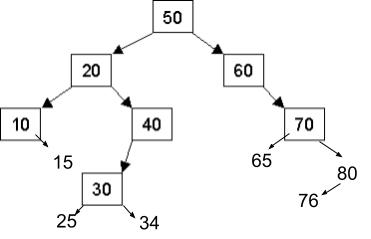
Jack Corddry

1.

A.



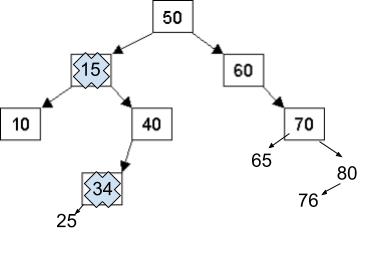
B.

In order: 10 15 20 25 30 34 40 50 60 65 70 76 80

Post order: 15 10 25 34 30 40 20 65 76 80 70 60 50

Pre order: 50 20 10 15 40 30 25 34 60 70 65 80 76

C.



2.

A.

struct Node {

int data;

Node\* parent, left, right;

};

B.

void insert(Node addme, Node\* head) {

If addme’s value is the same as head’s value

Return

Else

If addme’s value is greater than head’s

If head’s right pointer is nullptr

Link up head and addme using head’s right pointer and addme’s

parent pointer

else

Recursively call insert with the right pointer as head

Else

If head’s left pointer is nullptr

Link up head and addme using head’s left pointer and addme’s

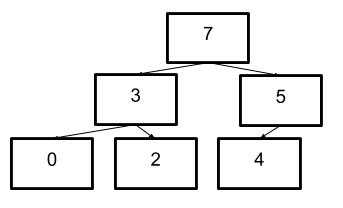
parent pointer

else

Recursively call insert with the left pointer as head

3.

A.



B.

{7, 3, 5, 0, 2, 4}

C.

{5, 3, 4, 0, 2}

4.

1. O(C+S)
2. O(logC+S)
3. O(logC+LogS)
4. O(LogS)
5. O(1)
6. O(logC+S)
7. O(S^2)
8. O(C\*logS)

5.

B.

Without the string parameter, there would be no way to concatenate the base classes onto each of the subclasses without using an additional container. The string “path” must be used to add the parent=> to each subclass.