Due: Wednesday, 6/12/2019

Answer the following questions and submit typeset solutions at the beginning of class on the due date. As stated in the Honor Code on the syllabus, any outside sources must be listed under the corresponding problem. Further, your final submission must be completely your own work. Use the largest left child for any tree operation requiring a choice of a new root. All key additions/deletions to any tree should build cumulatively (i.e., do not start with the original tree each time), unless otherwise directed.

1. [8 points] Consider the following expression tree:

   ![Expression Tree]

   a. List the nodes as they are encountered in a pre-order traversal
   b. List the nodes as they are encountered in an in-order traversal
   c. List the nodes as they are encountered in a post-order traversal
   d. Evaluate the expression tree

2. [8 points] Convert the following post-order expression to an expression tree using the algorithm outlined in the course slides:

   \[ a \ b \ + \ a \ b \ - \ * \ 2 \ + \]

   Show the stack during the execution of the algorithm (items that are pushed onto the stack may be combined in one diagram; however, the stack must be redrawn whenever a pop occurs).

3. [8 points] Show the result of inserting 14, 6, 17, 3, 11, 9, 2, 15, 18, 13 into an empty binary search tree. Show the tree after each insertion.

4. [8 points] Prove the following statement by induction on the height of a binary tree: A perfect binary search tree of height \( h \) has \( 2^h \) leaves.
5. [8 points] Show the results of inserting 12, 10, 15, 17, 19, 14, 18 into an empty AVL tree. Show the tree after every insertion.

6. [10 points] Consider the following splay tree:

```
   10
  /  \
 11   12
    /  \
   13
```

   a. Show the result of accessing keys 5, 8, 7 in order in the splay tree. Show the tree after each access.
   b. Show the result of deleting keys 10, 8, 7 in the splay tree. Start with the original tree and show the tree after each deletion.

7. [8 points] Show the result of accessing keys 7, 5, 12 in the splay tree in problem 6 using top-down splay tree access.

8. [8 points] Find the values of $M$ and $L$ for a B-tree if the disk block size is 8192, a key uses 16 bytes, a link uses 4 bytes, and a data value uses 64 bytes. For 1,000,000 values, how many levels would need to be accessed in the worst case? How many levels would need to be accessed in the worst case for a binary search tree?

9. [10 points] Show the resulting B-tree, starting with the tree on slide 126 of the B-Trees slide set, after each of the following insertions: 107, 106, 10, 75, 77, 61, 64, 63, and 71.

10. [8 points] Show the red-black tree using top-down insertion of the following keys: 12, 10, 15, 17, 19, 14, 16, and 18. Show the tree after each insertion.

11. [8 points] Show the 2-3-4 tree after inserting the following keys: 12, 10, 15, 17, 19, 14, 16, 13, 22, 25, and 24. Show the tree after each insertion.

12. [8 points] Convert the following red-black trees into 2-3-4 trees (double circles are red):

   a. 
   b. 

```
``