Lab-5

problem 1: Implement minstack. write down your idea and your logic for concluding the operation O(1).

we : implemented a stack using LinkedList with the primitive operations and another class MinStack that will do all the stack does plus computing the min.

By defining a stack that will keep the min on top of the stack, this will help us when we pop, we don't loose track of the next min value in the stack.

we add to the stack when value is pushed to our Minstack stack.

Running Time:

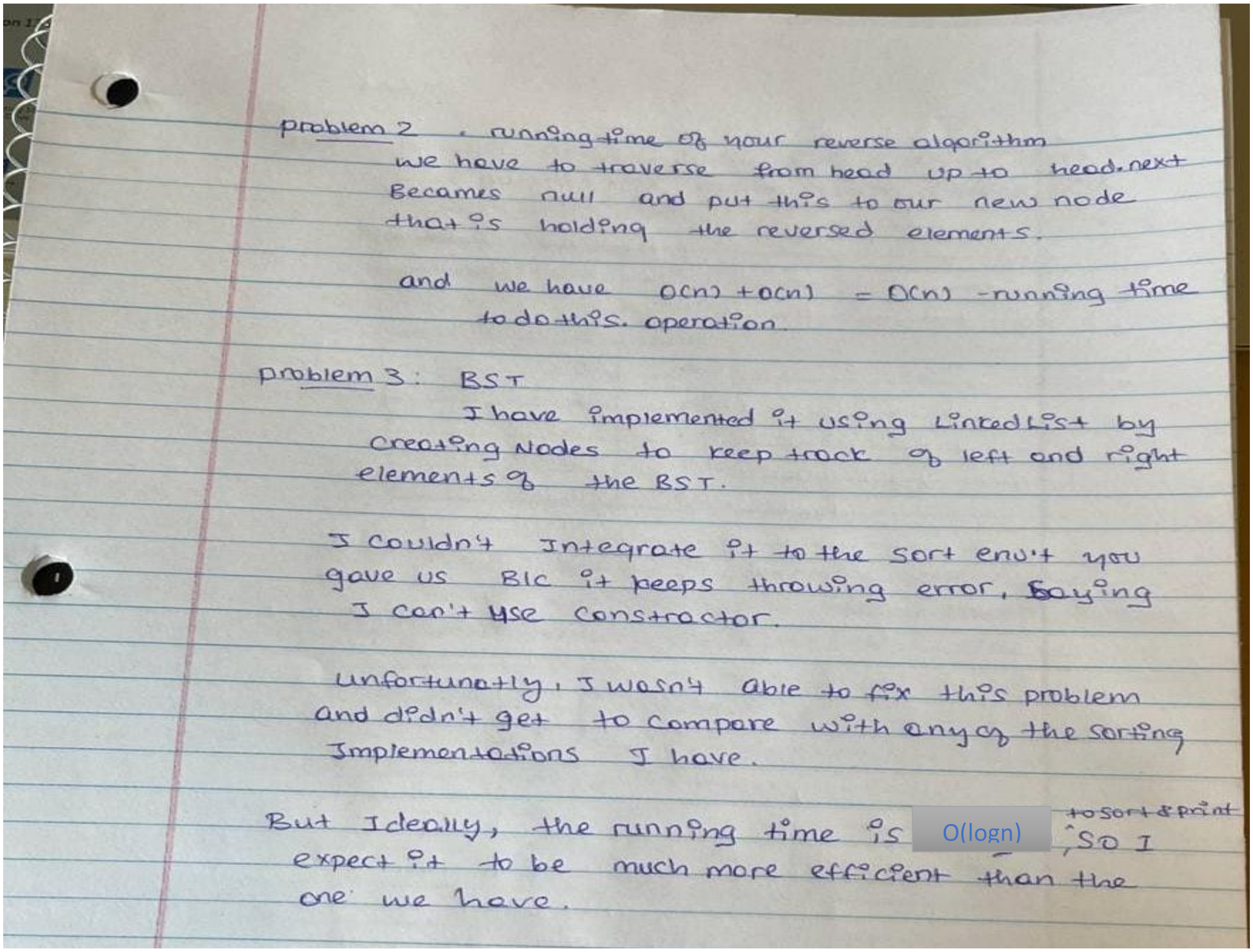
The running time is O(1) constant time, Bcause we are not adding operation that are costly the operation we add are

```
push(val)
add(val) .........  + 1
top ← top+1 ......  + 2
If(min > val) then  + 1
    min ← val ....  + 1
keepMin ← ~~min~~ push(min) ... + 2
                              ‾‾‾
                               C
```

same thing happens when pop operation is called as well. so it's still constant amount of work done at each step

∴ $\Theta(1)$ → running time for all operation

Note: I have min() method that returns the minimum value By reading from the top of the keepmin stack.



problem 2 . running time of your reverse algorithm
we have to traverse from head up to head.next
Becames null and put this to our new node
that is holding the reversed elements.

and we have O(n) + O(n) = O(n) - running time
to do this. operation.

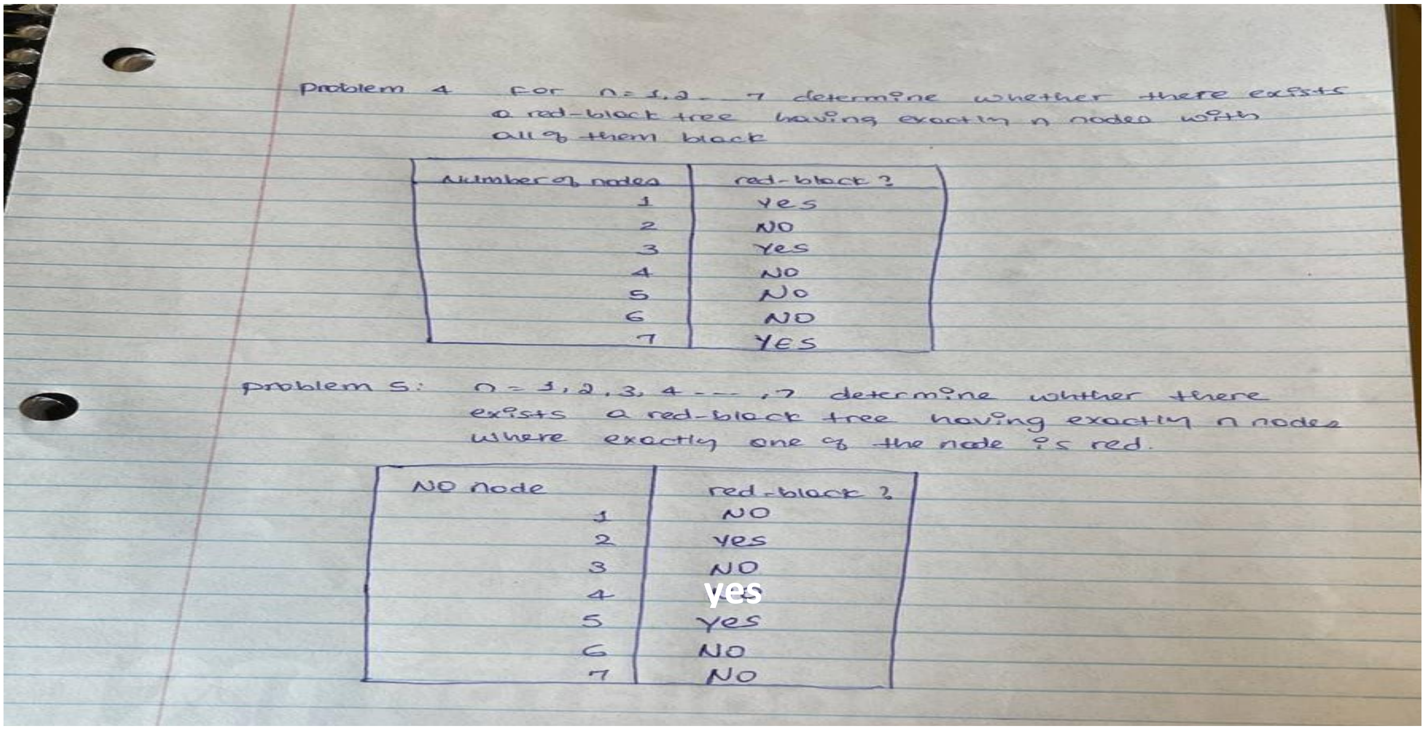
problem 3: BST

I have implemented it using LinkedList by creating Nodes to keep track of left and right elements of the BST.

I couldn't Integrate it to the sort env't you gave us B/c it keeps throwing error, ~~s~~ saying I can't use constractor.

unfortunatly, I wasn't able to fix this problem and didn't get to compare with any of the sorting Implementations I have.

But Ideally, the running time is O(logn) to sort & print, so I expect it to be much more efficient than the one we have.



problem 4 For n=1,2 _ _ 7 determine whether there exists a red-black tree having exactly n nodes with all of them black

Number of nodes	red-black ?
1	Yes
2	NO
3	Yes
4	NO
5	No
6	NO
7	YES

problem 5: n = 1, 2, 3, 4 - - - , 7 determine whther there exists a red-black tree having exactly n nodes where exactly one of the node is red.

NO node	red-black ?
1	NO
2	yes
3	NO
4	yes
5	yes
6	NO
7	NO