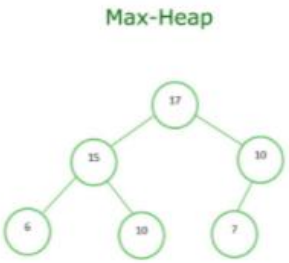
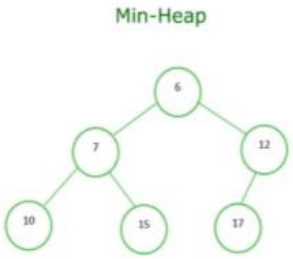
- **What is a binary Heap?**
    - Its tree which obeys the property that the root of any tree is greater than or equal to (or smaller than or equal to) all its children (heap property).
    - The **primary use** of such a data structure is to implement a priority queue.

- **Types of Binary Heaps: Min and Max Binary Heap**





- **Max Binary Heap: Insert Method**
    - **Insert method**: adds a node to the end of an array of values and bubbles up to correct the spot.
    - **bubbleUp method**: Comparing child and parent nodes, and swapping the nodes if the child is larger than the parent.

Figure 1

```
     41
    /  \
   39   33
  /  \  /
 18  27 12

[41, 39, 33, 18, 27, 12] 55
```

- Insert newNode 55

Figure 2

```
     41
    /  \
   39   33
  /  \  / \
 18  27 12 55

[41, 39, 33, 18, 27, 12, 55]
```

Figure 3
- Implement the **bubbleUp** method
- compare new node to parent and swap if newNode is larger

```
      41
     /  \
    39   55
   /  \  / \
  18  27 12 33

[41, 39, 55, 18, 27, 12, 33]
```

Figure 4

- compare new node to parent and swap if newNode is larger

```
      55
     /  \
    39   41
   /  \  / \
  18  27 12 33


[55, 39, 41, 18, 27, 12, 33]
```

Sequential Search

- **What is Sequential Search?**
  - Method that searches for data by simply starting a loop at the beginning of the list and compares each element to the data you are searching for. If you find a match, the search is over.