Homework 2 (Due 3/21/18, 11:55PM)

100 total pts. **Please justify your answers with detailed explanations, correct answers with no explanations will receive 0 pts**. Submit all answers as one PDF file on Moodle. The name of your answer file should follow this format: FIRSTNAME_LASTNAME.pdf.

1. In class, we learned about comparison sorting algorithms implemented for arrays. Describe how the worst case time complexities of the following sorting algorithms change if they were implement for **doubly linked lists**? Assume that references to the middle of the linked list exist. Explain in detail, with big-O reductions.

A. (5 pts) Bubble sort?

B. (5 pts) Selection sort?

C. (5 pts) Insertion sort?

D. (5 pts) Merge sort? How would the space complexity change?

E. (5 pts) Quicksort? How would the space complexity change?

2. Lucky Charms is an arguably popular American breakfast cereal. It consists of expensive/delicious/delectable marshmallow pieces, and cheap/abhorrent oat pieces. In the year 2049, General Mills (the creator of Lucky Charms), is suspected of reducing the number of expensive marshmallow pieces to cut costs. As a newly hired Federal Fairness Federation investigator, you are tasked with the following:

A. (10 pts) Given M digital boxes of Lucky Charms (with each box represented as an array), and each box containing N total marshmallow and oats pieces, describe an algorithm which would determine the percentage of marshmallows present in O(M*N + log(M*N)) time. Assume that M and N are large. Do NOT reduce M and N to “n” in the big-O analysis.

B. (10 pts) Lucky Charms has 8 different marshmallow shapes. You have uncovered Lucky Charms’ secret marshmallow recipes, and have discovered that certain marshmallow shapes are more expensive to produce than other marshmallow shapes. Records from 2018 indicate that marshmallow shapes were distributed evenly and fairly into each box of cereal (i.e. each total count of each different marshmallow shape was relatively the same per box). You suspect that General Mills is no longer upholding this standard, and is distributing fewer expensive marshmallows, and more cheap marshmallows. Given the M digital boxes and N total pieces per box in Part A (initially unsorted), describe an algorithm which would determine whether General Mills is fairly distributing expensive and cheap marshmallows in O(M*N) time.

C. (10 pts) In the year 3000, Lucky Charms has become the #1 cereal in our Milky Way galaxy. To suit the palates of different species on various planets throughout the galaxy, General Mills executives have developed a new Lucky Charm piece. They claim that each piece produces a random flavor when chewed, supporting up to 1 quadrillion flavors. Given the tasting data of M digital boxes with N total pieces per box from Part A, how would future investigators experimentally determine the veracity of General Mills' claim of random flavors (from a set of 1 quadrillion flavors), without using statistical methods? Assume that computer memory is limited.

3. Draw the resulting AVL tree, if the following keys are inserted in order into an initially empty AVL tree:

A. (10 pts) 50, 28, 13, 68, 81, 7, 10, 93, 87, 100, 90, 39

B. (5 pts) How many of each rotation were used in Part A: single right, single left, double right, and double left?

4. Draw a binary tree which yields the following in-order and post-order traversals:

A. (10 pts) A binary tree which adheres to both of these traversals -

in-order: 92, 34, 86, 64, 20, 52, 74, 22, 51, 19, 63, 81

post-order: 34, 92, 20, 64, 86, 22, 74, 51, 63, 81, 19, 52

B. (5 pts) Write the pre-order traversal for the binary tree in PartA.

5. (15 pts) Write code (with comments) for the recursive method, sumBST(BSTnode currentnode). This method performs a post-order traversal of a BST of integers starting from the root, and finds the sum of all integers (`data`) in the BST.

Assume that the BST is never empty before calling sumBST().

No justification is necessary. Please submit your code (10 out of 15 pts) in writing/typed with the previous 4 questions, i.e. no .java file necessary. Provide comments (5 out of 15 pts) to explain what each part of your code is doing.

```
public class BSTnode {
     BSTnode left;
     BSTnode right;
     int data;
}
```

```
public class BST {
     BSTnode root;
     public int sumBST( BSTnode currentnode );
}

public static void main( String[] args ) {
     BST mytree = new BST();
     mytree.add( 15 );
     mytree.add( 9 );
     mytree.add( 60 );
     mytree.add( 53 );
     mytree.sumBST( mytree.root ); // performs (post-order) 9+53+60+15
}

public int sumBST( BSTnode currentnode ) {
     // YOUR CODE HERE









}
```