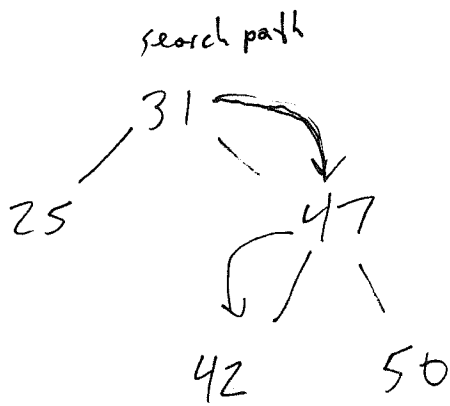


4.

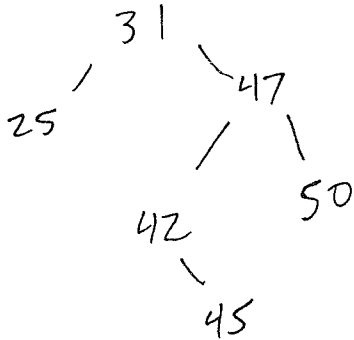


add(45)

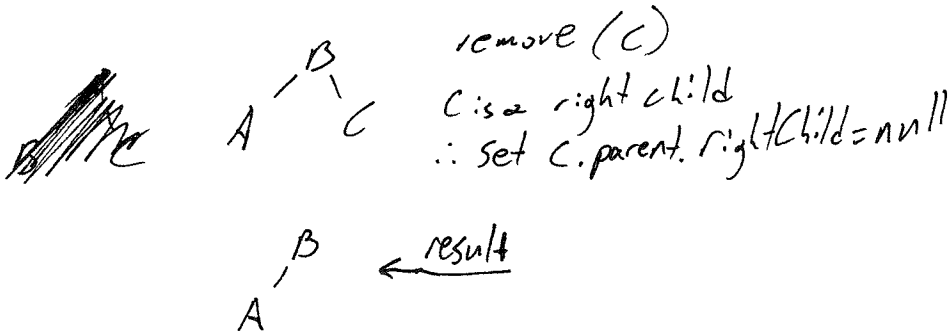
temp	comparison	direction
31	> 0	right
47	< 0	left
42	> 0	right

↪ insert as right child of 42

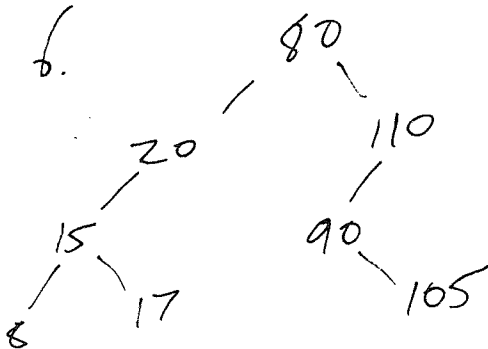
result



5.



6.

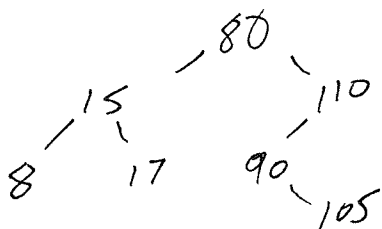


remove(20)

replace element to be removed
 with only child

Set 20's child as 20's parent's child
 same side as 20 was on!

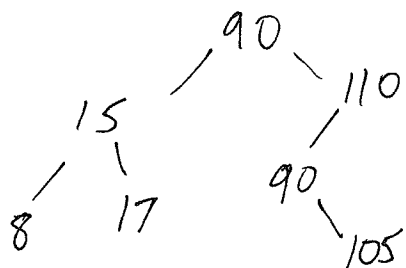
Result



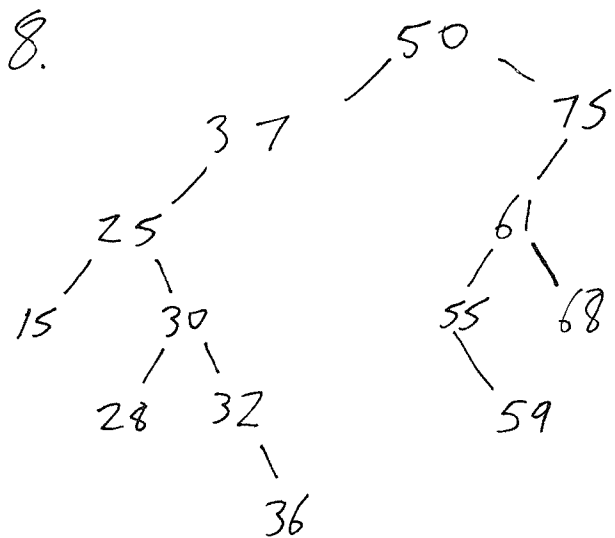
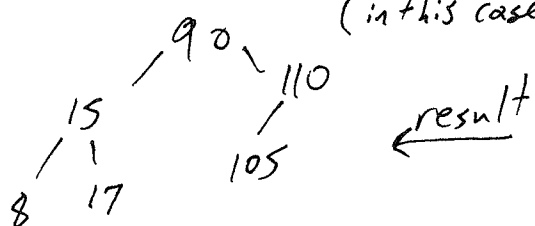


remove(80)
 inorder predecessor = 17
 inorder successor = ~~90~~ 90

1. Copy successor's value to deletion position



2. Handle deletion of successor
 (in this case, a non-leaf w/ 1 child)

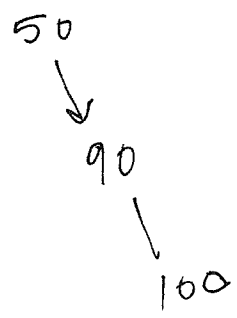


element	Successor	
61	68	} have right children: right then left as far as possible
25	28	
55	59	
37	50	} no right children: up-left as far as possible then parent
28	30	
75	null	
36	37	

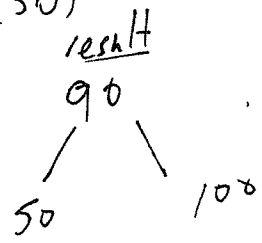
75 is greatest number in tree \therefore successor is null (parent of 50)

9.

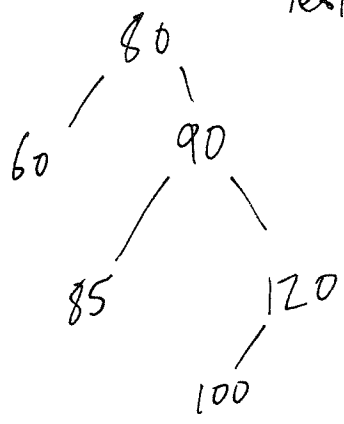
left rotation(50)



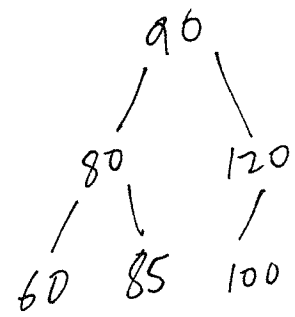
→



left rotation(80)

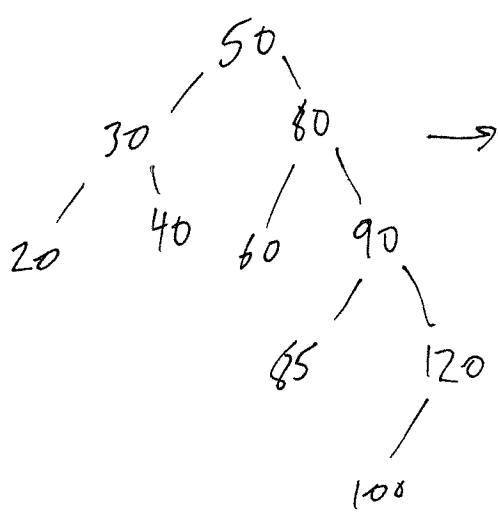


→

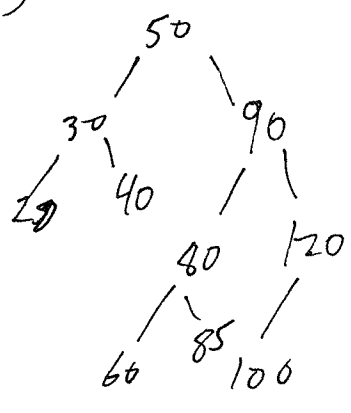


note: 85 goes from being a left child to a right child
also right subtree to left subtree

left rotation(80)

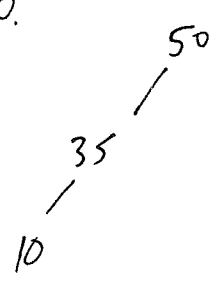


→

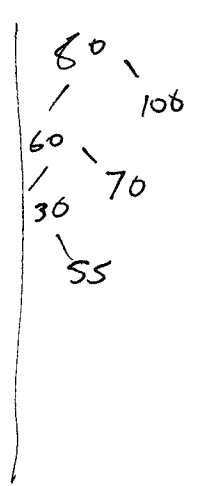
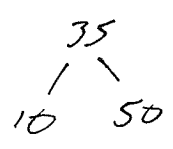


10.

right rotation(50)

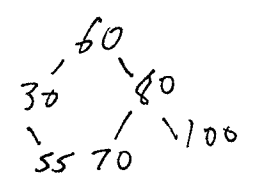


→



→

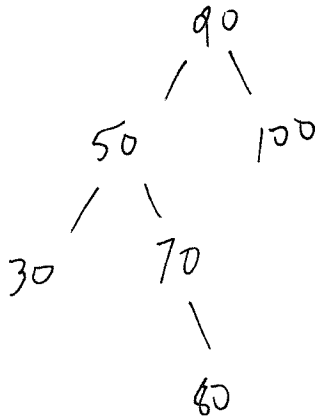
right rotation(80)



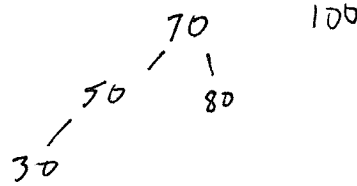
11.

left Rotate (50)

result



height(t) = 3

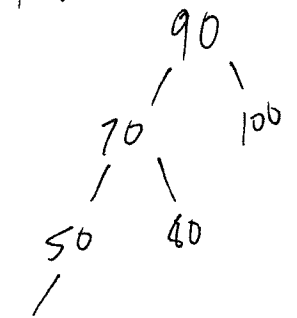


height(t) = 3

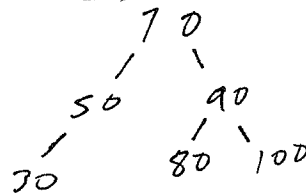
12. result from 11

right Rotate (90)

result



height(t) = 3



height(t) = 2

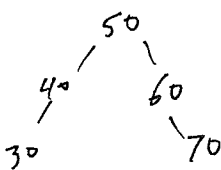
13.

Three AVL trees

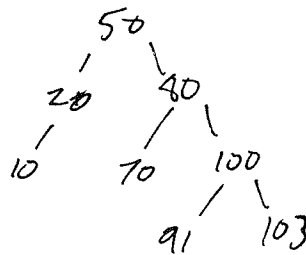
a.

25

b.

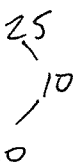


c.



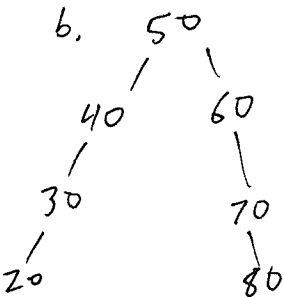
Three non-AVL Trees

a.



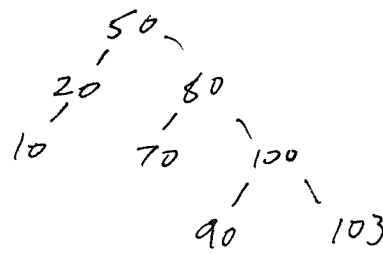
height(left) = -1
height(right) = 1

b.



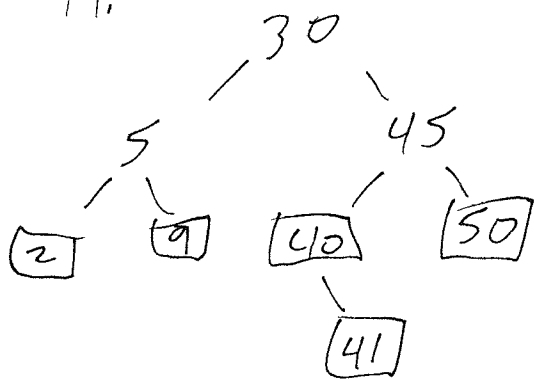
left/right subtrees are
not AVL trees

c.



height(left) = 1
height(right) = 3

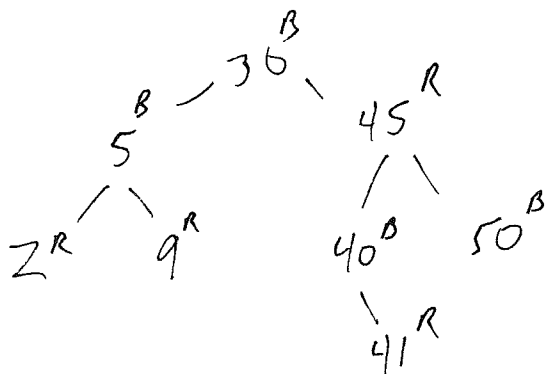
14.



5 paths to leafs or
single-child nonleafs (boxed)

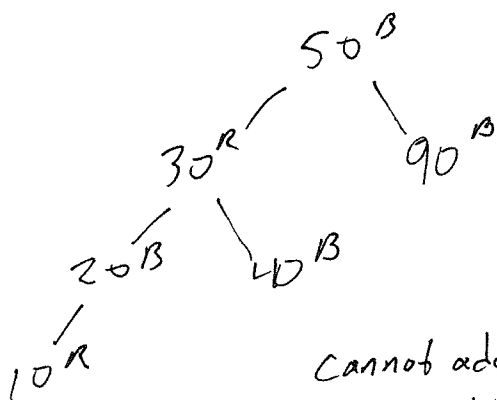
1. 30, 5, 2
2. 30, 5, 9
3. 30, 45, 40, 41
4. 30, 45, 40
5. 30, 45, 50

15.



1. No reds have red children
 2. All paths have 2 black
- ∴ It is a red-black tree

16. Somewhat unbalanced R-B Tree



cannot add (15) without rebalancing
new Red would violate red rule
new Black would violate path rule