1a.

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.

2a.

struct Node

{

value int;

Node\* left;

Node\* right;

Node\* parent;

}

2b. Insert function:

create a current and previous pointer

until a nullptr is reached

if the value is less than value at the node

set the previous pointer to the current pointer

set the current pointer to the left child

if the value is greater than the current value

set the previous pointer to the current pointer

set the pointer to the left child

create a new node at the current position

set the child pointers to nullptr

set the parent pointer to previous

3a.

3b. element: 0 1 2 3 4 5

value: 8 3 6 0 2 4

3c. element: 0 1 2 3 4

value: 6 3 4 0 2

4a. O(C + S)

4b. O(logC +S)

4c. O(logC + logS)

4d. O(logS)

4e. O(1)

4f. O(logC +S)

4g. O(C\*SlogS)

4h. O(C\*logS)