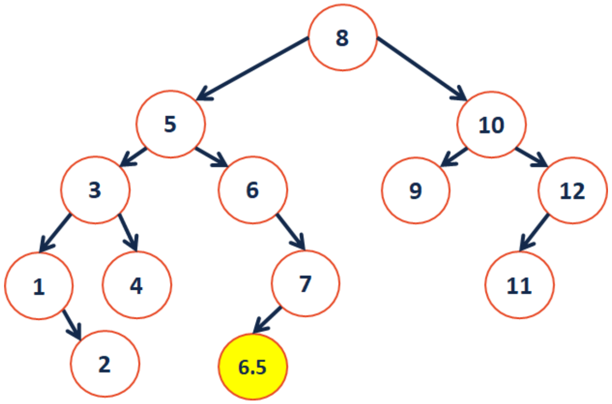
CS 225

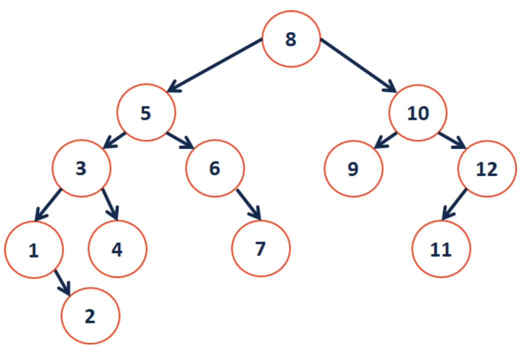
#20: AVL Analysis

October 12, 2018 · *Wade Fagen-Ulmschneider*

AVL Insertion



AVL Removal



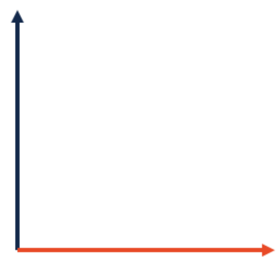
Running Times:

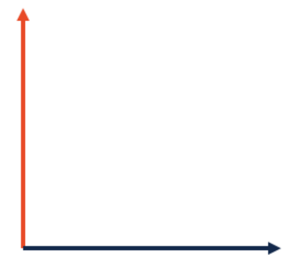
	AVL Tree
find	
insert	
remove	

Motivation:

Big-O is defined as:

Let **f(n)** describe the height of an AVL tree in terms of the number of nodes in the tree (**n**). Visually, we can represent the big-O relation:





f(n) ≤ c × g(n): Provides an <u>upper bound</u>:

The height of the tree, **f(n)**, will always be <u>less than</u> **c × g(n)** for all values where **n** > **k**.

f^{-1}(h) ≥ c × g^{-1}(h): Provides a <u>lower bound</u>:

The number of nodes in the tree, **f^{-1}(h)**, will always be <u>greater than</u> **c** × **g^{-1}(h)** for all values where **n** > **k**.

Plan of Action:
Goal: Find a function that defines the lower bound on **n** given **h**.

Given the goal, we begin by defining a function that describes the smallest number of nodes in an AVL of height **h**:

Theorem:
An AVL tree of height **h** has at least ______________________.

I. Consider an AVL tree and let **h** denote its height.

II. Case: _________________

III. Case: __________________

IV. Case: _________________

Inductive hypothesis (IH):

Proving our IH:

V. Using a proof by induction, we have shown that:

...and by inverting our finding:

Summary of Balanced BSTs:

Advantages	Disadvantages

CS 225 – Things To Be Doing:
1. Theory Exam 2 is ongoing!
2. MP4 extra credit submission ongoing – due Monday!
3. lab_huffman is due on Sunday
4. Daily POTDs are ongoing!