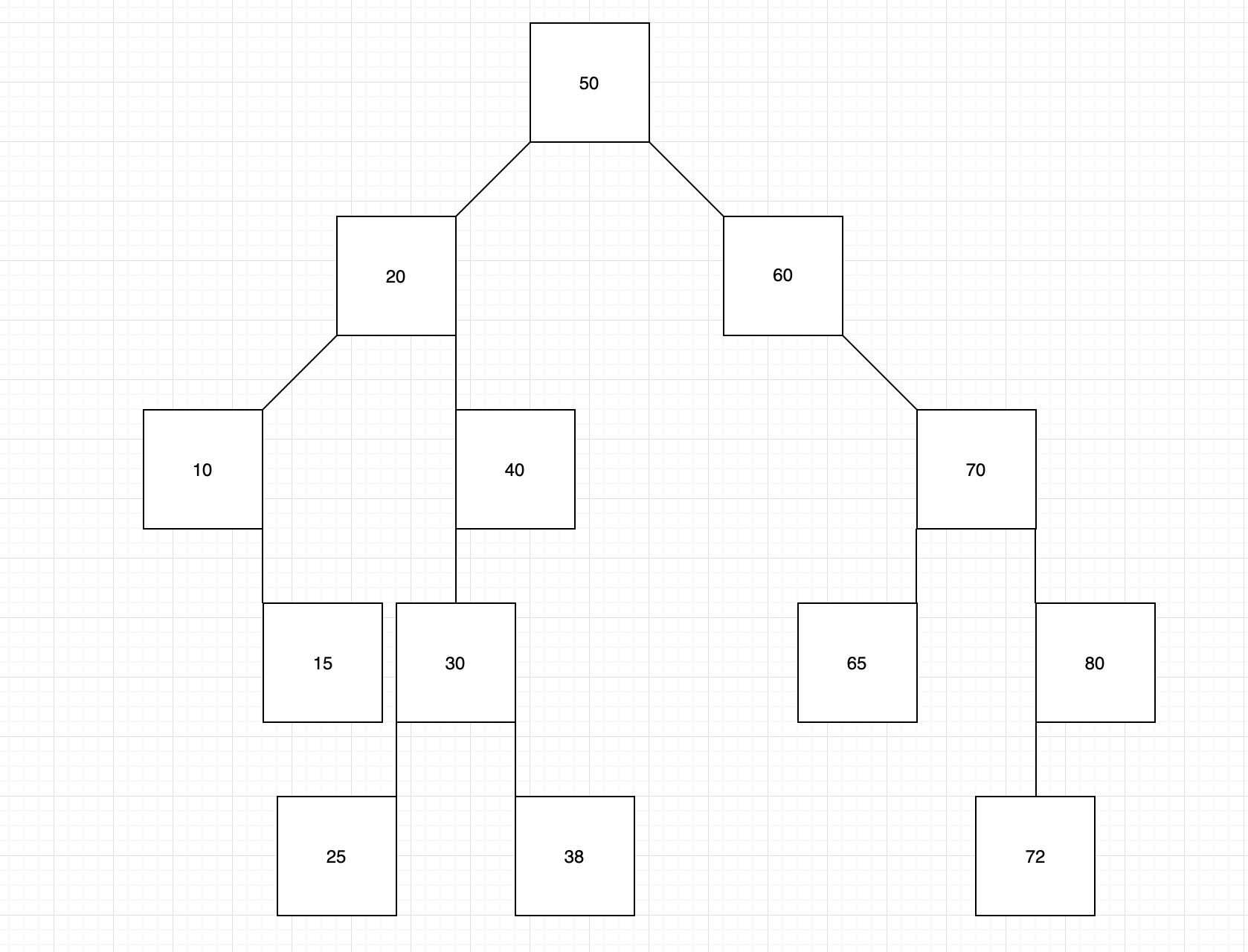
Nathan Tjoar

005081232

Homework

1a. 

1b. In-order printout:

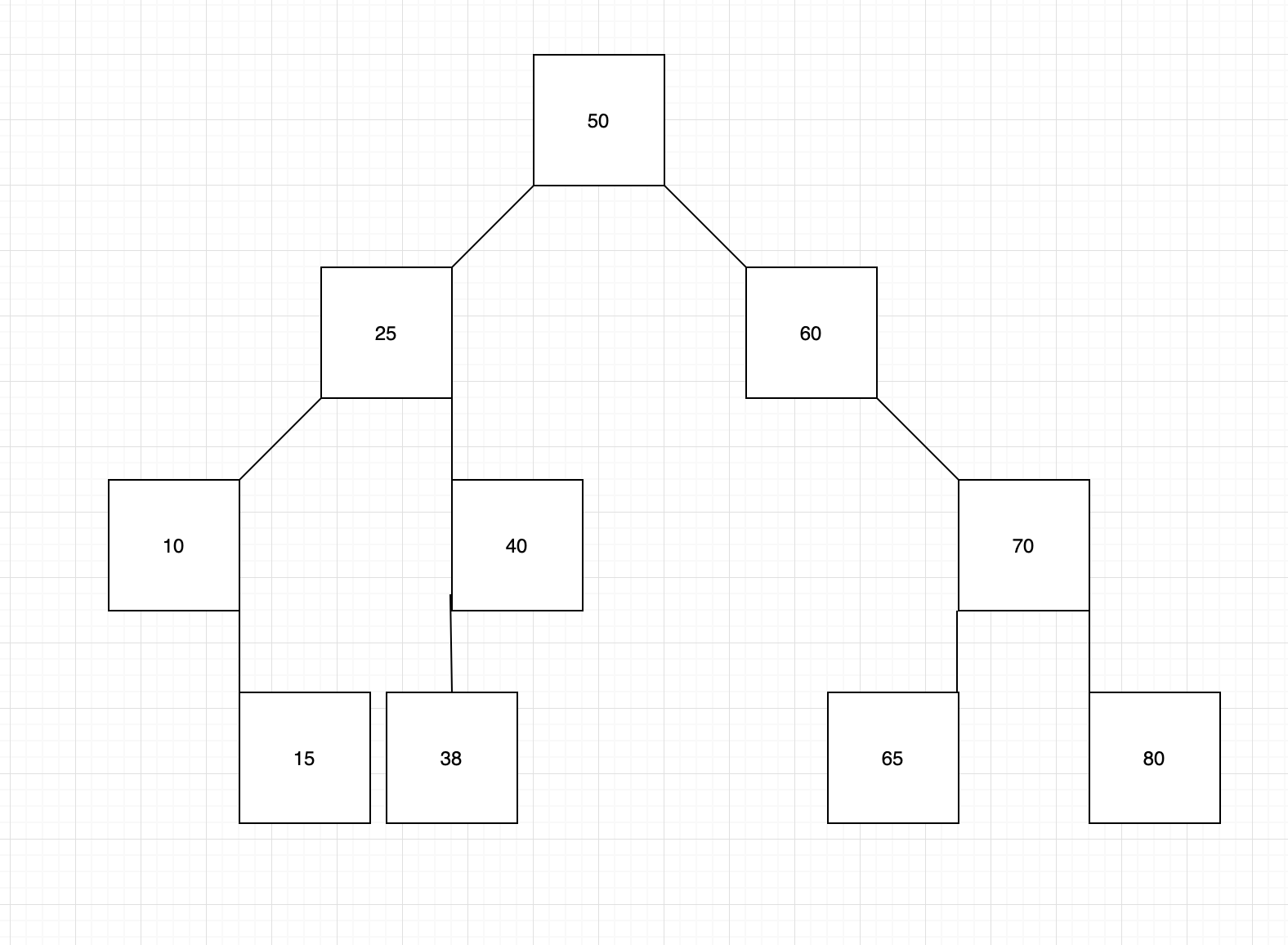
10, 15, 20, 25, 30, 38, 40, 50, 60, 65, 70, 72, 80

Pre-order printout:

50, 20, 10, 15, 40, 30, 25, 38, 60, 70, 65, 80, 72

Post-order printout:

15, 10, 25, 38, 30, 40, 20, 65, 72, 80, 70, 60, 50

1c. 

2a.

struct Node

{

Node(int key, Node\* parent)

{

m\_parent = parent;

less = nullptr;

more = nullptr;

data = key;

}

Node\* less;

Node\* more;

Node\* m\_parent;

int data;

};

Node\*root;

2b.

Node\* insert(Node\* node, int key)

{

if node is nullptr

add a new node

if key is less than our current node’s value

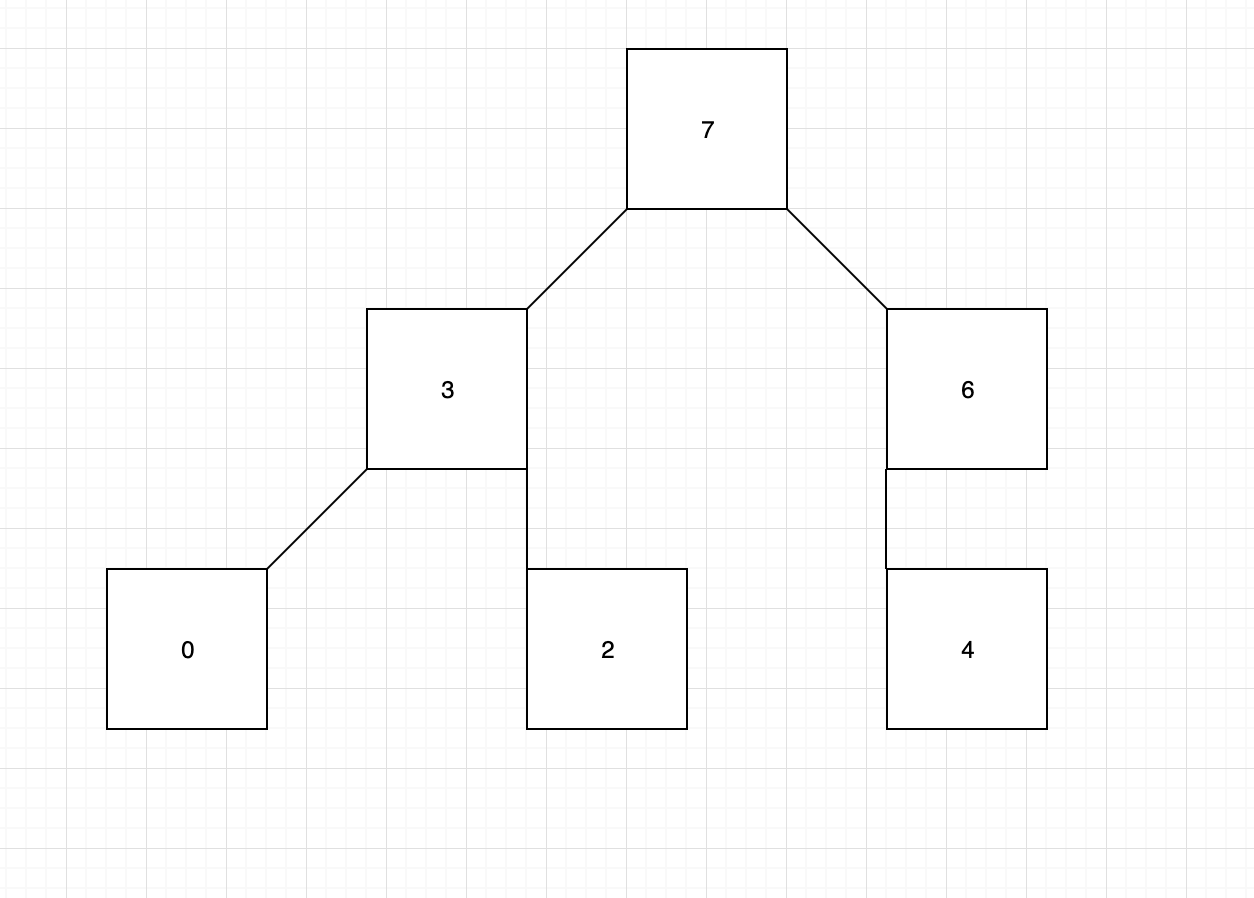
move left

else if key is greater than our current node’s value

move right

return node

}

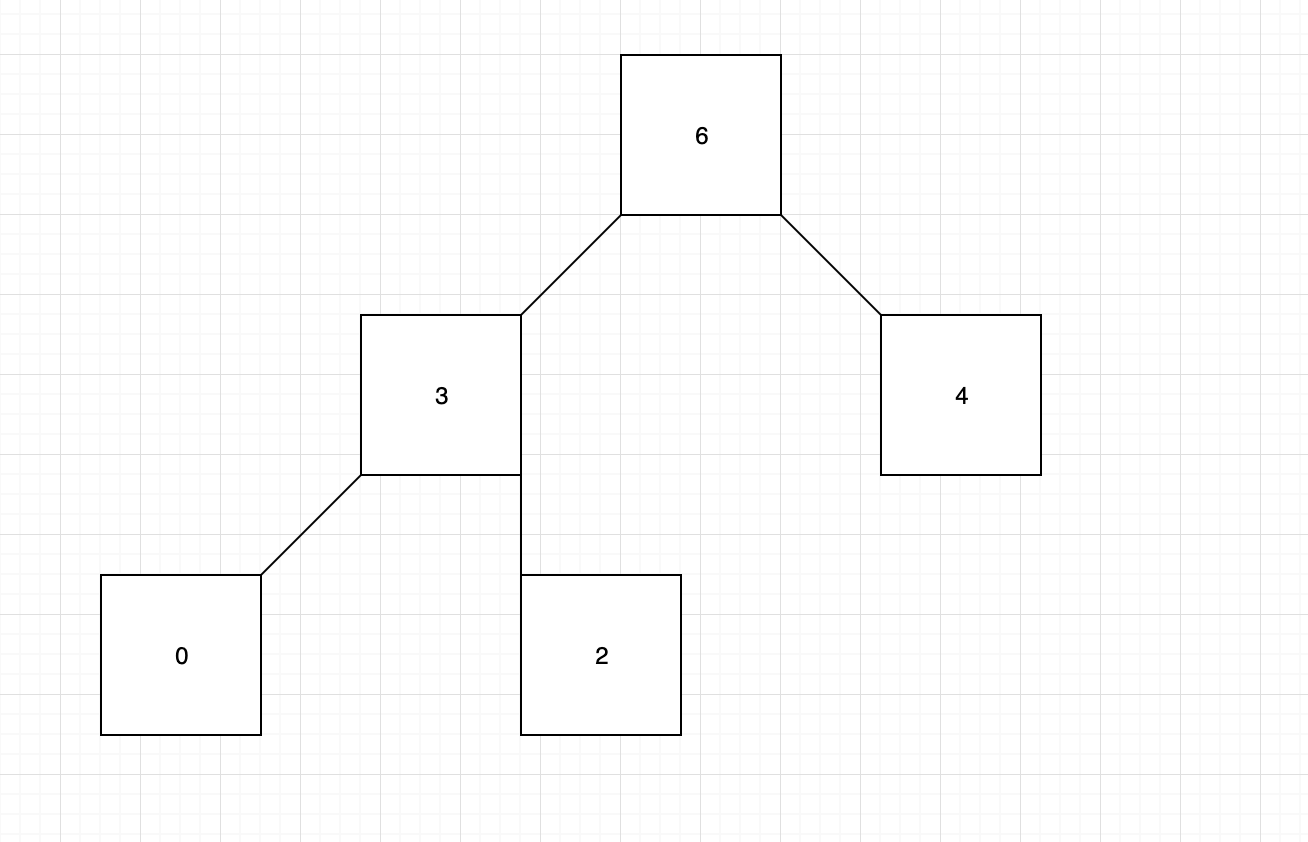
3a. 

3b. In an array, this would have:

Items: 7 | 3 | 6 | 0 | 2 | 4

Index: 0 | 1 | 2 | 3 | 4 | 5

3c.



In an array, this would have:

Items: 6 | 3 | 4 | 0 | 2

Index: 0 | 1 | 2 | 3 | 4

4a. O(C + S)

4b. O(S + log(C))

4c. O(log(S) + log(C))

4d. O(log(S))

4e. O(1)

4f. O(log(C) + S)

4g. O(S)

4h. O(C \* log S)