CS 225
Data Structures

March 12 – BST Balance
G Carl Evans
BST Analysis

Therefore, for all BST:

Lower bound: $h \geq O(\lg(n))$

Upper bound: $h \leq O(n)$
BST Analysis

The height of a BST depends on the order in which the data is inserted into it.

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

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

Q: What is the average height of all the arrangements?
# BST Analysis – Running Time

<table>
<thead>
<tr>
<th>Operation</th>
<th>BST Average case</th>
<th>BST Worst case</th>
<th>Sorted array</th>
<th>Sorted List</th>
</tr>
</thead>
<tbody>
<tr>
<td>find</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>insert</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>delete</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>traverse</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Height-Balanced Tree

What tree makes you happier?

Height balance: \( b = \text{height}(T_R) - \text{height}(T_L) \)

A tree is height balanced if:
BST Rotation

We will perform a rotation that maintains two properties:

1.

2.
BST Rotation Summary

- Four kinds of rotations (L, R, LR, RL)
- All rotations are local (subtrees are not impacted)
- All rotations are constant time: $O(1)$
- BST property maintained

GOAL:

We call these trees:
AVL Trees

Three issues for consideration:
- Rotations
- Maintaining Height
- Detecting Imbalance