CS 225
Data Structures

September 19 – BST Remove
G Carl Evans
Exam 2 - Topics

• C++ review pointers and references
• Lists
  • Array with runtimes
  • Linked with runtimes
  • Stacks and Queues as special cases
• Trees
  • Terminology
  • Binary
  • BST

Website with more details and practice on PL by end of the day
template<class K, class V>
TreeNode * _find(TreeNode ** root, const K & key) {

root

10 13 25 37

12 40 66 84 89 95

51

38
template<class K, class V>
void insert(TreeNode *& root, const K & key) {

```cpp
template<typename K, typename V>

TreeNode* _remove(TreeNode* & root, const K & key) {

}
```
remove(40);
remove(25);
remove(10);
remove(13);
## BST Analysis – Running Time

<table>
<thead>
<tr>
<th>Operation</th>
<th>BST Worst Case</th>
</tr>
</thead>
<tbody>
<tr>
<td>find</td>
<td></td>
</tr>
<tr>
<td>insert</td>
<td></td>
</tr>
<tr>
<td>delete</td>
<td></td>
</tr>
<tr>
<td>traverse</td>
<td></td>
</tr>
</tbody>
</table>
# 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: \( b = \text{height}(T_L) - \text{height}(T_R) \)

A tree is height balanced if: