1a.

50

25

30

20

40

15

10

60

75

70

650

80

35

1b. 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

1c.

50

35

25

40

15

10

60

75

70

650

80

2a. struct Node

{

Node\* left;

Node\* right;

Node\* parent;

int value;

};

2b. void insert(Node\*& root, Node\*& newNode) //newNode’s pointers all nullptr

{

if the root is nullptr

set the root to newNode

otherwise

set newNode’s parent to root

if the newNode’s key is greater than the root’s

insert to the root’s right pointer

otherwise //<=

insert to the root’s left pointer

}

3a.

8

3

2

0

6

4

3b. Array representation: [8, 3, 6, 0, 2, 4]

3c. [6, 3, 4, 0, 2]

4a. O(C + S)

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

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

4d. O(log(S))

4e. O(1)

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

4g. O(S\*log(S))

4h. O(C\*log(S))