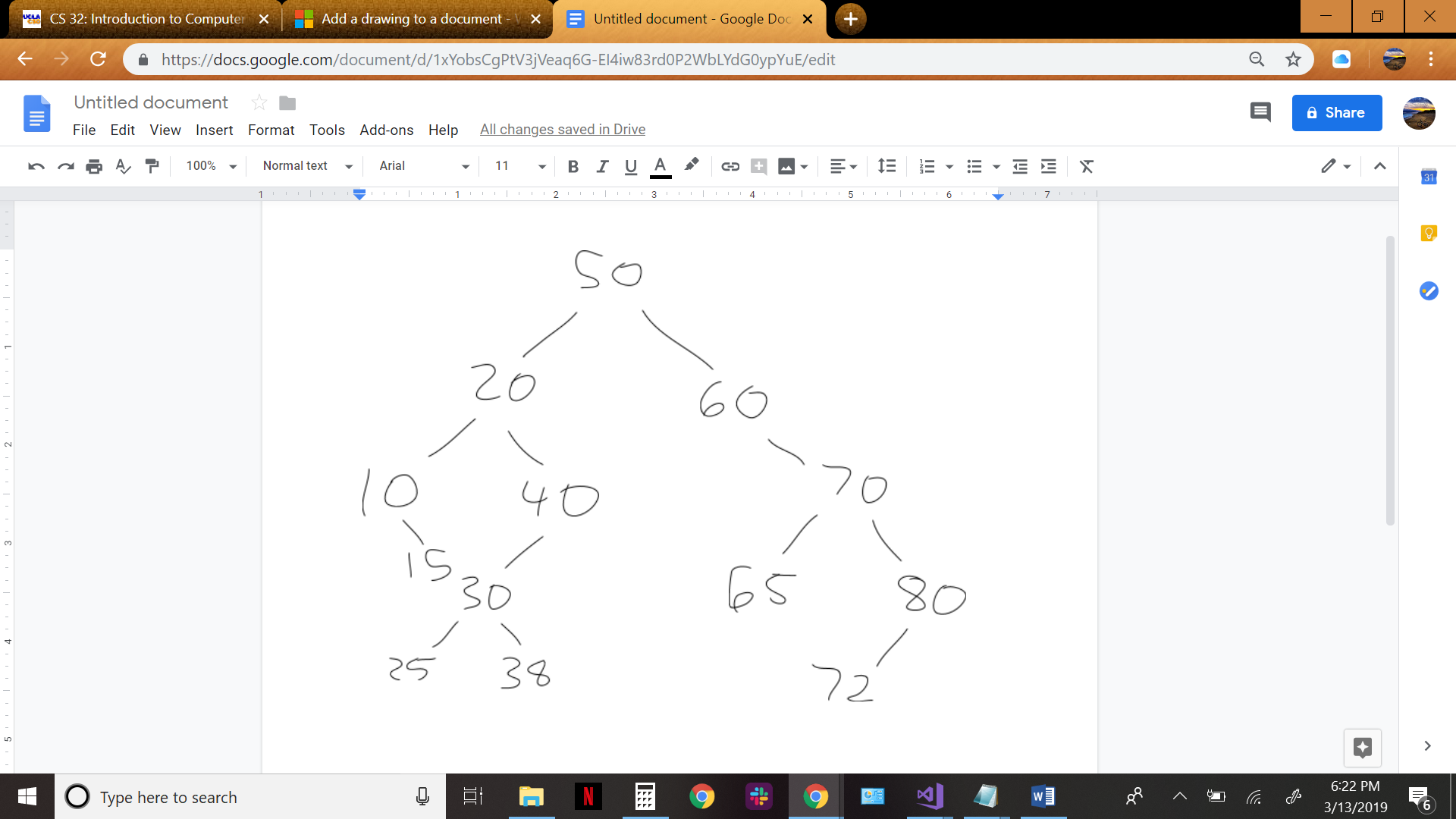
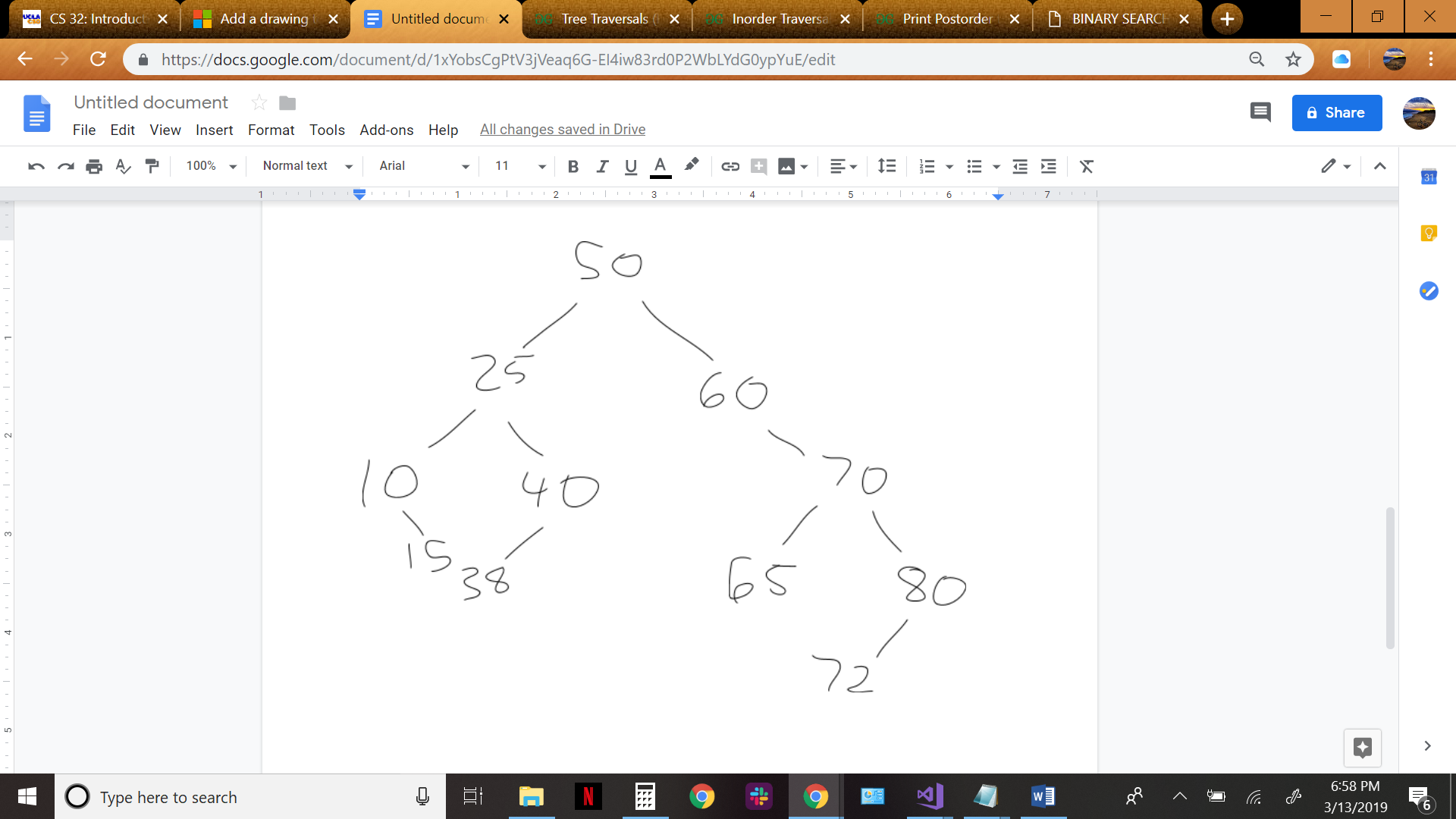
CS 32 Professor Smallberg Lecture 2 Leo Gretzinger

HW 5 Document 3/13/19

1. Binary Search Tree
   1. Adding 80, 65, 72, 15, 38 and 25 would result in:
   2. In-order: 10 15 20 25 30 38 40 50 65 70 72 80 Pre-order: 50 20 10 15 40 30 25 38 60 70 65 80 72 Post-order: 15 10 25 38 30 40 20 65 72 80 70 60 50
   3. 



1. a. struct TreeNode {

int data;

TreeNode\* left;

TreeNode\* right;

TreeNode\* parent;

};

b. Node\* insert(int value, TreeNode\* tnp)

if pointer is null, create new node with value

if value is greater than node’s right value

right node is returned from insert with value and node’s right

node’s right is right node returned above

right node’s parent is node

if value is less than node’s right value

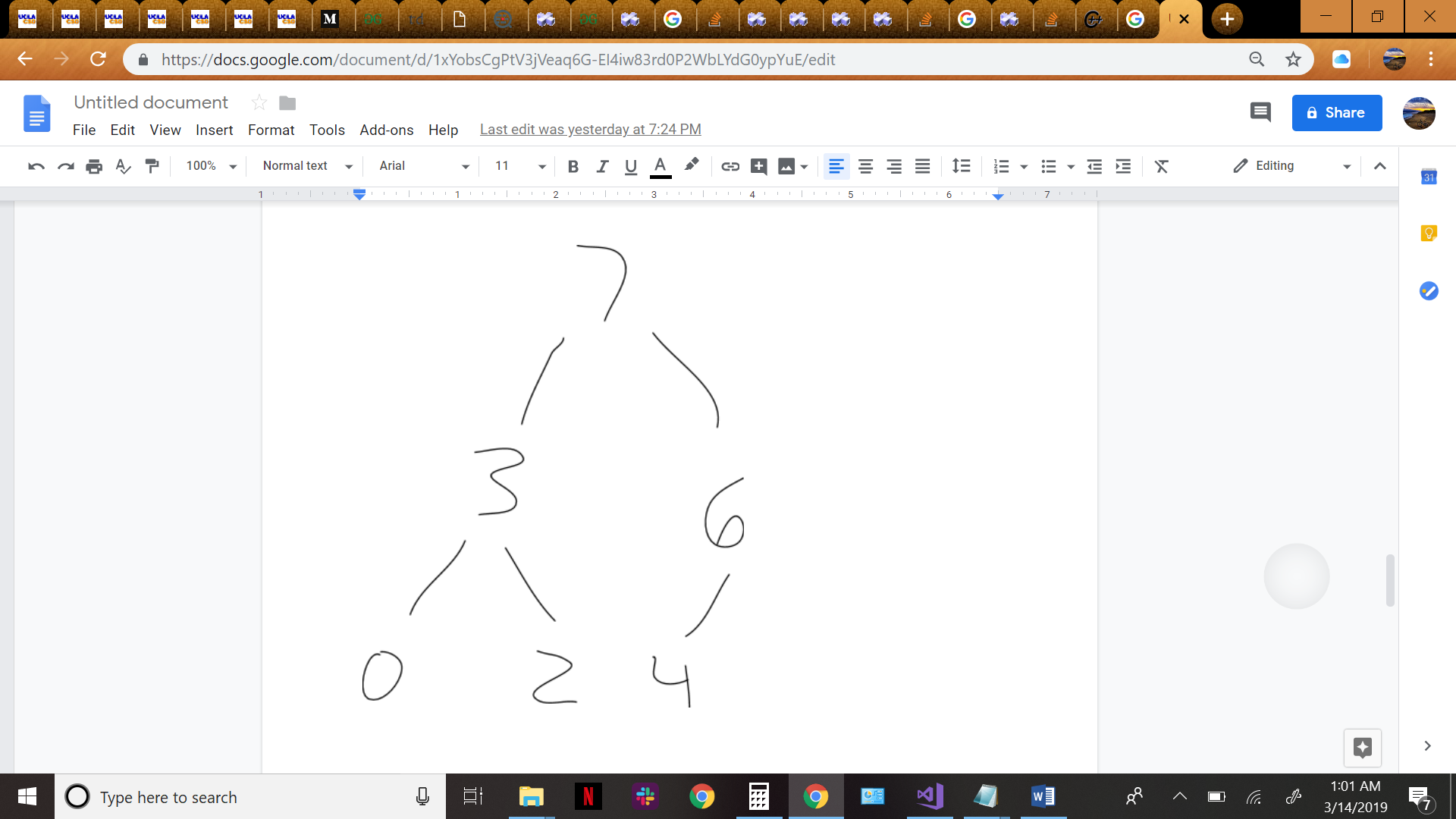
left node is returned from insert with value and node’s left

node’s left is left node returned above

left node’s parent is node

return the node

1. a.



b. Indecies: 0 1 2 3 4 5

Values: 7 3 6 0 2 4

c. Indecies: 0 1 2 3 4

Values: 6 3 4 0 2

1. a. O(C + logS)

b. O(logC + S)

c. O(logC + logS)

d. O(logS)

e. O(1)

f. O(logC + S)

g. O(S \* logS)

h. O(C \* logS)