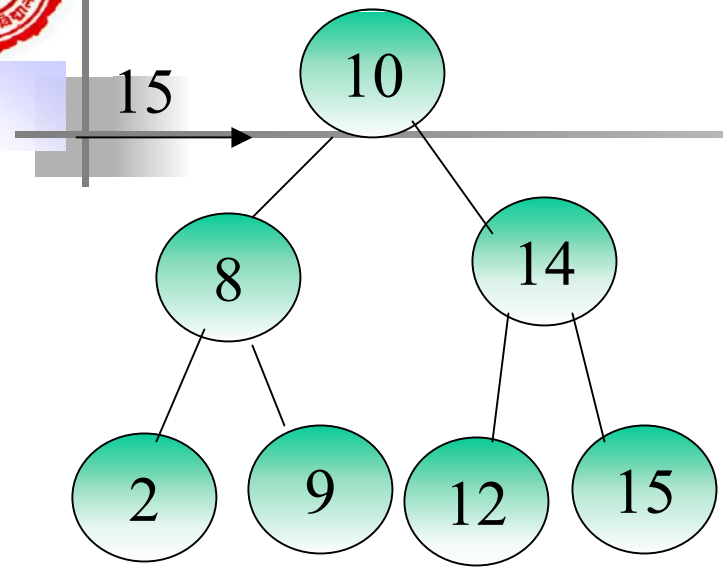
Delete node

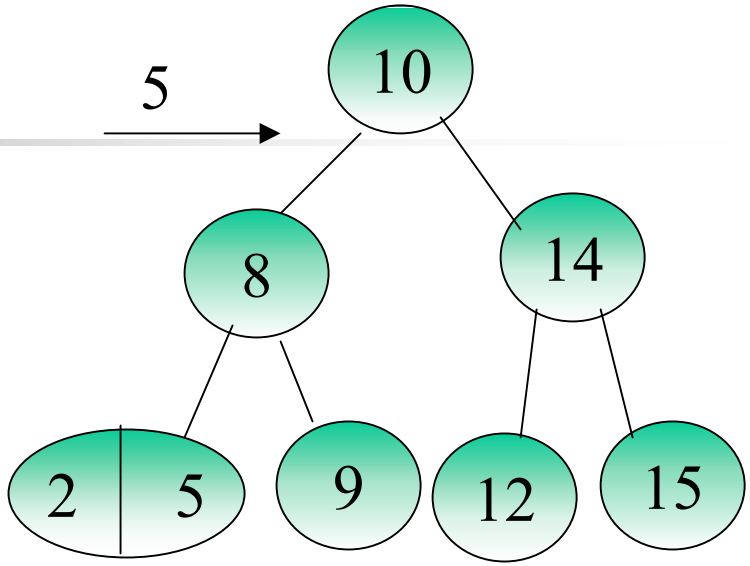
Search node

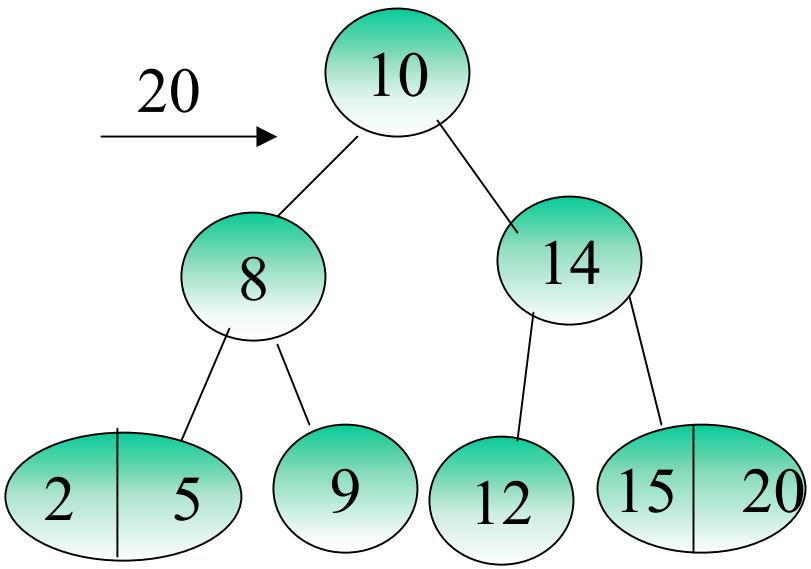
INSERTION EXAMPLES

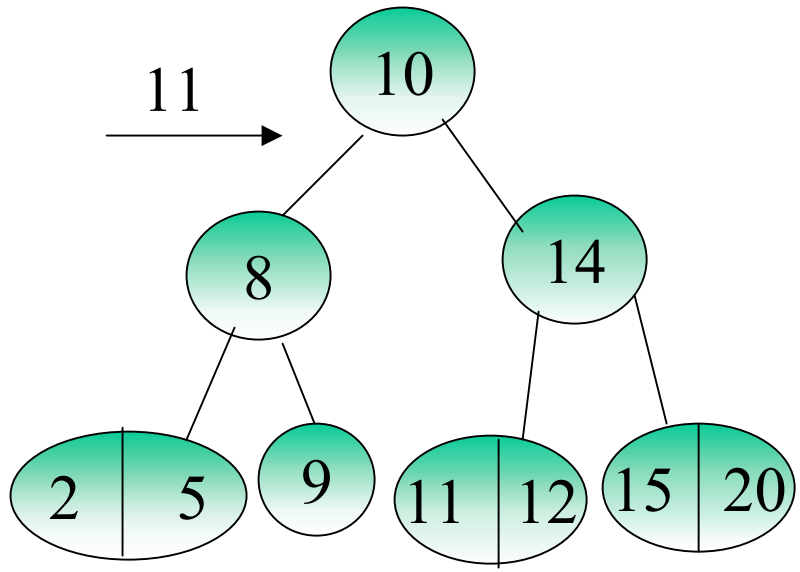


INSERTION EXAMPLES …

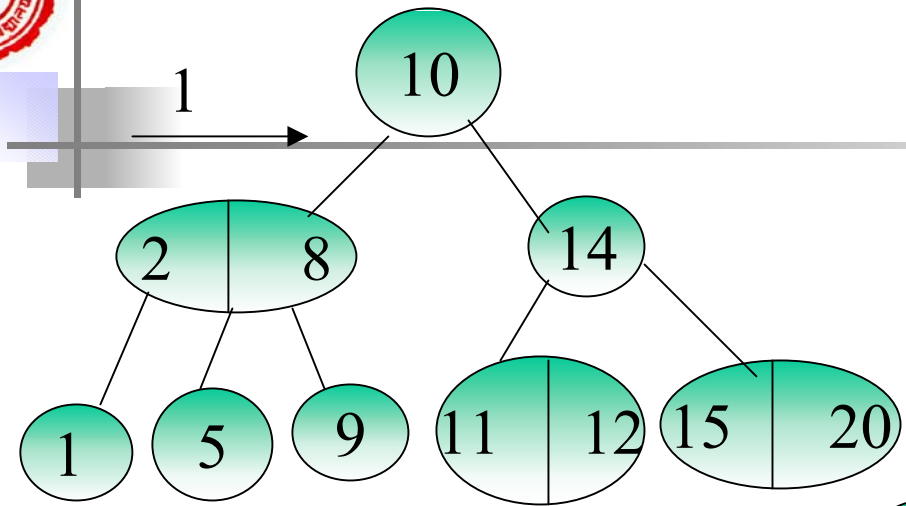


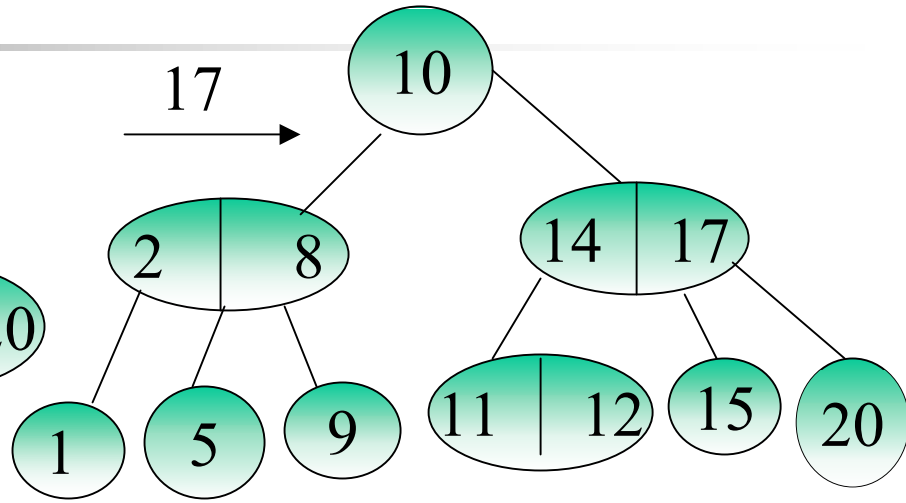


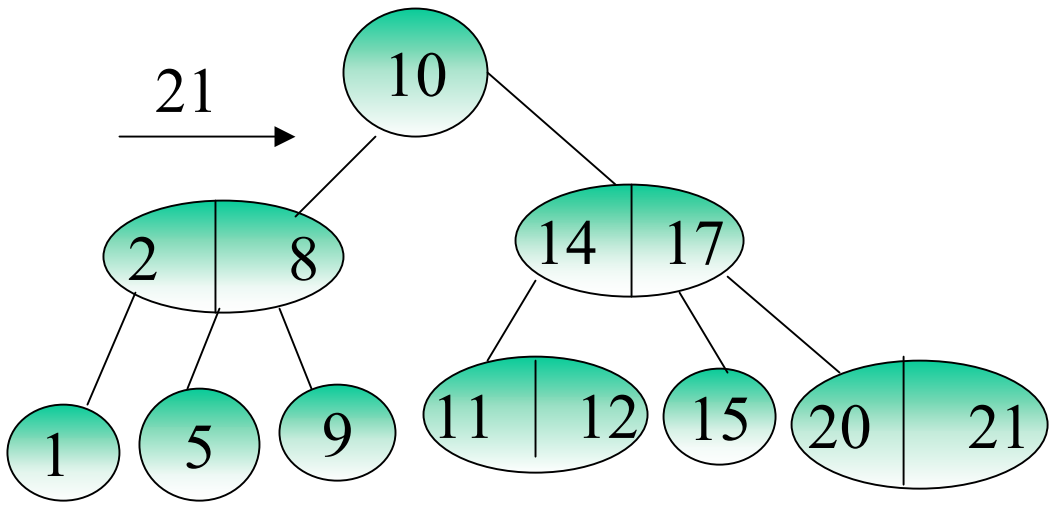




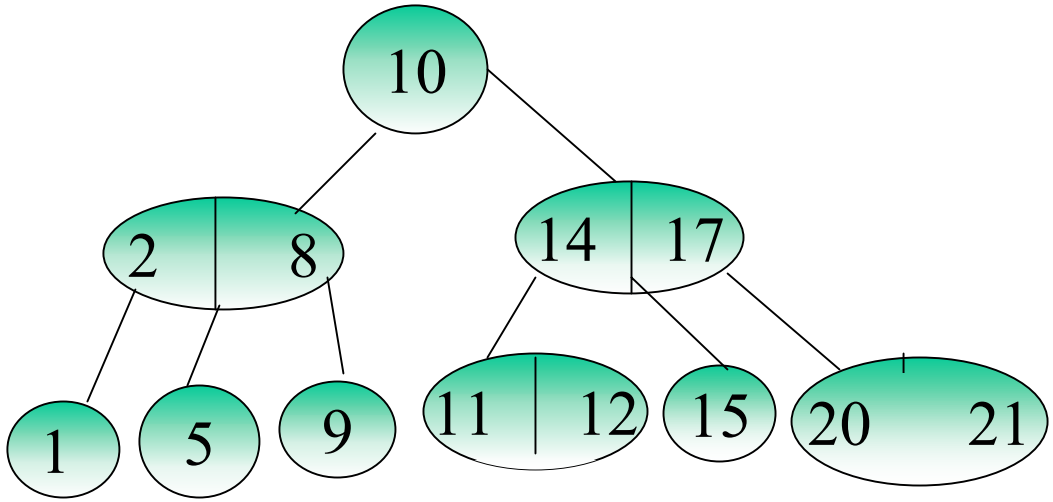
INSERTION EXAMPLES …

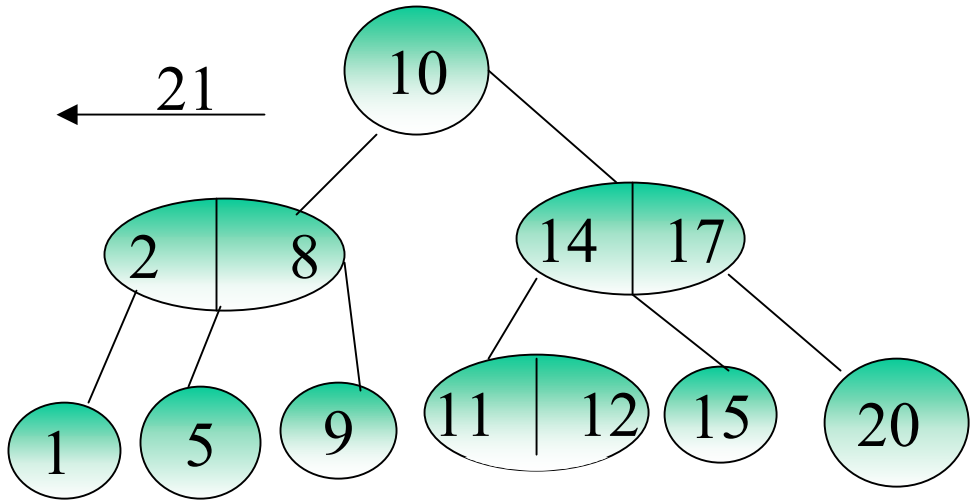




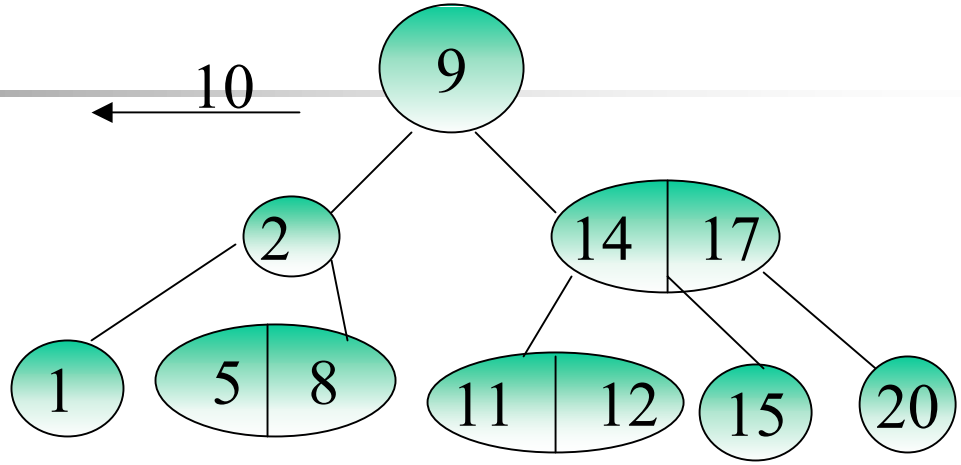


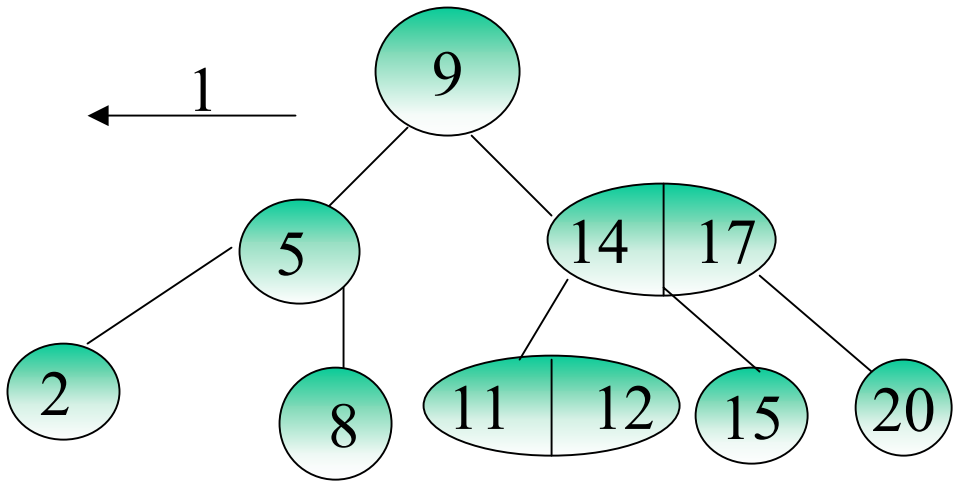
DELETION EXAMPLES



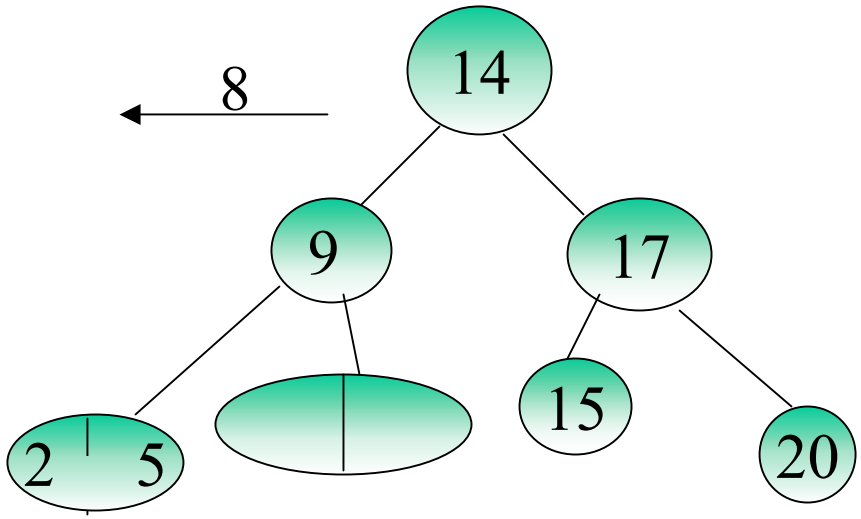


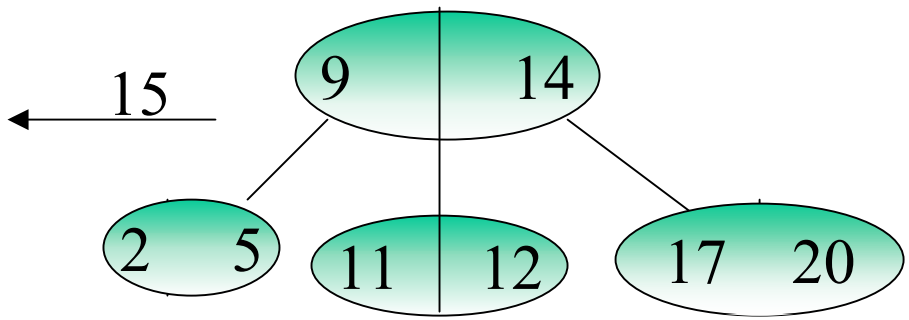
DELETION EXAMPLES ...





DELETION EXAMPLES …





B TREE

B-Tree Definition

A B-Tree of order d is a tree with following properties:

- Each node (except possibly the root node) contains at most d records and at least $\lfloor d/2 \rfloor$ records.
- The root node can have at most d records and as few as one record.
- An internal node containing k records ($1 <= k <= d$) with key values k_1 to k_k, have pointers to k+1 subintervals of keys stored in sub-trees.
- A leaf node has empty sub-tree below it. All leaf nodes are at the same level.

B-TREE Node

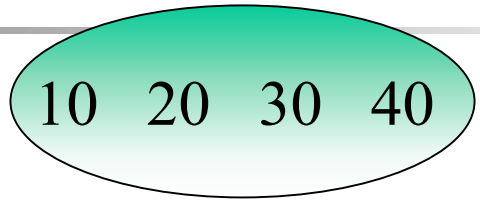
```
typedef struct btnode {
                        int       k;
                        T         datalist[maxkeyno];
    struct btnode       * ptr[maxkeynode + 1];
                        }
```

Height of a B-Tree of order d containing n nodes is given by $\lfloor \log_{\lfloor d/2 \rfloor + 1}(n+1) \rfloor >= h >= \lceil \log_{d+1}(n+1) \rceil$

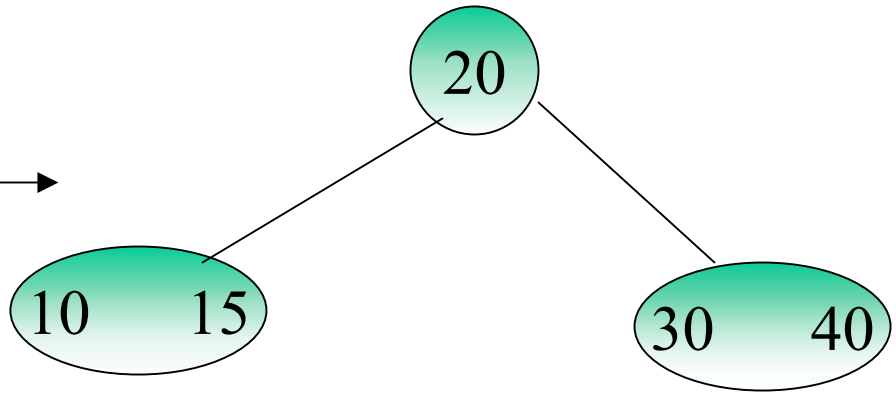
2-3 tree is a B-Tree of order 2

Insertion Examples on a B-Tree of order 4

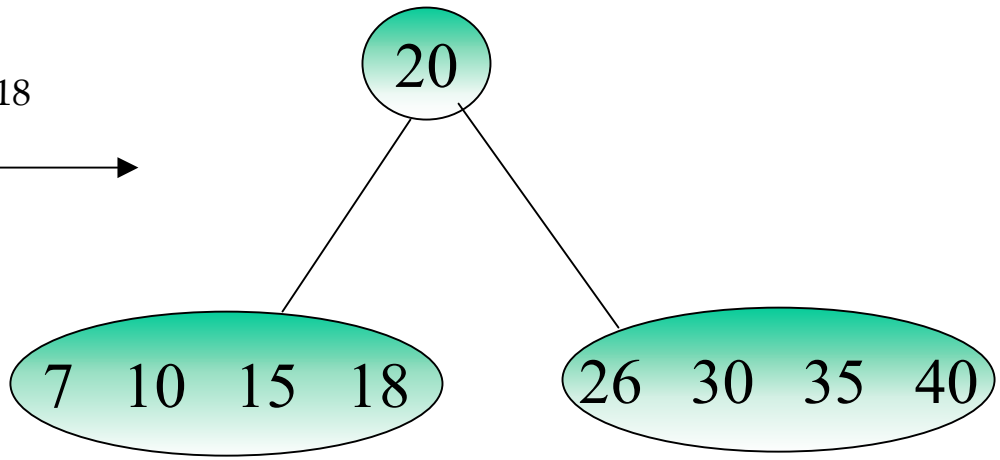
20, 40, 10, 30



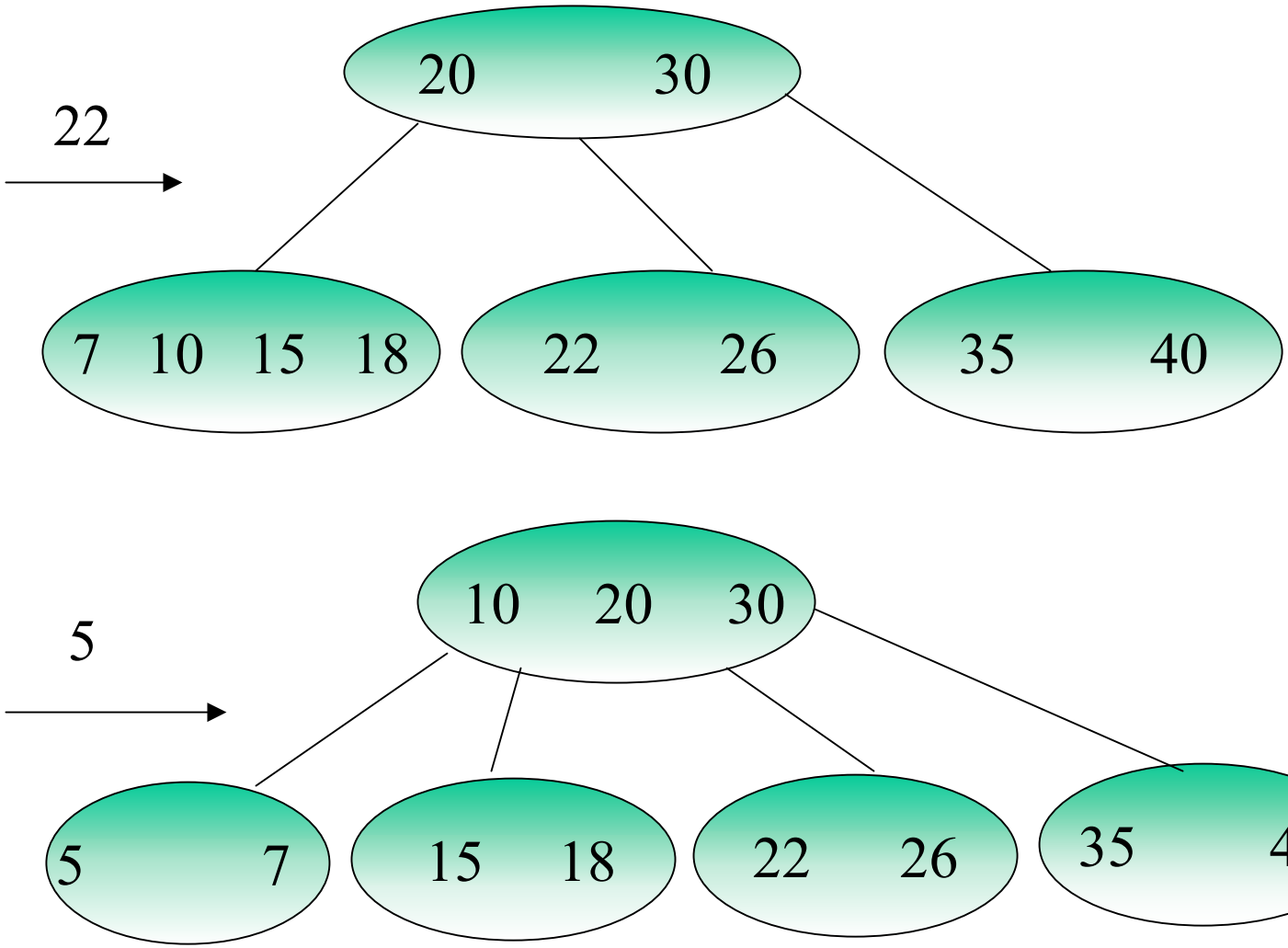
15



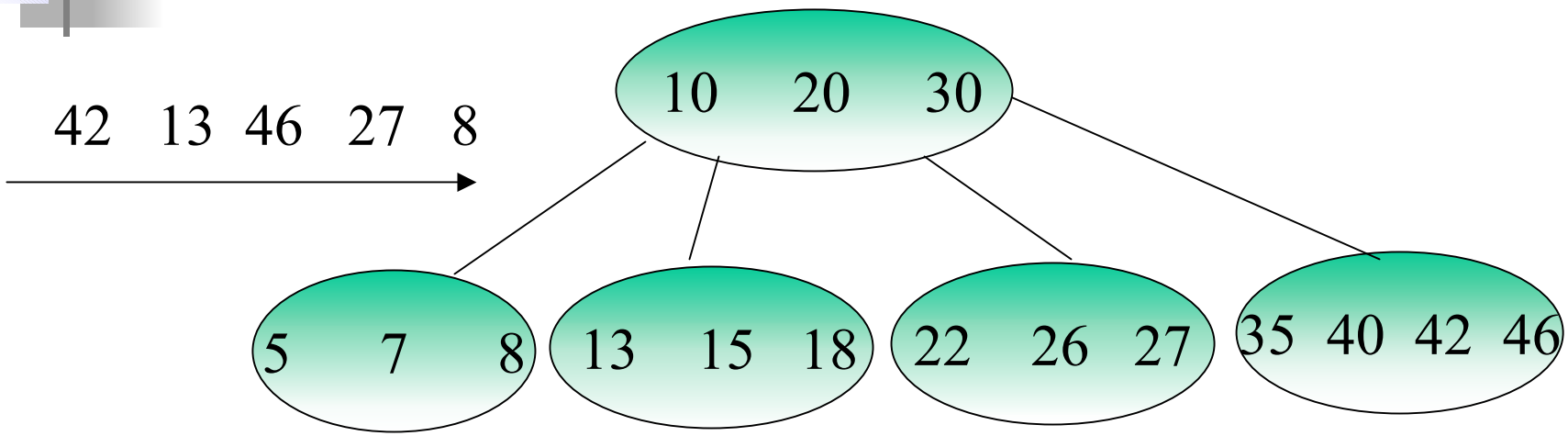
35, 7, 26,18

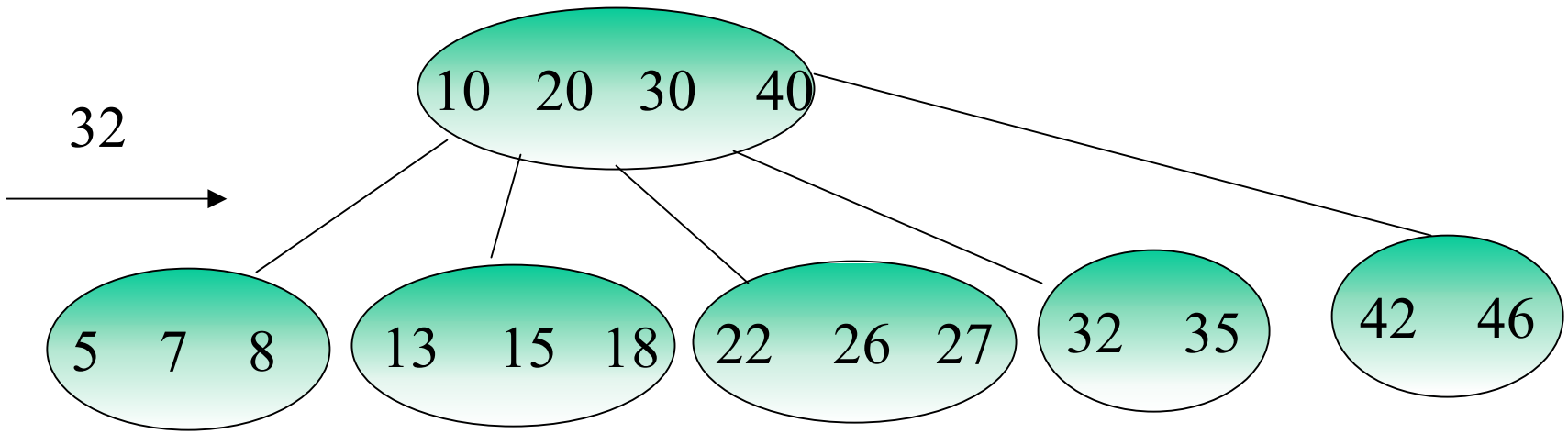


Insertion Examples …

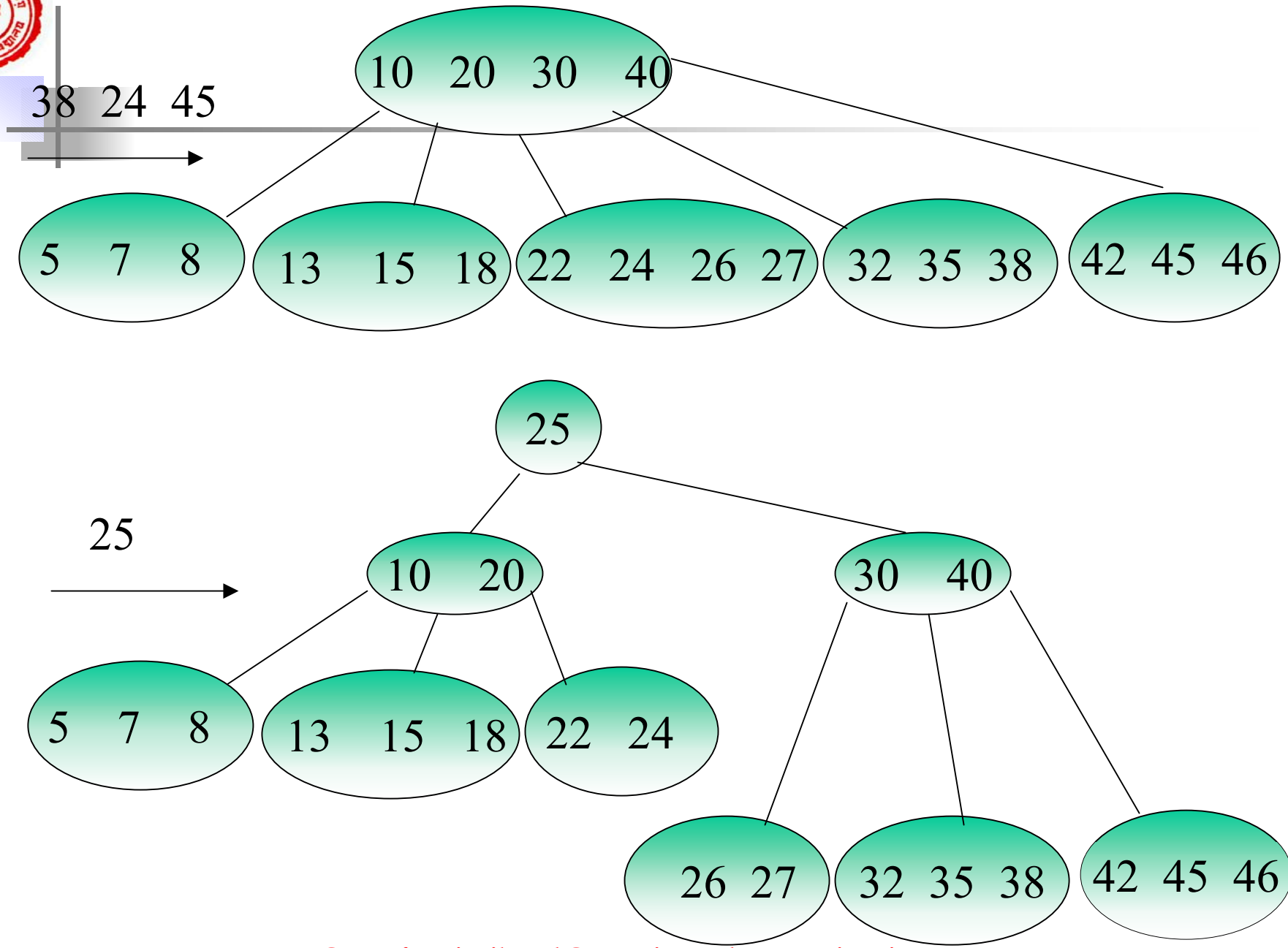


Insertion Examples …

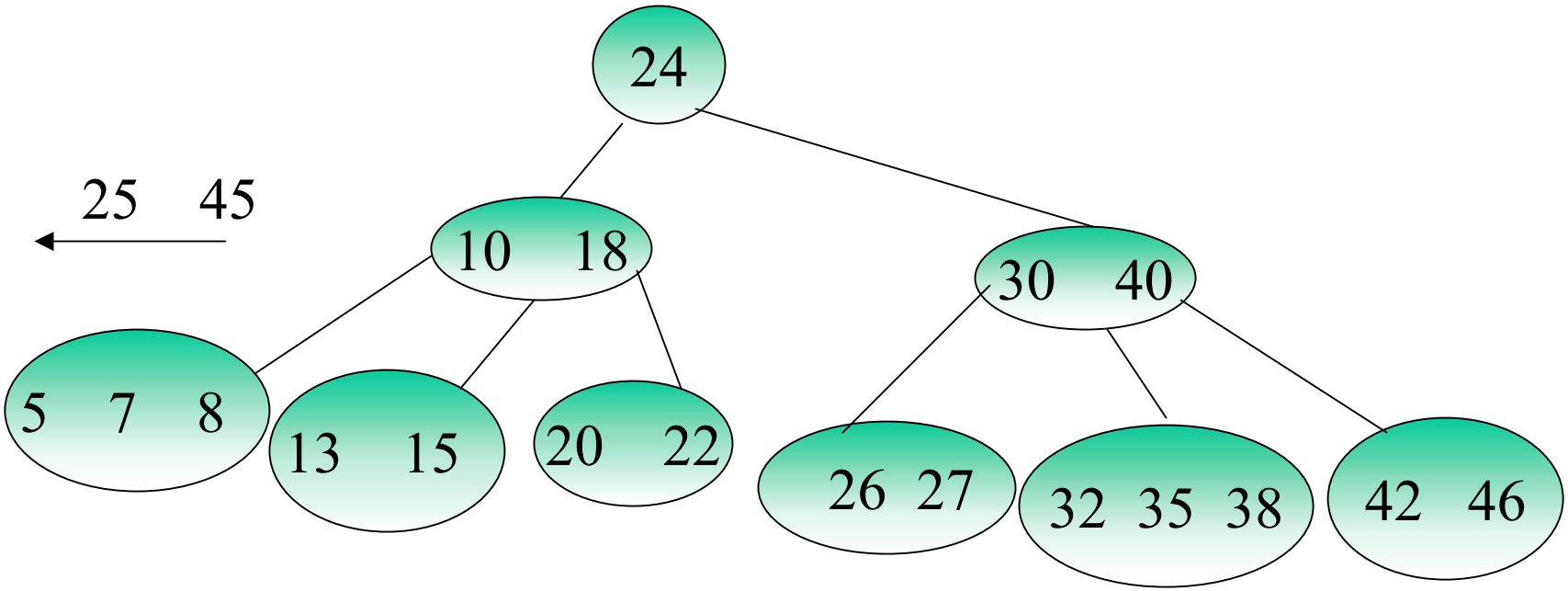




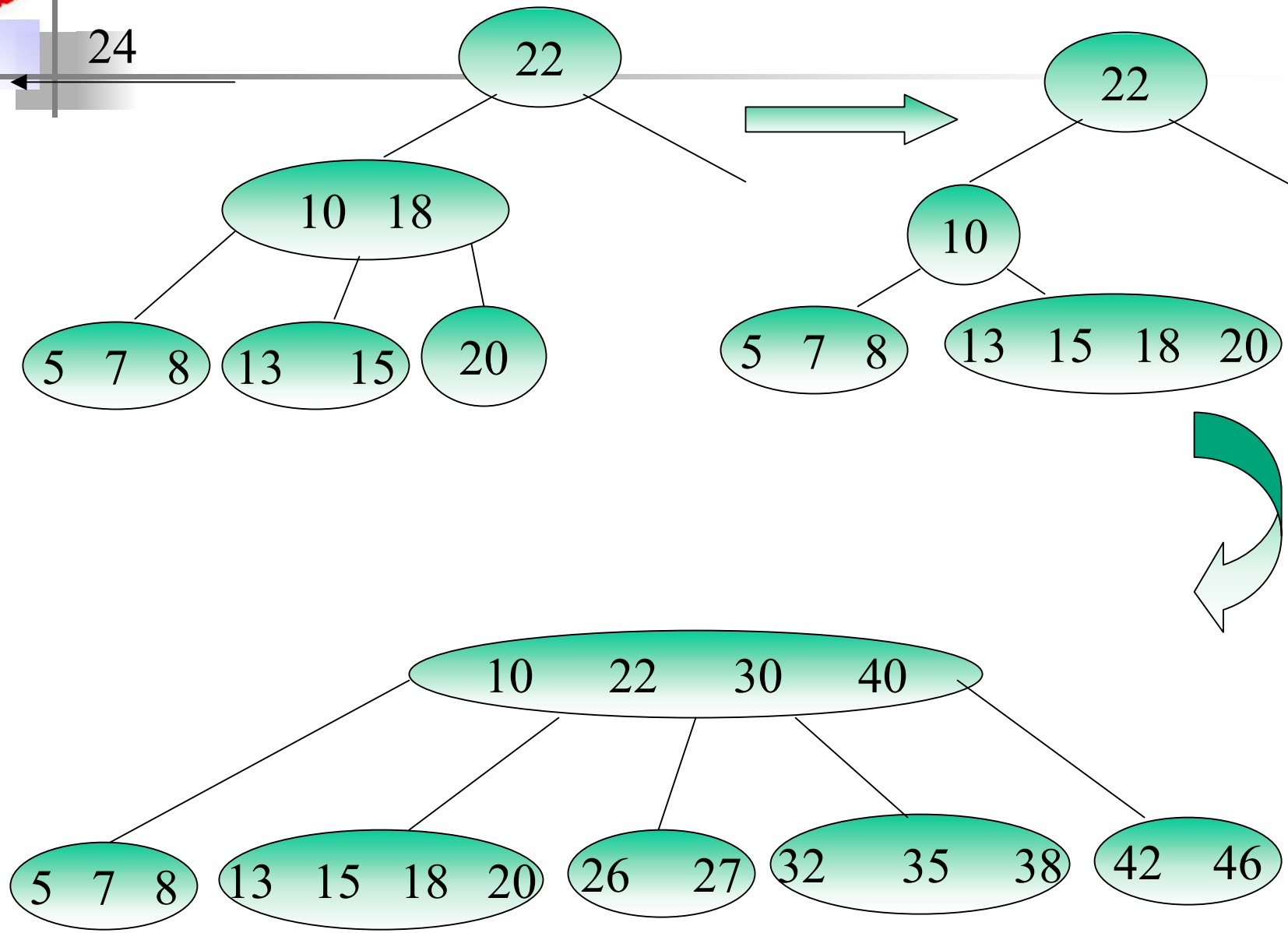
Insertion Examples …



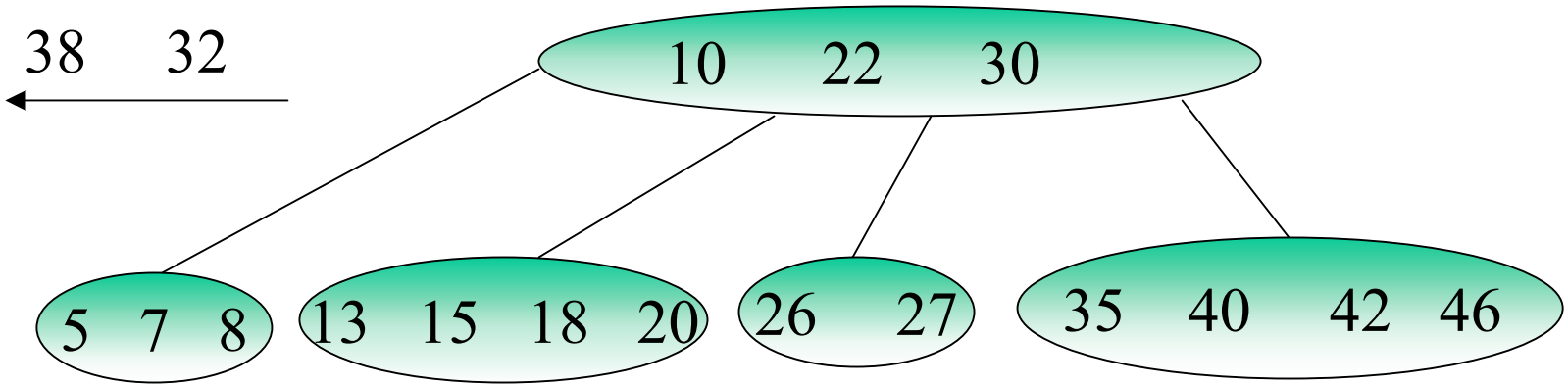
Deletion examples

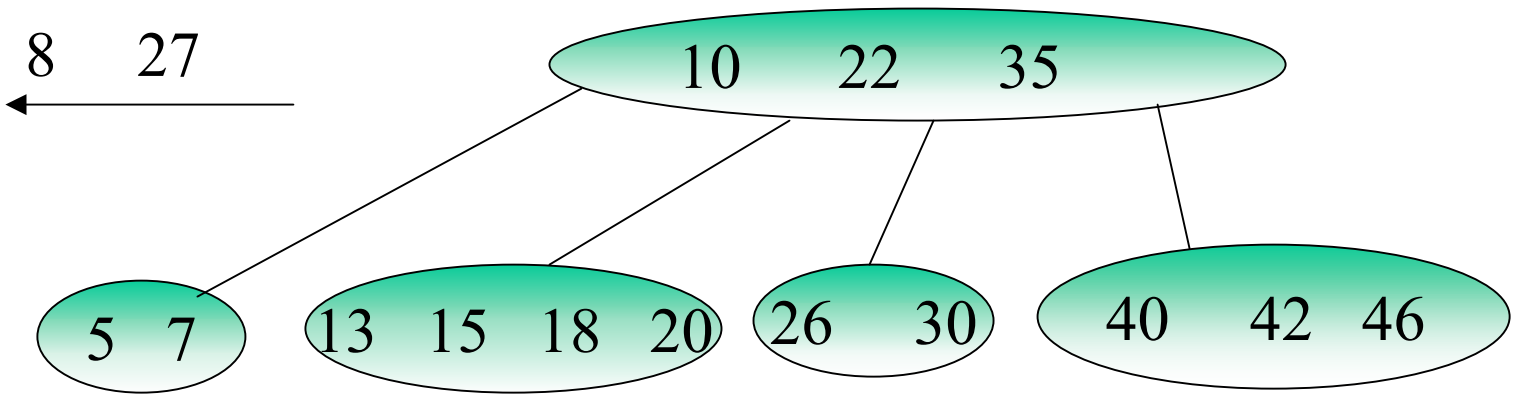


Deletion examples …

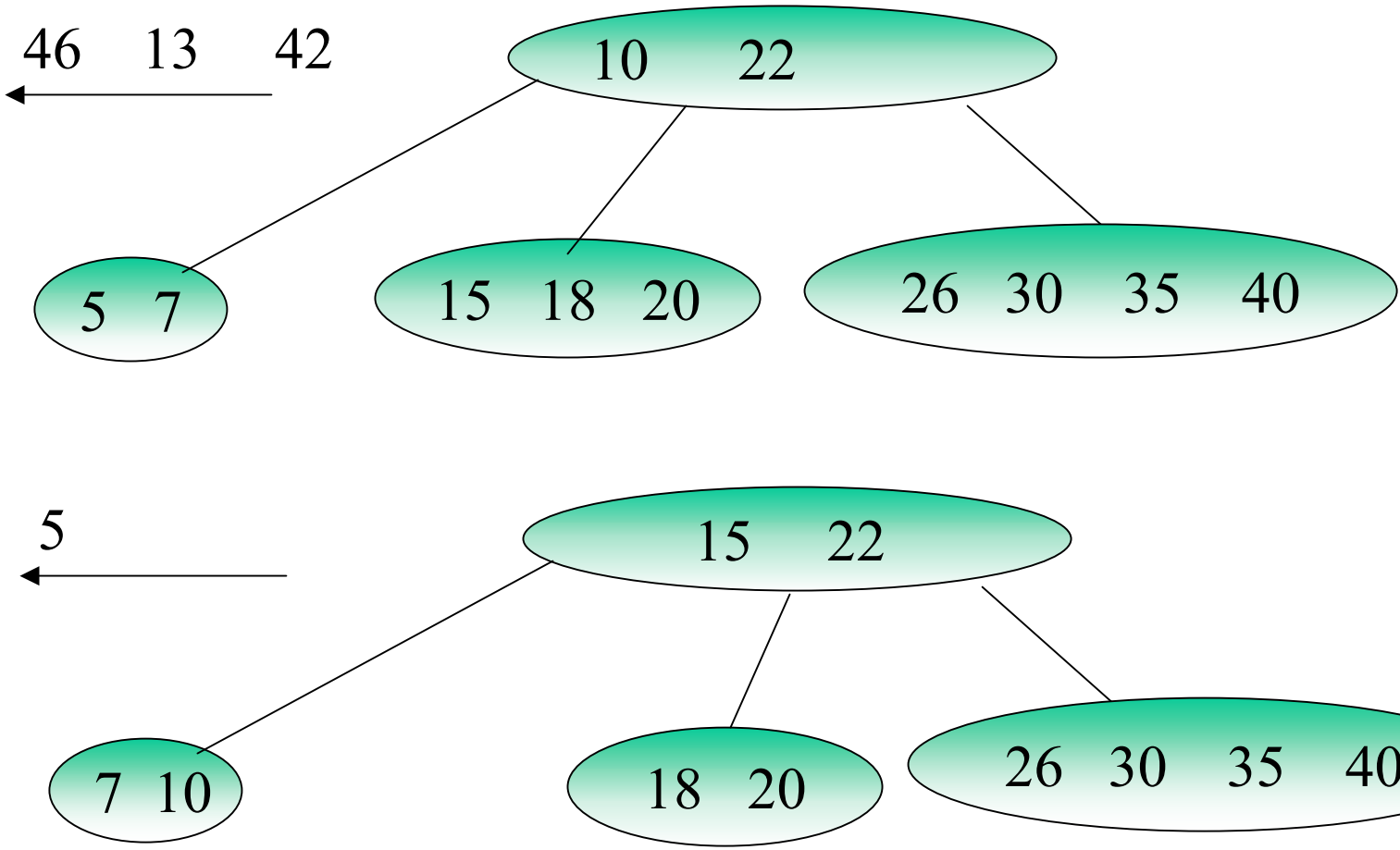


Deletion examples …

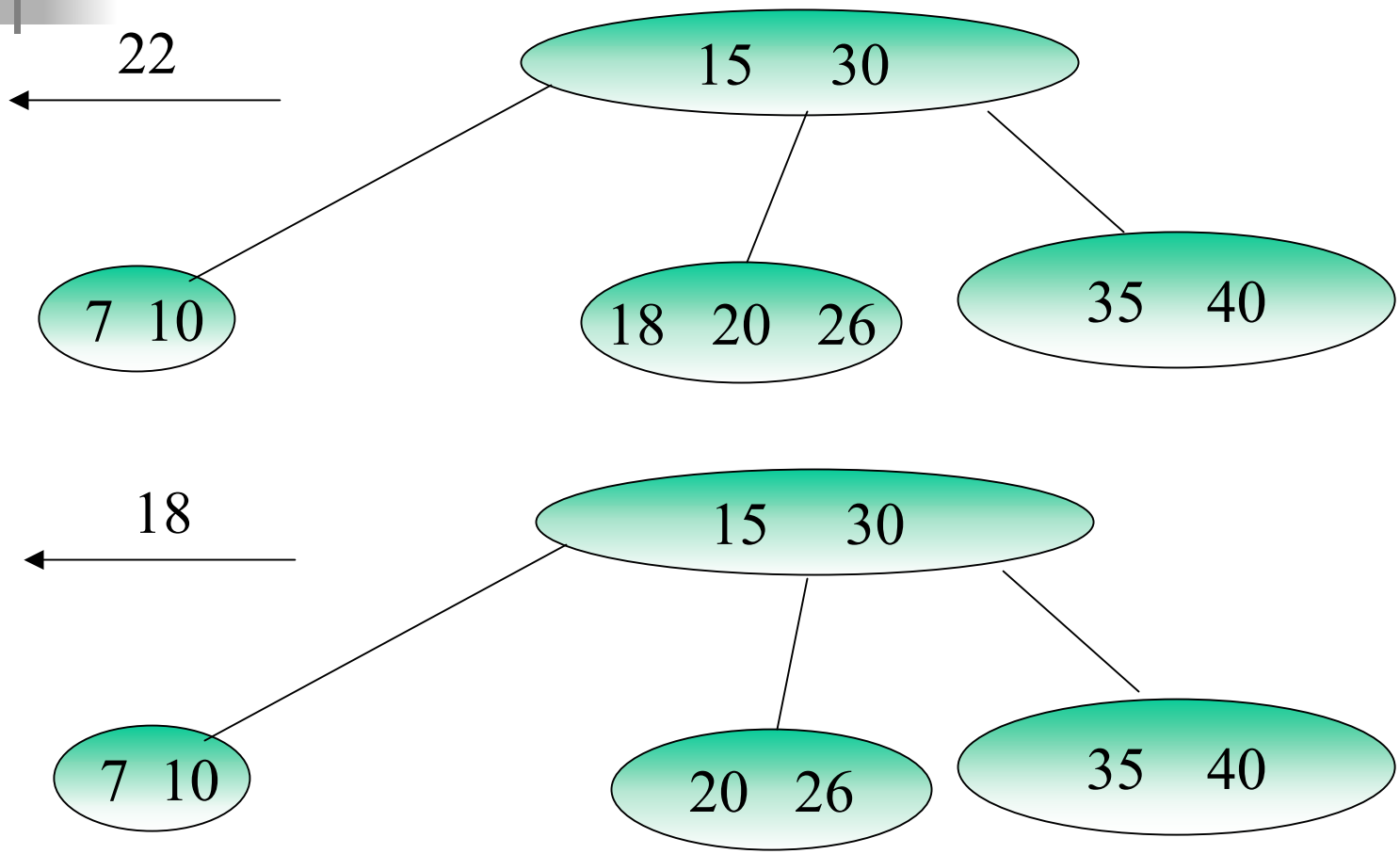




Deletion examples …



Deletion examples …



Delete 26 , 7 , 35 , 15