#### **Outline**

#### CS 331 — AVL Trees

Dr. Mattox Beckman

Illinois Institute of Technology Department of Computer Science

Spring, 2012

- Introduction
  - Objectives
- **Balancing Trees**
- Rotations

	<b>←</b> □ <b>→</b>	<b>◆御 ▶ ◆ 恵 ▶ ◆ 恵 → り</b> へで			
Dr. Mattox Beckman (IIT)	CS 331 — AVL Trees	Spring, 2012 1 / 16	Dr. Mattox Beckman (IIT)	CS 331 — AVL Trees	Spring, 2012 2 / 16
	Introduction Objectives			Introduction Objectives	
Objectives	Motivation				

# Objectives

You should be able to...

- Explain why height-balanced trees are necessary.
- Explain how to perform two of the four kinds of rotations:
  - left, right
- Identify the proper kind of rotation for a particular situation.

Do you remember how long it takes...

- To insert an element into a BST on average?
- To insert an element into a BST worst case?
- To delete an element from a BST on average?
- To delete an element from a BST worst case?

### Motivation

### Some Good Insertions

**Insert These Nodes** 

8 6 16 30 7 2 12

Do you remember how long it takes...

- To insert an element into a BST on average? //  $\mathcal{O}(\lg n)$
- To insert an element into a BST worst case? //  $\mathcal{O}(n)$
- To delete an element from a BST on average? //  $\mathcal{O}(\lg n)$
- To delete an element from a BST worst case? //  $\mathcal{O}(n)$

Dr. Mattox Beckman (IIT)

CS 331 — AVL Trees
Spring, 2012 4 / 16
Dr. Mattox Beckman (IIT)
CS 331 — AVL Trees
Spring, 2012
Introduction
Objectives
Objectives

### Some Good Insertions

# Some Good Insertions

Insert These Nodes 8 6 16 30 7 2 12 Insert These Nodes 8 6 16 30 7 2 12

8



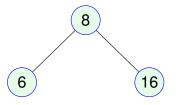
CS 331 — AVL Trees

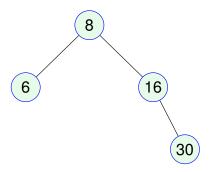
# Some Good Insertions

### Some Good Insertions

**Insert These Nodes** 8616307212

**Insert These Nodes** 8616307212





◆□▶◆□▶◆■▶◆■▶ ■ 900

◆ロ → ◆昼 → ◆ き → ● り へ ○

Dr. Mattox Beckman (IIT)

CS 331 — AVL Trees Objectives Introduction

Spring, 2012 5/16 Dr. Mattox Beckman (IIT)

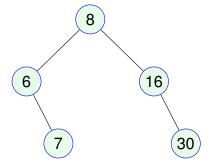
CS 331 — AVL Trees Introduction

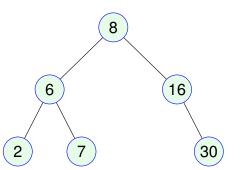
# Some Good Insertions

# Some Good Insertions

**Insert These Nodes** 8616307212

**Insert These Nodes** 8616307212



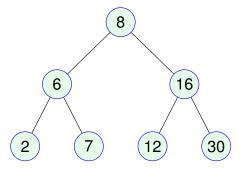


# Some Good Insertions

# Some Good Insertions

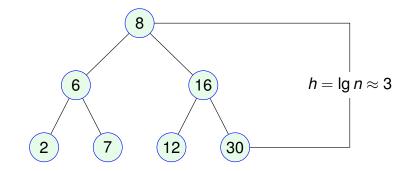
**Insert These Nodes** 8 6 16 30 7 2 12

**Insert These Nodes** 8 6 16 30 7 2 12



CS 331 — AVL Trees

Introduction Objectives



**◆□▶◆□▶◆壹▶ 壹 り**900 Dr. Mattox Beckman (IIT)

◆ロト ◆部 ト ◆ 恵 ト ◆ 恵 ・ 夕 へ ② CS 331 — AVL Trees

Some Bad Insertions

# Some Bad Insertions

**Insert These Nodes** 

Dr. Mattox Beckman (IIT)

30 2 16 6 7 12 8

**Insert These Nodes** 30 2 16 6 7 12 8

30

Objectives

Introduction

6/16

Spring, 2012

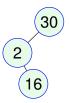
# Some Bad Insertions

### Some Bad Insertions

**Insert These Nodes** 30 2 16 6 7 12 8

**Insert These Nodes** 30 2 16 6 7 12 8





◆□▶◆□▶◆■▶◆■▶ ■ 900

◆ロト ◆部 ト ◆ 恵 ト ◆ 恵 ・ 夕 へ ②

Dr. Mattox Beckman (IIT)

CS 331 — AVL Trees Introduction Objectives

Spring, 2012 6/16 Dr. Mattox Beckman (IIT)

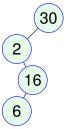
CS 331 — AVL Trees Introduction

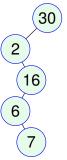
# Some Bad Insertions

# Some Bad Insertions

**Insert These Nodes** 30 2 16 6 7 12 8

**Insert These Nodes** 30 2 16 6 7 12 8





CS 331 — AVL Trees

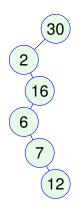
Spring, 2012 6 / 16

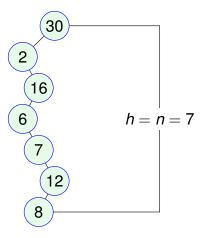
# Some Bad Insertions

### Some Bad Insertions

**Insert These Nodes** 30 2 16 6 7 12 8

**Insert These Nodes** 30 2 16 6 7 12 8





4□ > 4□ > 4□ > 4□ > 4□ > 4□

∢ロト∢部ト∢医ト∢医トー室

Dr. Mattox Beckman (IIT)

CS 331 — AVL Trees

Spring, 2012 6/16 Dr. Mattox Beckman (IIT)

CS 331 — AVL Trees

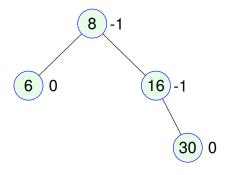
Balancing Trees

Balancing Trees

### Balance

# Some Insertions, with Balances

• The balance of a node is the depth of the left subtree minus the depth of the right subtree.



- Depth is the longest path from the node to a leaf.
- Leaves always have balance of zero.

**Insert These Nodes** 30 2 16 32 37 12



# Some Insertions, with Balances

# Some Insertions, with Balances

**Insert These Nodes** 30 2 16 32 37 12

**Insert These Nodes** 30 2 16 32 37 12





**◆□▶◆□▶◆壹▶ 壹 り**900 ◆ロ → ◆昼 → ◆ き → ● り へ ○ Spring, 2012 8/16 Dr. Mattox Beckman (IIT) CS 331 — AVL Trees

Dr. Mattox Beckman (IIT)

CS 331 — AVL Trees

Balancing Trees

Balancing Trees

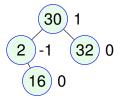
# Some Insertions, with Balances

# Some Insertions, with Balances

**Insert These Nodes** 30 2 16 32 37 12

**Insert These Nodes** 30 2 16 32 37 12





CS 331 — AVL Trees

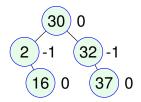
Spring, 2012 8 / 16

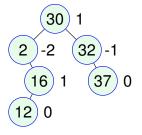
## Some Insertions, with Balances

# Some Insertions, with Balances

**Insert These Nodes** 30 2 16 32 37 12

**Insert These Nodes** 30 2 16 32 37 12





◆ロト ◆部 → ◆恵 → ・恵 ・ かなべ

Dr. Mattox Beckman (IIT)

CS 331 - AVL Trees

Spring, 2012

Dr. Mattox Beckman (IIT)

CS 331 — AVL Trees

CS 331 — AVL Trees

Balancing Trees

8/16

Balancing Trees

# **Updating Balance During Add**

# Add Example 1

### • Perform an add as normal, using recursion.

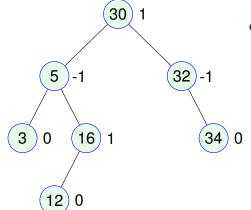
#### • The leaf will have balance zero.

#### Upon return:

- If you went left, increment your balance.
- If you went right, decrement your balance.
- If the balance becomes zero, stop updating balances. (Why?)
- If the balance is +/-1, return to the parent.
- If the balance is +/-2, rebalance the node.

# Example

Insert a 1.



The node 1 goes to the left of 3.

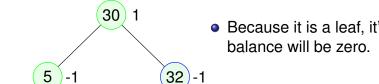
Spring, 2012

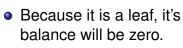
# Add Example 1

# Add Example 1

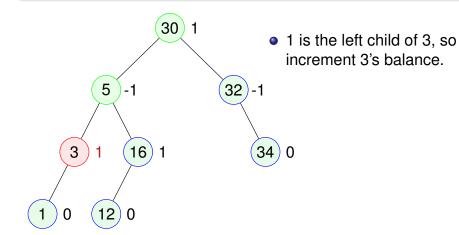
Example

Insert a 1.









◆ロト ◆部 → ◆恵 → ・恵 ・ かなべ Dr. Mattox Beckman (IIT)

(34) 0

CS 331 — AVL Trees Spring, 2012 10/16 Balancing Trees

Dr. Mattox Beckman (IIT)

CS 331 — AVL Trees Balancing Trees

# Add Example 1

Dr. Mattox Beckman (IIT)

3

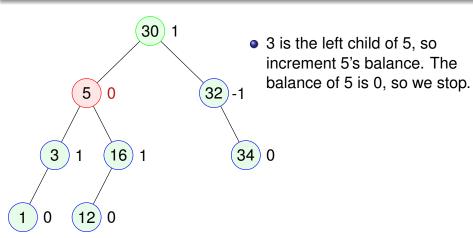
(16) 1

(12) 0

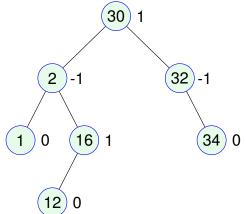
# Examples

Example

Insert a 1.



Example Insert a 33.



• The node 33 goes to the left of 34.

# Examples

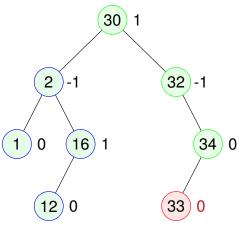
# Examples

Example

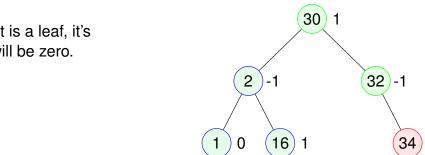
Insert a 33.

Example

Insert a 33.



• Because it is a leaf, it's balance will be zero.



• 33 is the left child of 34, so increment 34's balance.

4□ > 4□ > 4□ > 4□ > 4□ > 4□

Spring, 2012

11/16

Dr. Mattox Beckman (IIT)

(12) 0

CS 331 — AVL Trees

(33) 0

Balancing Trees

Spring, 2012

Examples

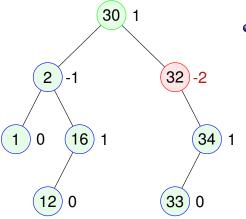
# Examples

Dr. Mattox Beckman (IIT)

Dr. Mattox Beckman (IIT)

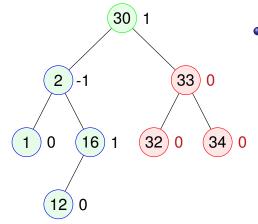
# Example

Insert a 33.



• 34 is the right child of 32, so decrement 32's balance. This node is out of balance, so we will rebalance here.

Example Insert a 33.



Here is the result of rebalancing. Let's talk about that next....

CS 331 — AVL Trees

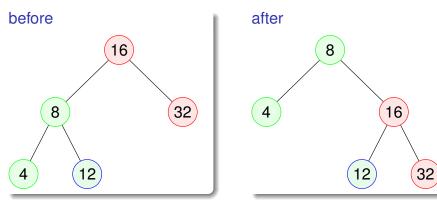
Balancing Trees

Rotations Rotations

## Effect of a rotation

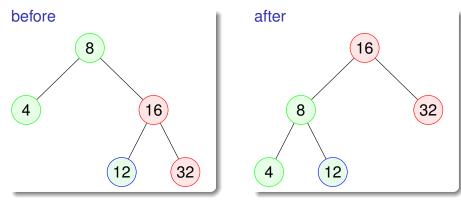
#### Effect of a rotation

• This is a Right Rotation.



• What happens to the balance of the root?

• This is a Left Rotation. It should look familiar.



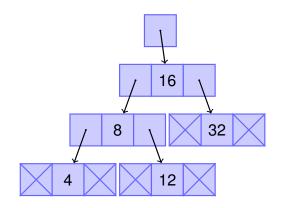
• What happens to the balance of the root?



### Effects of Rotations

# Implementing a Right Rotation

- A Left Rotation adds 2 to the balance of the node. Use it when the balance is −2.
- A Right Rotation subtracts 2 from the balance of the node. Use it when the balance is 2.
- The "heavy" part of the tree needs to be on the *outer* side for this to work.

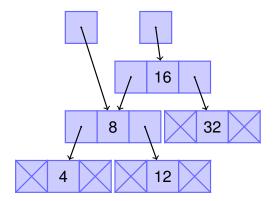


```
curr.balance += 2;
curr.left.balance++;
tmp = curr.left;
curr.left =
    curr.left.right;
tmp.right = curr;
curr.parent = tmp;
```

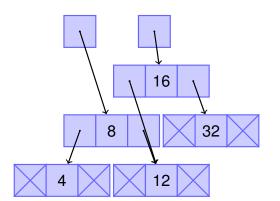
Rotations Rotations

# Implementing a Right Rotation

# Implementing a Right Rotation



```
curr.balance += 2;
curr.left.balance++;
tmp = curr.left;
curr.left =
 curr.left.right;
tmp.right = curr;
curr.parent = tmp;
```



```
curr.balance += 2;
curr.left.balance++;
tmp = curr.left;
curr.left =
  curr.left.right;
tmp.right = curr;
curr.parent = tmp;
```

◆ロト ◆園 ト ◆夏 ト ◆夏 ト 夏 り へ ②

◆ロ → ◆昼 → ◆ き → ● り へ ○

Dr. Mattox Beckman (IIT)

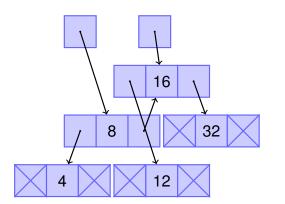
CS 331 — AVL Trees Rotations

Spring, 2012 15 / 16 Dr. Mattox Beckman (IIT)

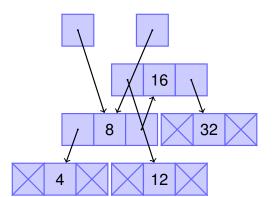
CS 331 — AVL Trees

Rotations

### Implementing a Right Rotation



```
curr.balance += 2;
curr.left.balance++;
tmp = curr.left;
curr.left =
 curr.left.right;
tmp.right = curr;
curr.parent = tmp;
```



Implementing a Right Rotation

```
curr.balance += 2;
curr.left.balance++;
tmp = curr.left;
curr.left =
  curr.left.right;
tmp.right = curr;
curr.parent = tmp;
```

You have to update the parent's link also.

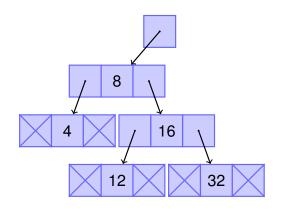
Rotations Rotations

# Implementing a Right Rotation

# **Bad Insertions with Rotations**

#### **Insert These Nodes**

123456



```
curr.balance += 2;
curr.left.balance++;
tmp = curr.left;
curr.left =
    curr.left.right;
tmp.right = curr;
curr.parent = tmp;
```

4□ ► 4₫ ► 4 분 ► 4 분 ► 9 < 0</p>

4□ > 4□ > 4 = > 4 = > = 900

Dr. Mattox Beckman (IIT)

CS 331 — AVL Trees
Rotations

Spring, 2012 15 / 16

Dr. Mattox Beckman (IIT)

CS 331 — AVL Trees

Rotations

Spring, 2012

16 / 1

Bad Insertions with Rotations

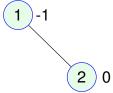
### **Bad Insertions with Rotations**

#### **Insert These Nodes**

123456

1 0

Insert These Nodes 1 2 3 4 5 6



CS 331 — AVL Trees

Spring, 2012

Rotations Rotations

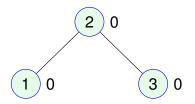
### **Bad Insertions with Rotations**

## **Bad Insertions with Rotations**

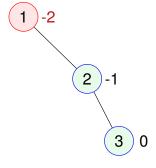
**Insert These Nodes** 

123456

**Insert These Nodes** 123456



• Try inserting the next two yourself before looking ahead!



**◆□▶◆□▶◆壹▶ 壹 り**900

◆ロ → ◆昼 → ◆ き → ● り へ ○

Dr. Mattox Beckman (IIT) CS 331 — AVL Trees Rotations

Spring, 2012 16 / 16

CS 331 — AVL Trees Dr. Mattox Beckman (IIT)

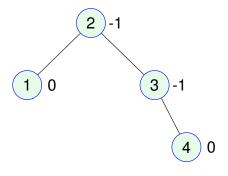
Rotations

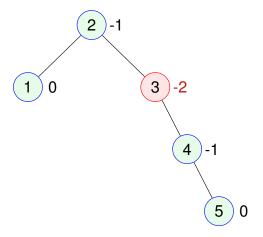
# **Bad Insertions with Rotations**

# **Bad Insertions with Rotations**

**Insert These Nodes** 123456

**Insert These Nodes** 123456

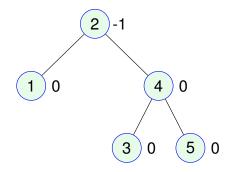




Rotations Rotations

### **Bad Insertions with Rotations**

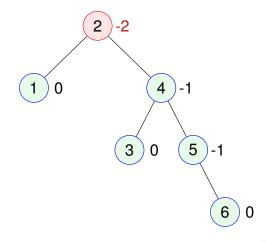
**Insert These Nodes** 123456



• Try inserting the next two yourself before looking ahead!

### **Bad Insertions with Rotations**

**Insert These Nodes** 123456



**◆□▶◆□▶◆壹▶ 壹 り**900 Dr. Mattox Beckman (IIT) CS 331 — AVL Trees Spring, 2012 16 / 16

Rotations

Dr. Mattox Beckman (IIT)

CS 331 — AVL Trees

◆□ > ◆□ > ◆□ > ◆□ > ◆□ > ◆□

Spring, 2012

# **Bad Insertions with Rotations**

**Insert These Nodes** 123456

