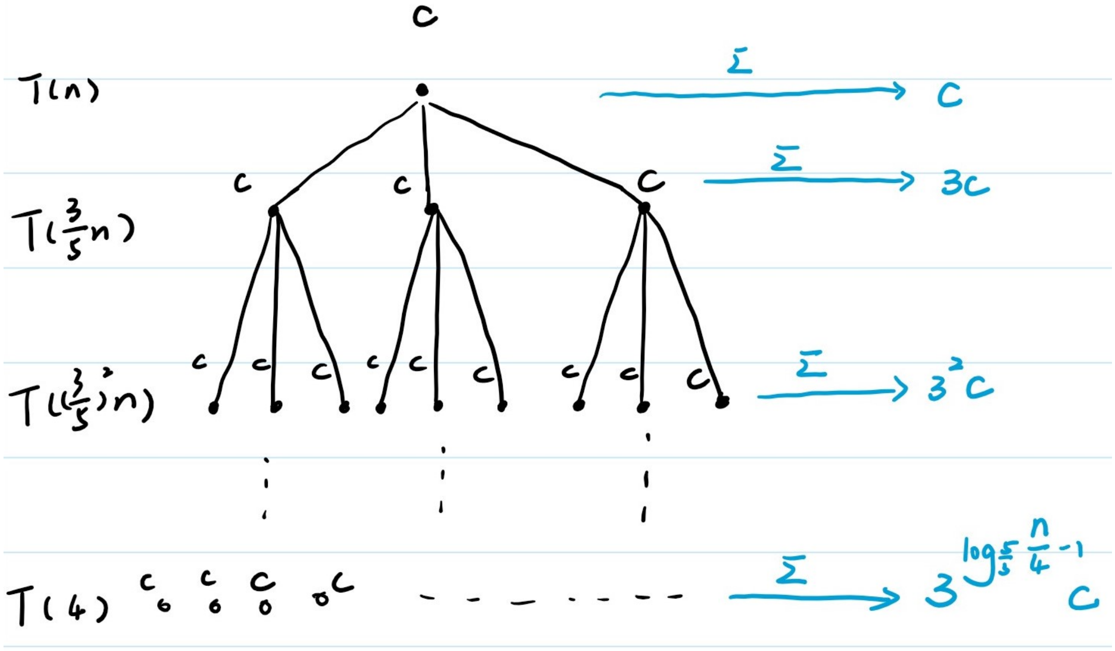
Problem 3: Recurrences

a) We know that $T(n) \in O(1)$ when $n < 5$ since this case, we apply the bubble sort with at most 4 elements.

Otherwise, $T(n) = T\left(\frac{3}{5}n\right) + T\left(\frac{3}{5}n\right) + T\left(\frac{3}{5}n\right) + c = 3T\left(\frac{3}{5}n\right) + c$

Because in each recurrence, it recursively call itself with array length $\frac{3}{5}n$

b) we can draw a recursion tree.



therefore, the total running time is

$$c \cdot \sum_{i=0}^{\log_{\frac{5}{3}}\frac{n}{4}-1} 3^i = \Theta\left(3^{\log_{\frac{5}{3}}\frac{n}{4}}\right) = \Theta\left(3^{\log_{\frac{5}{3}} n}\right) = \Theta(n^{\log_{\frac{5}{3}} 3})$$

c) By Master Theorem, we have a = 3, b = 5/3, k = 0.
And the case here satisfied should be $3 = a > b^k = 1$

Therefore, the running time should be $\Theta(n^{\log_{\frac{5}{3}} 3})$

d) Example: {4, 3, 5, 2, 1}
The first recursive call turns it into {3, 4, 5, 2, 1}
The second recursive call turns it into {3, 4, 1, 2, 5}
The last recursive call turns it into {1, 3, 4, 2, 5} which is not a sorted array.