Quiz 8 - AVL Trees

CS 14 - Data Structures

May 1, 2013

Questions:

1. State precisely the two invariants that every AVL tree must hold. Also give a sentence justifying why that particular invariant is useful.

   (a)

   (b)

2. Name an advantage and a disadvantage of AVL trees compared to binary search trees.

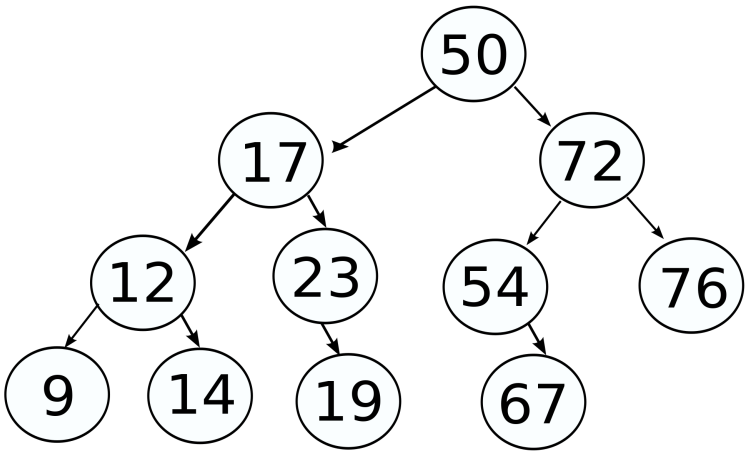
3. If you're using a Map ADT implemented by someone else, would you prefer they implemented it with a BST or AVL tree? Why?

4. Write the best, worst, and average case run times for each of these functions.

	best	average	worst
(a) insert			
(b) delete			
(c) search			

Note: For an exam, you MUST be able to give code/pseudocode for all of these functions. You should also be able to write a recursive function that checks to see if a Tree is a valid AVL tree.

5. Given this tree structure:



(a) Assuming the tree is a binary search tree, and not an AVL tree, draw the tree structure created by the following code. Hint: Don't try to do it all at once in your head. Draw each insertion/deletion as an entirely separate tree.

```
t->insert(71);
t->delete(19);
t->delete(23);
t->delete(50);
t->insert(77);
t->insert(78);
t->insert(79);
t->insert(80);
```

(b) Repeat part (a) assuming the tree is an AVL tree.

Note: You will need a lot of extra paper to complete this question. On an exam, you will be given half a page to complete each intermediate diagram.