Name: Daniel Kim

ID: 102353420

CSCI 3104, Algorithms | **Profs. Chen & Grochow**

Requiz S7 Version A | **Spring 2020, CU-Boulder**

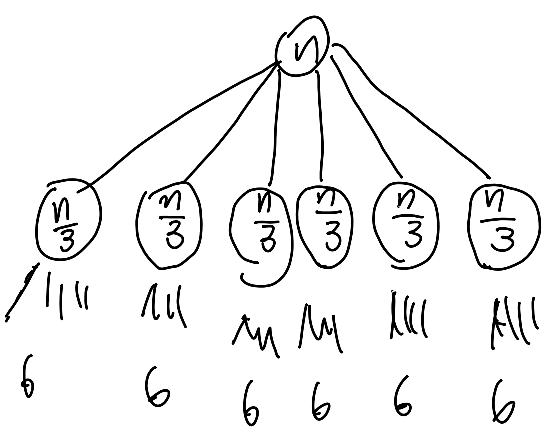
Instructions: This quiz is open book and open note. You **may** post clarification questions to Piazza, with the understanding that you may not receive an answer in time and posting does count towards your time limit (30 min for 1x, 37.5 min for 1.5x, 45 min for 2x). Questions posted to Piazza **must be posted as PRIVATE QUESTIONS.** Other use of the internet, including searching for answers or posting to sites like Chegg, is strictly prohibited and will count as violations of the academic honor code. Such violations are, at a minimum, grounds to receive a 0. Proofs should be written in **complete sentences. Show and justify all work to receive full credit.**

Standard 7. *Using the tree method, solve the following recurrence relation.*

We know that: $T(n) = \sum_{i=0}^{\text{num-levels}} \text{cost at level } i$

$$T(n) = \begin{cases} 1 & : n < 3, \\ 6T(n/3) + n & : n \geq 3. \end{cases}$$

There are $\boxed{\log_3(n)}$ levels of the tree since we are dealing with sub-problems of size $\frac{n}{3}$ in each recursive call.



Work

n

$6 \cdot \left(\frac{n}{3}\right) = \frac{6n}{3} = 2n$

$6 \cdot 6 \cdot \frac{n}{9} = 4n$

$\therefore$ at level i, is $2^i \cdot n$

Next, add up the nonrecursive work at each level

$T(n) = \sum_{i=0}^{\log_3(n)} 2^i \cdot n$

$= n \sum_{i=0}^{\log_3(n)} 2^i$

$= n \cdot \frac{1 - 2^{\log_3(n)} \cdot 2}{1 - 2}$

$= n \cdot -(1 - 2^{\log_3(n)+2})$

$= n \cdot (2^{\log_3(n)+2} - 1)$

Therefore, the above expression is cn which gives us an overall result of $\Theta(n)$

1