UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

COS 212 Tutorial 4: Version B

- 09/03/2012
- 50minutes
- 1 questions for a total of 17 marks.

Name:_____

Student/staff Nr:_____

Marker (office use):_____

Question 1 Balancing ..... (17 marks)

1.1 Write the pseudo code to perform a `left` rotation of a node N about its parent P. (3)

Solution:

```
tmp = root
while(tmp != null)
        if tmp has a RIGHT child (textbook says left)
                rotate child about tmp
                set tmp to the original RIGHT child
        else tmp = tmp.LEFT
```

Answer:

1.2 Rewrite the DSW algorithm's `createBackbone` pseudo code so that a backbone will be created where the largest element in the tree will be the root of the backbone. (4)

Solution: The idea is that left rotations are performed instead of right rotations. Look for the following (capital letters are the differences from the textbook):

```
tmp = root
while(tmp != null)
        if tmp has a RIGHT child (textbook says left)
                rotate child about tmp
                set tmp to the original RIGHT child
        else tmp = tmp.LEFT
```

Answer:

1.3 In the DSW algorithm's `createPerfectTree` phase, only every second node in the backbone is rotated about its parent to create a new tree, except for the very last node in the backbone, why is this? (1)

Solution: Will cause the tree to become lopsided to the left again.

Answer:

Question 2 AVL Trees ..... (**??** marks)

2.1 Insert the following keys, in the given order, into an initially empty AVL tree. Only draw your final tree: (3)

`55,60,63,65,70,50,40,30,25,20`

Answer:

Solution: 1 Tree is perfectly balanced

1 Correct root

1 is a BST

2.2 After an insert operation on an AVL tree causes any node's balance factor to change, could the node's balance factor be used to determine into which subtree the insert occurred? Motivate your answer. (1)

Answer:

2.3 Assume an AVL tree exists with three nodes G,P and N. G has as its right child the node P and P has as its left child N. The balance factors for G, P and N are +1, -1 and -1 respectively. A node is removed from the left subtree of G causing an imbalance in the tree:

a) Rebalance the tree by completing the following, substituting the numerals for either N, P or G: (4)

Rotate (i) about (ii) and then (iii) about (iv)

Answer:

b) What is the balance factor for P after the rotation of the previous question has been completed? (1)

Answer:

2.4 When a rotation needs to be performed in an AVL tree, the choice of which nodes take part and the order in which they rotate matters. Why is this? (1)

> **Solution:** If done incorrectly we will end up with an unbalanced tree. Correct rotations lead to tree being as balanced as possible.

Answer:

2.5 Assume the tree in figure 1 is an AVL tree, redraw the tree after the root has been deleted. (3)

> **Solution:** EITHER:
>
> 1 mark for 35 as new root
>
> 1 mark for node 10 being left child of 35 AND 5 is left of 10 AND 25 is right of 10
>
> OR:
>
> 1 for 60 in the root
>
> 1 Node 78: Is the left of 100
>
> 1 77 is left child of 78 AND 81 is right of 78

Answer:

Solution: 1 222 is replaced by 250 as 100's right

1 Rotate Child

1 Rotate Child again about its new parent

Question 3 Self Restructuring Trees ..... (?? marks)

3.1 The naive approach for the `moving to the root` strategy for self-restructuring trees suggests that an accessed node simply be rotated about its parents until it is the root. Discuss the problem with this strategy. (2)

Answer:

3.2 Assume the tree in figure 2 is to be used for splaying. Draw the final tree after an access to node 5 has occurred. (4)

Solution: 2 for homogeneous splay

2 for homogeneous splay again

Answer:

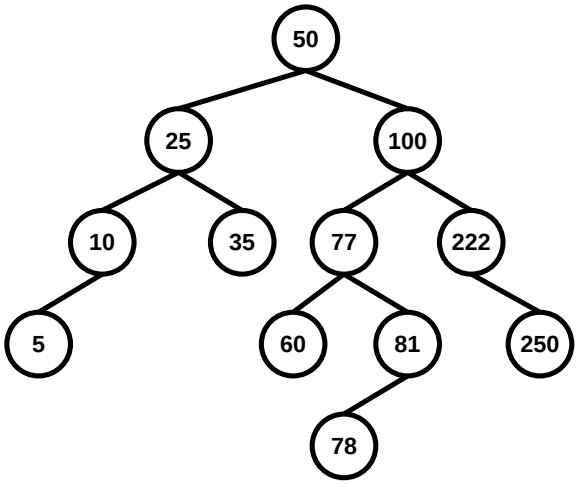

Figure 1: AVL Tree

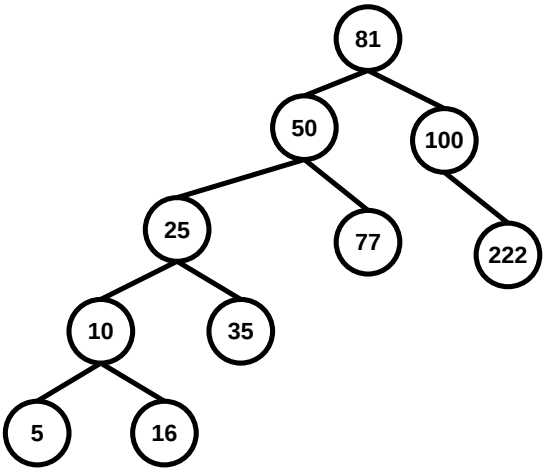

Figure 2: Splay Tree