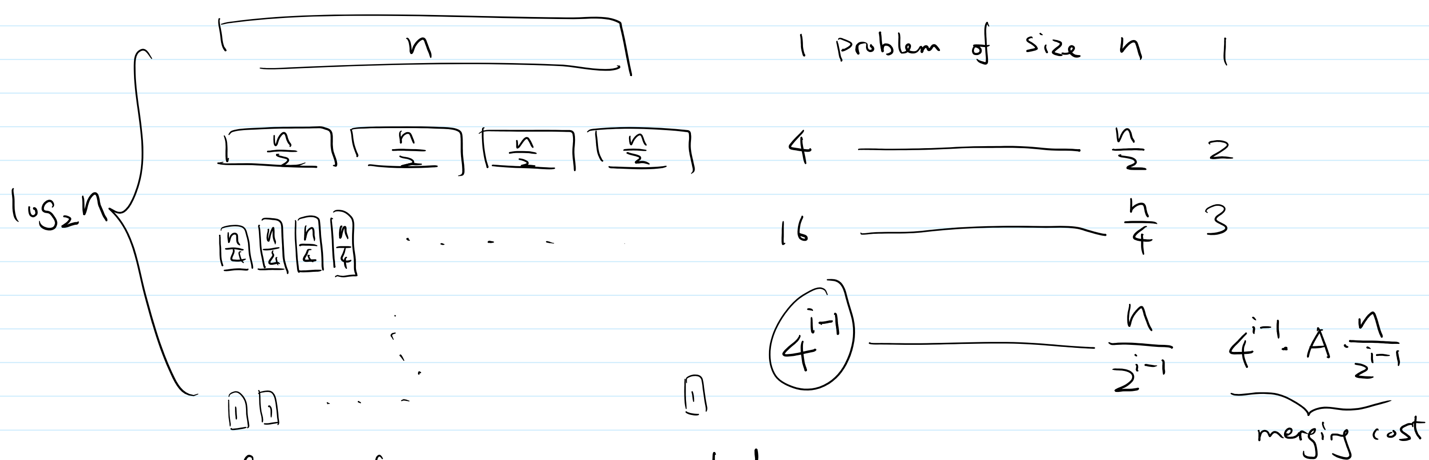
- Integer Multiplication
  - 1st attempt.
    - recursion. Let $T(n)$ to be the running time for multiplying 2 n-digit numbers.

$$T(n) = 4T\left(\frac{n}{2}\right) + A \cdot n$$

  - recursion tree



  - get the formula for recursion tree method

$$T(n) = 4\,\underline{T\left(\frac{n}{2}\right)} + \boxed{A \cdot n} \longrightarrow \text{merging cost at layer 1}$$

$$\left(T\left(\frac{n}{2}\right) = 4T\left(\frac{n}{4}\right) + A \cdot \frac{n}{2} \right)$$

$$= \underline{16\,T\left(\frac{n}{4}\right)} + \boxed{4 \cdot A \cdot \frac{n}{2}} + A \cdot n$$

merging cost for layer 2

$$\left(T\left(\frac{n}{4}\right) = 4T\left(\frac{n}{8}\right) + A \cdot \frac{n}{4} \right)$$

$$= 64\,T\left(\frac{n}{8}\right) + \boxed{16 \cdot A \cdot \frac{n}{4}} + \underline{4 \cdot A \cdot \frac{n}{2} + A \cdot n}$$

merging cost for layer 3

. . .

$$= \underline{\bigcirc \cdot T(1)} + \underline{\sum \text{merging cost for all layers}}$$

assume $T(1) = 0$ (does not change asymptotic behavior)

$$T(n) = \sum_{i=1}^{\#\text{layers}} \text{merging cost at layer } i$$

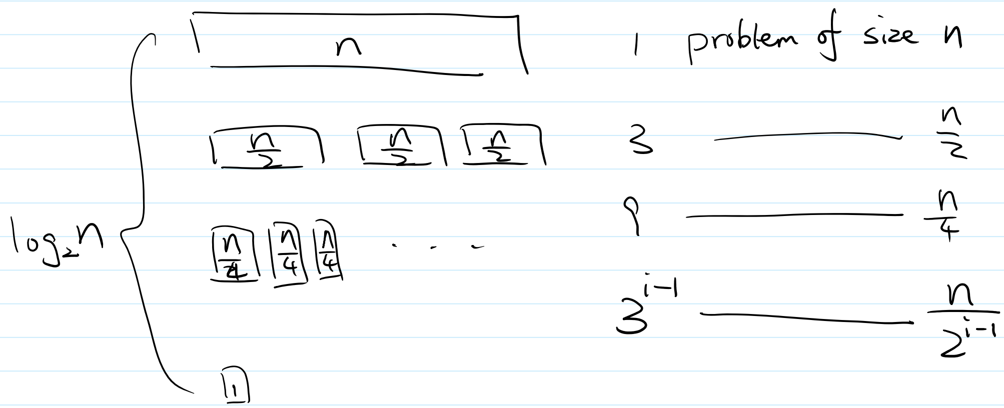
$$= \sum_{i=1}^{\log_2 n} 4^{i-1} \cdot A \frac{n}{2^{i-1}}$$

$$= (An) \cdot \sum_{i=1}^{\log_2 n} 2^{i-1} \qquad (1+2+4+\cdots+2^{\log_2 n - 1})$$

$$= A \cdot n\,(n-1) = O(n^2)$$

- Improved Algorithm

recursion : $T(n) = 3T\left(\frac{n}{2}\right) + A \cdot n$



by recursion Tree

$$T(n) = \sum_{i=1}^{\#\text{layers}} \text{merging cost at layer } i$$

$$= \sum_{i=1}^{\log_2 n} 3^{i-1} \cdot A \frac{n}{2^{i-1}}$$

$$= (An) \cdot \sum_{i=1}^{\log_2 n} \left(\frac{3}{2}\right)^{i-1} \qquad \left(1 + \frac{3}{2} + \left(\frac{3}{2}\right)^2 + \cdots + \left(\frac{3}{2}\right)^{\log_2 n - 1}\right)$$

$$\sum_{i=1}^{t} c^{i-1} = \boxed{\frac{c^t - 1}{c-1}} \approx \begin{cases} \boxed{c^t} & c > 1 \\ 1 & c < 1 \\ t & c = 1 \end{cases}$$

$$= (A \cdot n) \cdot \Theta\left(\left(\frac{3}{2}\right)^{\log_2 n}\right)$$

$$\boxed{\left(\frac{3}{2}\right)^{\log_2 n} \underset{(a = 2^{\log_2 a})}{=} 2^{\log_2\left[\left(\frac{3}{2}\right)^{\log_2 n}\right]} \underset{(\log a^b = b \log a)}{=} 2^{\log_2\left(\frac{3}{2}\right) \cdot \log_2 n} \underset{(2^{ab} = (2^b)^a)}{=} \left(2^{\log_2 n}\right)^{\log_2\left(\frac{3}{2}\right)} = n^{\log_2\left(\frac{3}{2}\right)}}$$

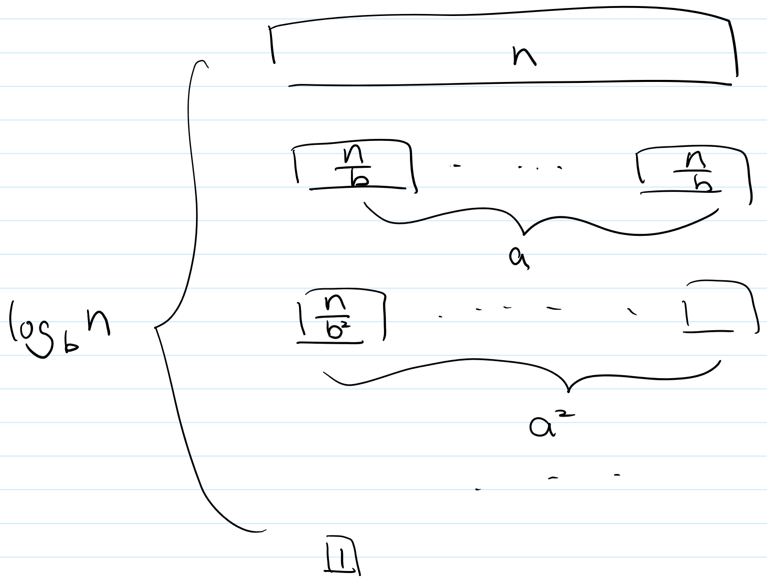
$$= \Theta\left(A \cdot n \cdot n^{\log_2\left(\frac{3}{2}\right)}\right)$$

$$= \Theta\left(A \cdot n \cdot n^{\log_2 3 - 1}\right)$$

$$= \Theta(A \cdot n^{\log_2 3})$$

$$= \Theta(n^{\log_2 3}) = o(n^2)$$

- Master Theorem $\quad T(n) = a\,T(\frac{n}{b}) + f(n)$



a^{i-1} problems of size $\frac{n}{b^{i-1}}$

merging cost $= a^{i-1} \cdot f\left(\frac{n}{b^{i-1}}\right)$

$$T(n) = \sum_{i=1}^{\#\text{layers}} \text{merging cost for layer } i$$

$$= \sum_{i=1}^{\#\text{layers}} a^{i-1} \cdot f\left(\frac{n}{b^{i-1}}\right)$$

$$= \sum_{i=1}^{\#\text{layers}} a^{i-1} \cdot \frac{n^c}{(b^c)^{i-1}} \qquad f(n) = n^c \text{ assumption}$$

$$= \sum_{i=1}^{\#\text{layers}} \underline{n^c \cdot \left(\frac{a}{b^c}\right)^{i-1}}$$

$$\frac{a}{b^c} = \begin{cases} = 1 & \text{case 2 in Thm} \quad n^c \log n \\ < 1 & \text{case 3 in Thm} \quad n^c \\ > 1 & \text{case 1 in Thm} \quad n^{\log_b a} \end{cases}$$
