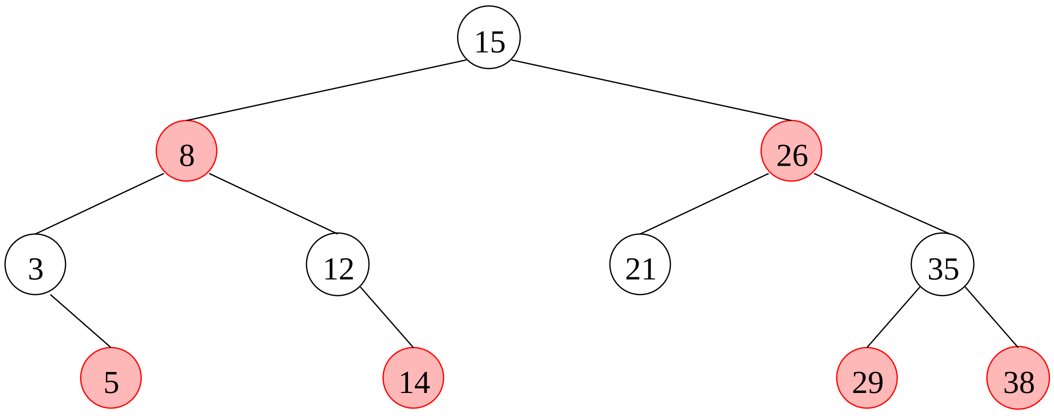
Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Harald Räcke
Chintan Shah

Wintersemester 2012
Übungsblatt 4
15 November 2012

Effiziente Algorithmen und Datenstrukturen I

Aufgabe 1 (10 Punkte)

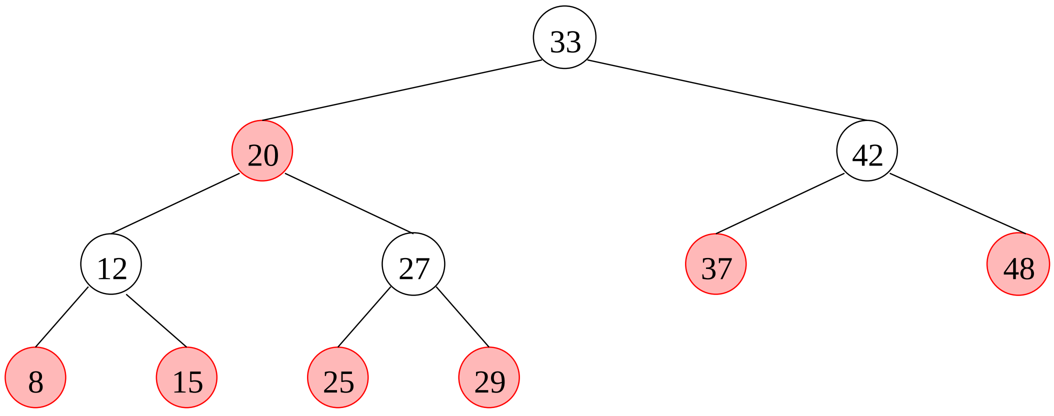
Carry out the following operations sequentially on the red-black tree shown below so that it remains a red-black tree and show what the tree looks like after each operation(always carry out each operation on the result of the previous operation):



1. Insert 10
2. Delete 29
3. Delete 21
4. Delete 3

Aufgabe 2 (10 Punkte)

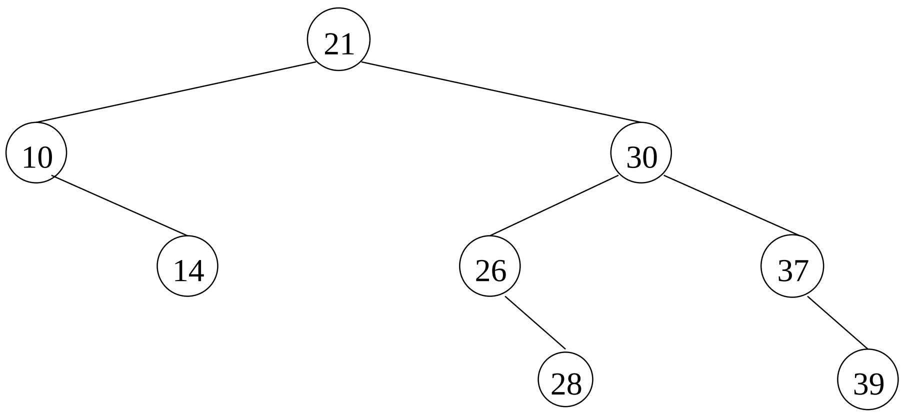
Carry out the following operations sequentially on the red-black tree shown below so that it remains a red-black tree and show what the tree looks like after each operation(always carry out each operation on the result of the previous operation):



1. Insert 5
2. Delete 27
3. Insert 27

Aufgabe 3 (10 Punkte)

Is the following binary tree an AVL tree? If not, rebalance it to an AVL tree.



Carry out the following operations sequentially on the tree so that it remains an AVL tree and show what the tree looks like after each operation(always carry out each operation on the result of the previous operation):

1. Insert 16
2. Delete 30
3. Insert 22
4. Insert 27

Aufgabe 4 (10 Punkte)

1. While inserting a node z in a red-black tree, what happens if z and its parent are red but z has no grandparent?
2. While deleting a node from an AVL tree, if node z is the only unbalanced node($|balance_{node}| > 1$) with a balance of -2(the right subtree of z has more height) and the right child of z has a balance of 0, how do we rebalance this tree?
3. If we insert a node in a red-black tree and then immediately delete the same node, is the resulting tree always identical to the original tree? Explain.