Homework 5

Joel George

1a. 50

20 60

10 40 70

15 30 65 80

25 35 75

b.

In: 10 15 20 25 30 35 40 50 60 65 70 75 80

Pre: 50 20 10 15 40 30 25 35 60 70 65 80 75

Post: 15 10 25 35 30 40 20 65 75 80 70 60 50

c. After deleting 30:

50

20 60

10 40 70

15 35 65 80

25 75

After deleting 20:

50

15 60

10 40 70

35 65 80

25 75

2a. struct Node

{

Node \*parent, \*left, \*right;

int value;

}

b.

create new Node

if tree is empty

set root pointer to new Node

set new Node parent and children pointers to nullptr

while we are not done…

if current Node’s value is equal to input value,

delete new Node (to avoid memory leak)

DONE

if current value is less than input value,

If there is a left child, go left

Else

make new Node’s parent pointer point to current Node, set child pointers to null

make current Node’s left pointer point to new Node

DONE

if current value is greater than input value,

If there is a right child, go right

Else

make new Node’s parent pointer point to current Node, set child pointers to null

make current Node’s right pointer point to new Node

DONE

3a. 8

3 6

0 2 4

b.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 8 | 3 | 6 | 0 | 2 | 4 |

c.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 6 | 3 | 4 | 0 | 2 |

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

b. O(log(C) + S)

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

d. O(C+log(S))

e. O(C+S)

f. O(log(C) + S)

g. O(C+Slog(S))

h. O(C\*log(S))