CS 367 - Introduction to Data Structures  
Thursday, November 10th, 2016

Midterm Exam 2: Tuesday, November 15th, 5:00 pm – 7:00 pm
- Lecture 1: Room 105 of Psychology Building
- Lecture 2: Room 6210 of Sewell Social Sciences
- Lecture 3: Room 132 of Noland Hall
- Bring ID and #2 Pencils
- See Exams page on Canvas

Program 4 due 10 pm Sunday, November 27th

Last Time
- Binary Search Tree (BST)
  - BSTnodes
  - BST class
  - implementing print
  - implementing lookup, insert, delete
  - complexities of BST methods

CS Options/Courses

Today
- Classifying Binary Trees
- Balanced Search Trees
- Red-Black Trees
  - tree properties
  - print, lookup
  - insert

Next Tuesday
- Exam 2 Sample Questions
- And Exam 2
Classifying Binary Trees

Full

Complete

Height-balanced

Balanced
Practice - Classifying Binary Trees

→ Identify which trees below are full, complete and/or height balanced.

A

B

C

D

E

F
Balanced Search Trees

Goal:

Idea:

AVL

BTrees
Red-Black Trees (RBT)

RBT:

Example:

Red-Black Tree Properties

- root property
- red property
- black property

Red-Black Tree Operations

- print
- lookup
- insert
- delete
Inserting into a Red-Black Tree

Goal: insert key value K into red-black tree T
and _________________________________.

If T is Empty

If T is Non-Empty
  • step down tree as done for BST
  • add a leaf node containing K as done for BST, and ______________________

→ Which of the properties might be violated as a result of inserting a red leaf node?
  root property
  black property
  red property

Non-Empty Case 1: K's parent P is black
Non-Empty Case 2

Non-Empty Case 2: K’s parent P is red

Fixing an RBT

Tri-Node Restructuring is done if P’s sibling S is null

Recoloring is done if P’s sibling S is red
Practice

1. Starting with an empty RBT, show the RBT that results from inserting 7 and 14.

2. Redraw the tree from above and then show the result from inserting 18.

3. Redraw the tree from above and then show the result from inserting 23.

4. Redraw the tree from above and then show the result from inserting 1 and 11.

5. Redraw the tree from above and then show the result from inserting 20.
More Practice!

6. Redraw the tree from the previous page and then show the result from inserting 29.

7. Insert the same list of values into an empty BST: 7, 14, 18, 23, 1, 11, 20, 29

What does this demonstrate about the differences between a BST and RBT?
More Practice?

8. Show the result from inserting 25 in the RBT below.

9. Redraw the tree from above and then show the result from inserting 27.
Cascading Fixes

Fixing an RBT UPDATED!

**Recoloring** is done if P’s sibling S is red

1. change P & S to black
2. if G is the root – done
   otherwise change G to red

**Tri-Node Restructuring** is done if P’s sibling S null

---

Copyright 2014-2016 Jim Skrentny  
CS 367 (F16): L20 - 11
RBT Complexity

print
lookup
insert