Problem Domain:

Create a breath first traversal method that takes in a binary tree and output a list of the values in the tree in the order they were encountered

Edge Cases:

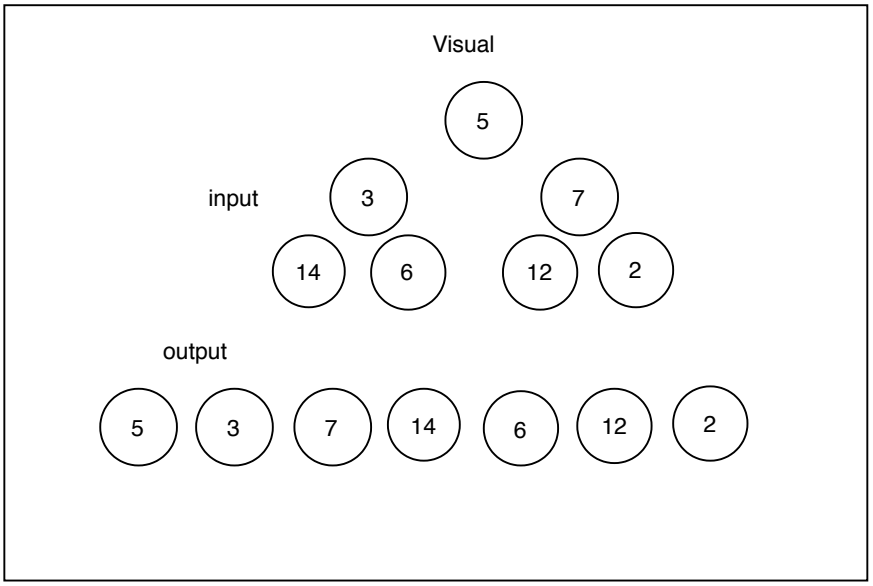
- if node is null
- if there is only one thing in the tree
- if the tree is not balanced

Big O:
Time- O(n)
Space- O(n*2)

Algorithm:

- create a new queue
- create a list
- enqueue the root from the given tree
- enqueue the root
- while queue is not empty, dequeue current node and add that value to the list
- enqueue the left of that current node, and then enqueue the right
-return list

Visual




Pseudo:

BreadthFirstTraversal(tree)
create empty Queue queue
create empty List list
root.enqueue
while tree !isEmpty....
node front = dequeue
list.Add front.value
if front.left !null, enqueue front.left
if front.right !null enqueue front.right

once front is null, return list