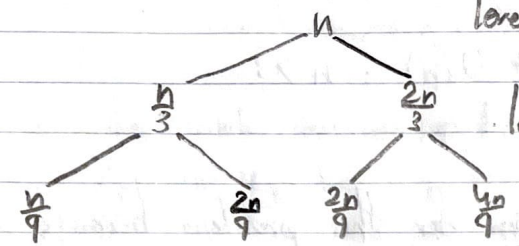
2.)

I will solve my recurrence relation from part 1 by using the tree method.

The tree



level 0 = n

level 1 = n

level 2 = n

Work done per level: n

I will now find k where it will represent the amount of times it will take us to reach the base case:
I will compare $n/3$ and $2n/3$ to see which k value will terminate last:

$T(\frac{n}{3}) = \frac{n}{3^{k_1}} \leq 3$

$n \leq 3(3^{k_1})$

$n \leq 3^{k_1+1}$

$\log_3 n \leq \log_3(3^{k_1+1})$

$\log_3 n < k_1+1$

$k_1 > \log_3 n - 1$

$T(\frac{2n}{3}) = (\frac{2}{3})^{k_2} n \leq 3$

$(\frac{2}{3})^{k_2} \leq \frac{3}{n}$

$\log_{\frac{2}{3}}(\frac{2}{3})^{k_2} \leq \log_{\frac{2}{3}} \frac{3}{n}$

$k_2 \leq \log_{\frac{2}{3}} \frac{3}{n}$

This value will be greater than k_1