Daxuan Shu

204-853-061

Winter 2017

CS 32

Homework 5

1. a)

50

35

25

30

40

15

10

75

65

80

70

60

20

b) In-order: 10, 15, 20, 25, 30, 35, 40, 50, 60, 65, 70, 75, 80

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

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

c) Using left sub-tree largest value:

50

75

80

65

70

60

35

25

40

10

15

2.

a) struct node

{

int value;

node \*pLeft, \*pRight, \*Parent;

}

b)

If the tree is empty

Allocate a new node with \*Parent = nullptr and put the value into it.

point the root pointer to the new node Done.

Start at the root of the tree

while we are not done

If the value we want to add is equal to current node’s value, Done.

If the value is less than current node’s value

If there is a left child, then go left

Else allocate a new node with \*parent points to Current and put the value into it and set current node’s left pointer to the new node. Done

If the value is greater than the current node’s value

If there is a right child, then go right

Else allocate a new node with \*parent points to Current and put the value into it and set current node’s right pointer to the new node. Done

3. a)

8

4

6

2

0

3

b)

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

c)

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

4.

a) O(C + 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)