Prove that $\log n! \in \Theta(n \log n)$

1. $\log(n!) = \log(n(n-1)\ldots 1) = \log(n) + \log(n-1) + \ldots \log(1)$

$\log(n!)$

$\log(n) + \log(n-1) \ldots \log(1)$

$>= \log(n/2) \ldots \log(n/2)$

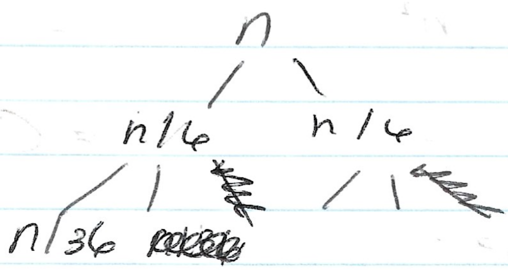
$n/2 \times (\log(n) - 1)$

$\lim_{n \to \infty} \dfrac{n/2}{n/2 \cdot \log n} = \Theta$

so therefore $\log(n!) \in \Theta(n \log n)$

2.

a) $T(n) = 2T(n/6) + 1$



1

2

4 ..... 2^i

$$\sum_{i=0}^{\log_6 2} 2^i = 2^{\log_6 n + 1} - 1$$

$= n^{\log_6 2}$

$= \Theta(n^{\log_6 2})$

b) $T(n) = 6T(n/4) + n$

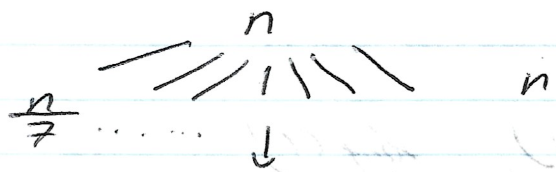
$\text{Cost} = 6^i \cdot n/4^i$

$$\sum_{i=0}^{\log_4 n} n \cdot (6/4)^i = n\left(\frac{(6/4)^{\log_4 n + 1} - 1}{6/4 - 1}\right)$$

$= \Theta(n^{\log_4 6})$

c. $T(n) = 7T(n/7) + n$

Cost: $n \cdot 7^i / 7^i = n$

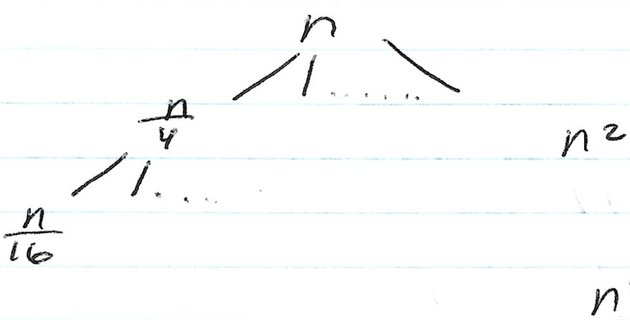


$$\sum_{i=0}^{\log_7 n} n = n(\log_7 n + 1)$$

$$= O(n \log n)$$

d) $T(n) = 9T(n/4) + n^2$

Cost: $9^i (n^2/4^i)$

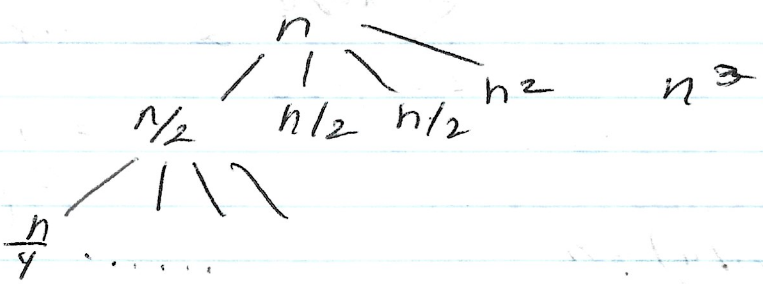


$$\sum_{i=0}^{\log_4 n} n^2 (9/4)^i$$

$$= n^2 \left(\frac{(9/4)^{\log_4 n + 1} - 1}{9/4 - 1} \right)$$

$= O(n^2)$ ← constant

e. $T(n) = 4T(n/2) + n^3$



Cost: $4^i (n^3/2^i)$

$$\sum_{i=0}^{\log_2 n} n^3 (2^i)$$

$O(n^3 \cdot 2^{\log_2 n})$

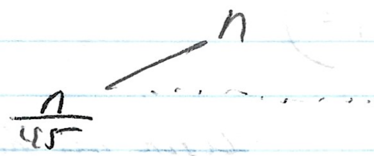
$O(n^4)$

$\Theta(n^3)$

f. $T(n) = 49(n/25) + n^{3/2} \log n$

cost: $49^i (n^{3/2} \log n / 25^i)$

$= n^{3/2} \log n \cdot (49/25)^i$



$$\sum_{i=0}^{\log_{25} n} n^{3/2} \log n \, (49/25)^i$$

$= O(n^{3/2} \log(n))$

g. $T(n) = T(n-1) + 2$ cost: $2n$

$= O(n)$

h. $T(n) = T(n-1) + n^c$ w/ $c \geq 1$

cost $= (n-1)^c$

$$\sum_{i=0}^{\cancel{\log}\, n-1} (n-i)^c$$

$= O(n^{c+1})$

i. $T(n) = T(\sqrt{n}) + 1$

~~[illegible]~~ $n^{1/2^i}$ w/ cost of 1

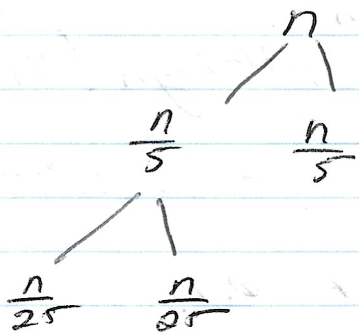
# of levels required is

$\log \log n$

$O(\log \log n)$

3.

a) $T(n) = 2T(n/5) + O(n^2)$



n^2

Work per node = $(n/5^i)^2$

$2^i \left(\frac{n}{5^i}\right)^2 = \frac{n^2}{5^{2i}} \cdot 2^i$

$$\sum_{i=0}^{\log_5 n} O\left(\frac{n^2}{5^{2i}} \cdot 2^i\right)$$

$\frac{2^i}{25^i}$ as it goes to ∞ it converges

$W = O(n^2)$

$S = O(n^2)$

b) $T(n) = T(n-1) + O(\log n)$

$\log n$

$$\sum_{i=0}^{n-1} O(\log(n-1))$$

$W = O(n \log n)$

and since there is only 1 node per level, the span would be

$S = O(\log n)$

c) $T(n) = T(n/3) + T(2n/3) + O(n^{1.1})$

size of sub problem: $n/3^i$ and $2n/3^i$

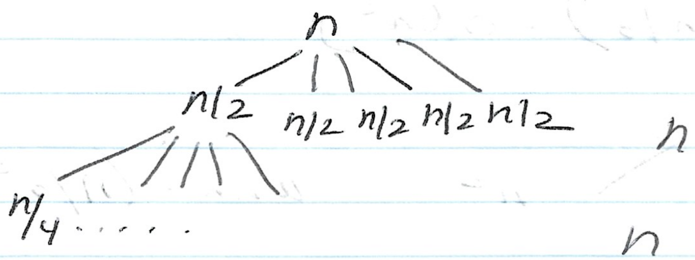
$$\sum_{i=0}^{\log_3 n} O(n^{1.1}) = O(n^{1.1})$$

work is $O(n^{1.1})$

W and $S = O(n^{1.1})$

4.

a) $T(n) = 5T(n/2) + O(n)$



Cost: $n/2^i$

work $= 5^i(n)$

$S = n(\log_2 n)$

$= O(n \log n$

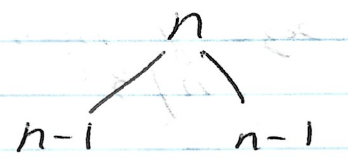
$$W = \sum_{i=0}^{\log_2 n} O(n \cdot 5^i)$$

$$= n\left(\frac{5^{\log n+1} - 1}{5-1}\right)$$

$= O(n \cdot (n^{\log_2 5})$

$W = O(n^{\log_2 5})$

b) $T(n) = 2T(n-1) + O(1)$

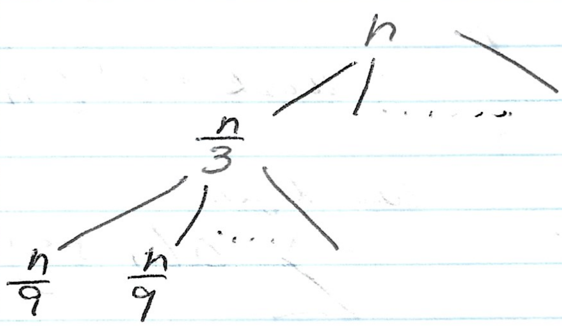


1

$$\sum_{i=0}^{n-1} O(1) = n$$

$w = O(n)$
$S = O(n)$

c) $T(n) = 9T(n/3) + O(n^2)$



n^2

$$\text{Work} = (n/3^i)^2$$

$$= \frac{n^2 \cdot \cancel{9^i}}{\cancel{9^i}} = n^2$$

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

Maximum cost at each level is $O(n^2)$ and this doesn't change so $S = O(n^2)$