CS 1332 Exam 2 - SECTION A

Spring Semester: March 14, 2016

Name (print clearly including your first and last name): _____

Signature: _____

GT account username (gtg, gth, msmith3, etc): _____

CIRCLE your section: A1 A2 A3 A4 A5 A6

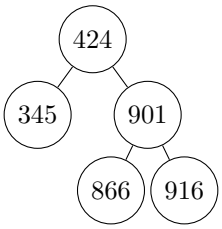
- Signing and/or taking this exam signifies you are aware of and in accordance with the **Academic Honor Code of Georgia Tech** and the **Georgia Tech Code of Conduct**.
- Write your name on every page.
- Notes, books, calculators, phones, laptops, smart watches, headphones, etc. are not allowed.
- Extra paper is not allowed. If you have exhausted all space on this test, talk with your instructor.
- Pens/pencils and erasers are allowed. Do not share.
- All code must be in Java.
- Efficiency matters. For example, if you code something that uses $O(n)$ time or worse when there is an obvious way to do it in $O(1)$ time, your solution may lose credit. If your code traverses the data 5 times when once would be sufficient, then this also is considered poor efficiency even though both are $O(n)$.
- Style standards such as (but not limited to) use of good variable names and proper indentation is always required. (Don't fret too much if your paper gets messy, use arrows or whatever it takes to make your answer clear when necessary.)
- Comments are not required unless a question explicitly asks for them.

Name: _____

[10] 1. The following requires you to draw a series of trees.

You are given the following AVL tree. First add 875 to the tree without performing any rebalancing. Second, redraw the AVL tree showing each individual rotation required to rebalance the AVL tree. If more than one rotation is required, show each intermediate tree.

For example, a double rotation requires two drawings: one for each of the two single rotations.

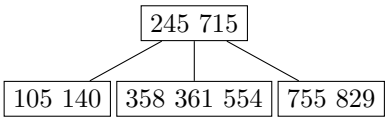


Name: _____

[15] 2. The following requires you to draw a series of trees.

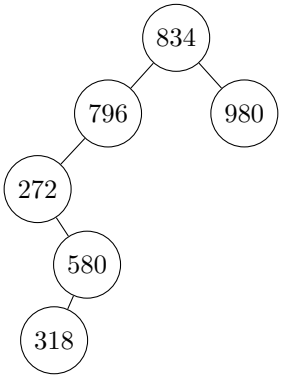
You are given the following 2-4 tree. First add 482 to the tree without performing any additional operations. Second, redraw the tree showing each individual rebalancing operation required. If more than one rebalancing operation is required, show each intermediate tree.

If necessary, use the 2nd data item in the node for promotions.



Name: _____

[5] 3. You are given the following splay tree. After adding 473, where will 473 be located? (You do not have to draw the final splay tree; just describe where 473 will be located)



Name: _____

[5] 4. How many items can be contained in a 2-4 tree node?

A. 1
B. 2
C. 3
D. 4
E. all of the above
F. A, B, and C only
G. B, C, and D only

5. Fill in the blank. Write the Big-O using proper notation. (For example, do not write just n, but rather $O(n)$.) In all cases, n refers to the number of data items in the data structure.

[5] (a) _____ What is the worst case Big O of searching a splay tree having n nodes?

[5] (b) _____ What is the worst case Big O of finding the minimum of an AVL tree having n data values?

[5] (c) _____ What is the worst case Big O of searching a hash table with n key-value pairs for a value if you do not know the key?

[5] (d) _____ You found a coin that has tails on both sides. Assuming that "heads" promotes the value to the next level, what is the Big O of adding a value to a SkipList of size n, using this coin in your CoinFlipper?

[5] (e) _____ What is the Big O for finding a value given a key in a hash table that uses external chaining with n key-value pairs? The keys have a good hash function and the backing array is reasonably sized.

Name: _____

6. You are given these details about a hash table that uses **external chaining**.

   Resolution strategy: external chaining

   Hash code function: use the number itself.

   Compression function: mod by the table size.

   Hash table size: 9. Do not resize the array.

[12] (a) Using each data item in the order given below, add the data item to the hash table.

`Data: 27, 48, 12, 25, 84, 20`

hashtable[0]

hashtable[1]

hashtable[2]

hashtable[3]

hashtable[4]

hashtable[5]

hashtable[6]

hashtable[7]

hashtable[8]

[3] (b) What is the load factor of the hash table in the previous question?

A. $\frac{6}{9}$

B. $\frac{4}{9}$

C. $\frac{3}{9}$

D. $\frac{2}{9}$

E. None of the above.

Name: _____

[5] 7. Given the max heap shown, remove the max and show the changes that result in the array and size. You only need to show the final state of the array and size. Skip slot 0.

```
size of max heap:  9
array[0]
array[1] 94
array[2] 67
array[3] 77
array[4] 47
array[5] 6
array[6] 75
array[7] 43
array[8] 26
array[9] 4
```

Name: _____

[20] 8. Complete the binary tree instance method `preorder()` recursively. The helper method header has been provided for you. Do not modify the method headers or add your own methods. Your method (and helper method) must efficiently return the preorder traversal of the binary tree. Include any import statements, if necessary. Your solution must use clean, proper recursion.

Left and/or right will be null when not referencing another node. Root will be null when the tree is empty.

Do not modify the code that is given which includes not modifying the header that is provided for the method you are writing.

An empty tree has 0 nodes. A tree having only the root node with no children has 1 node.

```
import java.util.List;


public class BinaryTree {
    private Node root;
    private class Node {
        public int data; // you can access this field directly
        public Node left, right;  // you can access these fields directly

        public Node(int data) {
            this.data = data;
            left = null;
            right = null;
        }
    }

    public List<Integer> preorder() {









    }

    private void preorder(Node current, List<Integer> list) {









    }
}
```

Name: _____

This page is blank beyond your Wildest Dreams.

Name: _____

This page is blank beyond your Wildest Dreams.