Name: Brian Sampson Date: 11/5/22

I pledge my honor that I have abided by the Stevens Honor System.

Point values are assigned for each question. Points earned: ____ / 100

1. Consider the algorithm on page 148 in the textbook for Binary Reflected Gray Codes. What change(s) would you make so that it generates the binary numbers **in order** for a given length `n`? Your algorithm must be recursive and keep the same structure as the one in the textbook. Describe only the change(s). (10 points)

Instead of copying list L1 to list L2 in reversed order, we should copy it in the same order to get the binary numbers in order for the given length n.

2. Show the steps to multiply 72 x 93 with Russian peasant multiplication, as seen in Figure 4.11b on page 154 in the textbook. (10 points)

We just add the m values that have a corresponding odd n value.

In this case that is:
9 : 744
1 : 5952

744 + 5952 = 6696 = 72 x 93

3. Suppose you use the LomutoPartition() function on page 159 in the textbook in your implementation of Quicksort. (10 points, 5 points each)
   a) Describe the types of input that cause Quicksort to perform its worst-case running time.

When the partition sizes are really unbalanced. For example, one segment has size n – 1 and the other segment has size zero.

   b) What is that worst-case running time?

   Worst case of lomutos partition is when it partitions n – 1 elements in the given array. We can say that this is $\theta(n)$ time.

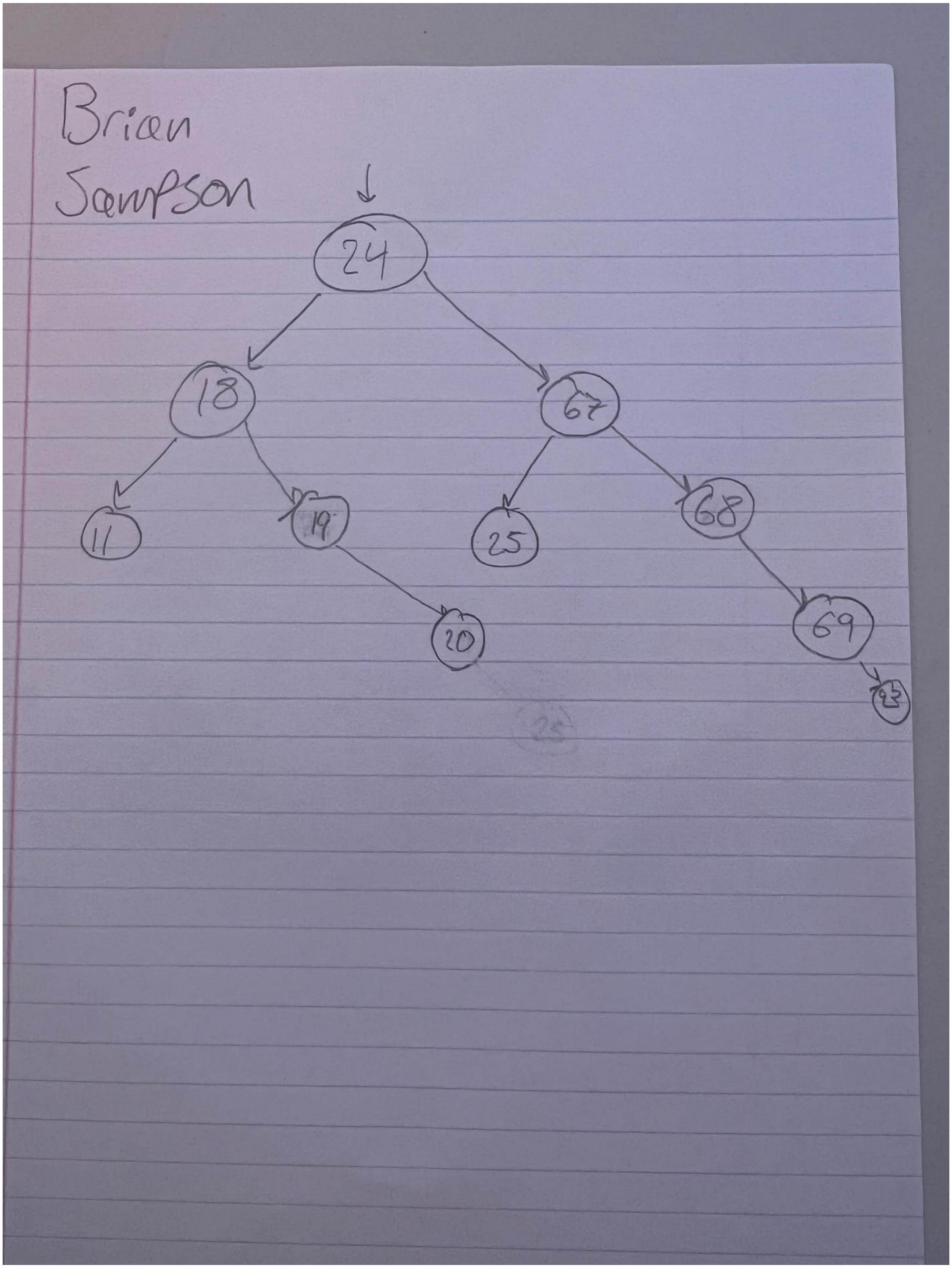
   We know that lomutos partition happens n times in the quicksort function. If lomutos partition is also n in its worst case, then we can say that **the total worst case is $\boldsymbol{\theta(n^2)}$ time**.

4. Compute 2205 x 1132 by applying the divide-and-conquer Karatsuba algorithm outlined in the text. Repeat the process until the numbers being multiplied are each 1 digit. For each multiplication, show the values of c_2, c_1, and c_0. Do not skip steps. (10 points)

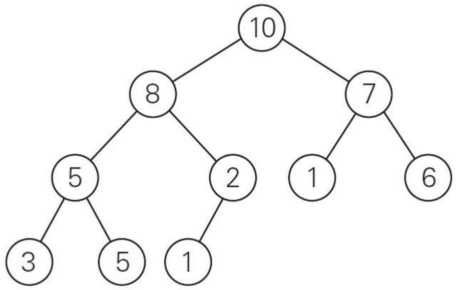
A = 22
B = 05

C = 11
D = 32
Step 1: A * C = 242 = C2
Step 2: B * D = 160 = C0
Step 3: (A+B)(C+D) = 1,161
Step 4: (Step 3) – (Step 2) – (Step 1) = 759 = C1
Step 5: 2,420,000 (Step 1 padded with 4 zeros) + 160 (Step 2 no padding) + 75,900 (Step 2 padded with 3 zeros) = 2,496,060

5. Draw the binary search tree after inserting the following keys: 24 18 67 68 69 25 19 20 11 93
   (10 points)



6. Consider the following binary tree. (16 points, 2 points each)



a) Traverse the tree preorder.
10, 8, 5, 3, 5, 2, 1, 7, 1, 6
b) Traverse the tree inorder.
3, 5, 5, 8, 1, 2, 10, 1, 7, 6
c) Traverse the tree postorder.
3, 5, 5, 1, 2, 8, 1, 6, 7, 10
d) How many internal nodes are there?
5
e) How many leaves are there?
5
f) What is the maximum width of the tree?
4
g) What is the height of the tree?
3
h) What is the diameter of the tree?
5

7. Use the Master Theorem to give tight asymptotic bounds for the following recurrences. (25 points, 5 points each)

a) $T(n) = 2T(n/4) + 1$
$\Theta(\sqrt{n})$

b) $T(n) = 2T(n/4) + \sqrt{n}$
$\Theta(\sqrt{n}\log_4(n))$

c) $T(n) = 2T(n/4) + n$
$\Theta(\mathrm{n})$
d) $T(n) = 2T(n/4) + n^2$
$\Theta(n^2)$
e) $T(n) = 2T(n/4) + n^3$
$\Theta(n^3)$

8. Consider the following function. (9 points)

```
int function(int n) {
    if(n <= 1) {
        return 0;
```

```
        }
        int temp = 0;
        for(int i = 1; i <= 6; i++) {
            temp += function(n / 3);
        }
        for(int i = 1; i <= n; i++) {
            for(int j = 1; j * j <= n; j++) {
                temp++;
            }
        }
        return temp;
    }
```

a) Write an expression for the runtime $T(n)$ for the function. (4 points)

T(n) = 6T(n/3) + $\Theta(n\sqrt{n})$

b) Use the Master Theorem to give a tight asymptotic bound. Simplify your answer as much as possible. (5 points)
$\Theta(n^{log_3 6})$