CS 225

#19: BST Analysis

October 11, 2017

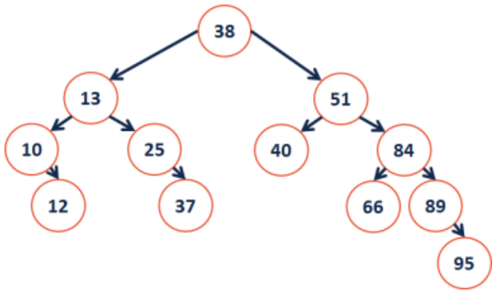
Removing an element from a BST:

`_remove(40)`

`_remove(25)`

`_remove(10)`

`_remove(13)`



One-child Remove	Two-child remove

BST.cpp

```
template <class K, class V>
void BST::_remove(TreeNode *& root, const K & key) {











}
```

BST Analysis:

Every operation we have studied on a BST depends on:

...what is this in terms of the amount of data, **n**?

Proving the relationship between h and n:

Q: What is the maximum number of nodes in a tree of height **h**?

Q: What is the minimum number of nodes in tree of height **h**?

Final BST Analysis
For every height-based algorithm on a BST:

Lower Bound:

Upper Bound:

Why use this over a linked list?

Q: How does our data determine the height?

1 3 2 4 5 7 6 vs. 4 2 3 6 7 1 5

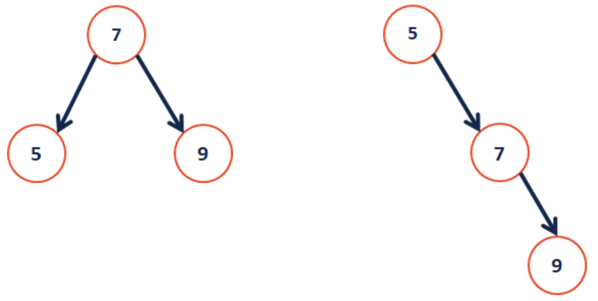
Q: How many different ways are there to insert data into a BST?

Q: What is the average height of every arrangement?

operation	BST Avg. Case	BST Worst Case	Sorted Array	Sorted List
find				
insert				
delete				
traverse				

Height Balance on BST

What tree makes you happier?



We define the **height balance** (b) of a BST to be:

We define a BST tree T to be **height balanced** if:

CS 225 – Things To Be Doing:

1. Exam #5 live now! (Theory Exam: lists, stacks, queues)
2. MP4 out today, due Monday after next
3. Labs start today, due Sunday
4. Daily POTDs