**Binary Search Tree Review**

* Data structure designed for **O(LogN) search**
* Can be viewed as a **recursive data structure (subtrees)**
* Have overall ordering: **values(L) < root < values(R)**
* Insert new nodes as leaves
  + We want to make it as balanced as possible via. rotation
* Delete from anywhere

**Rebalancing Trees**

An approach to balanced trees

* Insert into leaves as for simple BST, then periodically rebalance the tree

Questions: How frequently / when / how to rebalance?

* E.g. after X amount of insertions?

How to rebalance a BST?

* Maybe **move the median item to root**
  + Median / Midpoint item will always be the root – Roughly half of the nodes on the LEFT, half on the RIGHT