1a

50

20 60

10 40 70

15 30 65 80

25 36 74

1b)

In order

10, 15, 20, 25, 30, 36, 40, 50, 60, 65, 70, 74, 80

Pre order

50, 20, 10, 15, 40, 30, 25, 36, 60, 70, 65, 80, 74

Post order

15, 10, 25, 36, 30, 40, 20, 65, 74, 80, 70, 60, 50

1c)

Deleting “30”

50

20 60

10 40 70

15 25 65 80

36 74

Deleting “20”

50

15 60

10 40 70

25 65 80

36 74

2a)

struct Node

{

Node\* left;

Node\* right;

Node\* parent;

int value;

};

2b

insertNode( Node\* n, Node\* root)

{

if root is null

root equals n, set the parent pointer of n to null

if the value of n equals the value of root

return

if the value of n is less than that of root

if root has a left child

call insertNode with roots left child as root

else

set roots left child to n

if the value of n is greater than that of root

if root has a right child

call insertNode with root’s right child as root

else

set roots right child to n

}

3a)

8

3 6

0 2 4

3b)

8,3,6,0,2,4

3c)

6,3,4,0,2

4a)

O(C+S)

4b

O(S+ logC)

4c)

O(logC + logS)

4d)

O(logS)

4e)

O(1)

4f)

O(logC+S)

4g)

O(SlogS)

4h)

O(ClogS)