**Time Complexity of Iterative**

Time Complexity**:** O(n)

Iterate trough the linked list. In loop, change next to prev, prev to current and current to next.

Time Complexity of a loop is considered as O(n) if the loop variables is incremented / decremented by a constant amount.

**Time Complexity of Recursive**

Time Complexity: O(n)

1. Divide the list in two parts - first node and rest of the linked list.

2. Call reverse for the rest of the linked list.

3. Link rest to first.

4. Fix head pointer

Factorial

N\*fact(n-1) t(n) = c+ t(n-1) Problem Reduced from Size N to N-1

Backwards substitution: t(n) = c + c + t (n-2)

= c + c + c +t(n-3)

…= nc + t(n-n)

= nc+t(0)[Base Case (Constant time)]

= nc + Co

So Factorial is O(n)