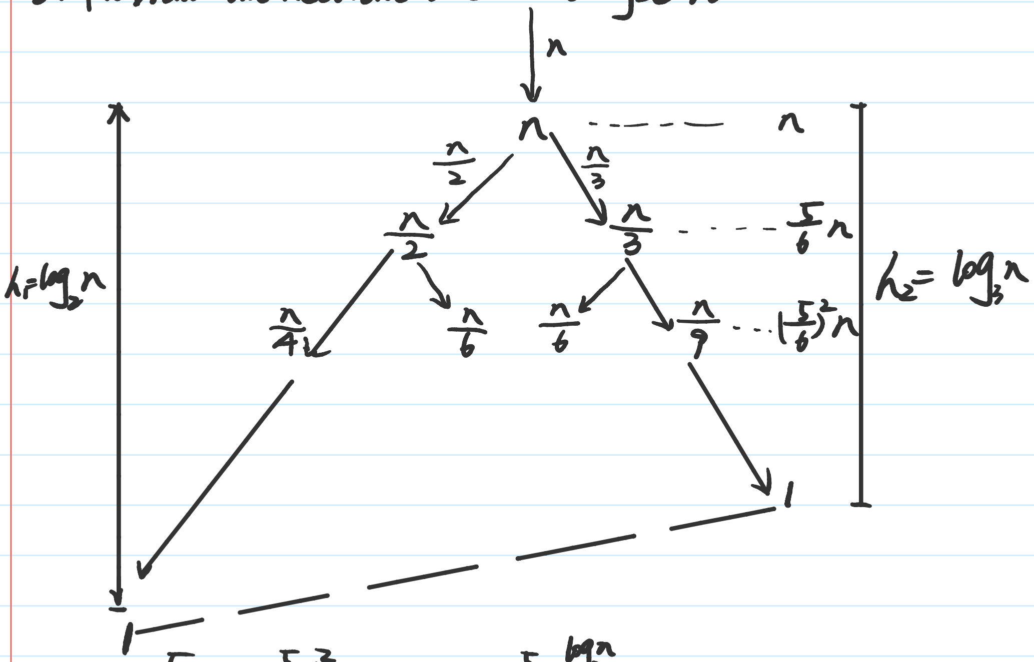
Assignment2

2021年9月27日 0:00

Name - Haoze He Net ID - hh2537

Q1:

Step1: Draw the Recursive tree to analyze it:



$$n + \frac{5}{6}n + (\frac{5}{6})^2 n + \dots + (\frac{5}{6})^{\log_2 n} \cdot n$$
$$= n[1 + \frac{5}{6} + (\frac{5}{6})^2 + \dots + (\frac{5}{6})^{\log_2 n}] \le n \sum_{i=1}^{\infty} (\frac{5}{6})^i = n \cdot \frac{1}{1-\frac{5}{6}}$$
$$T(n) = \Theta(n^2)$$

Step2: Use substition method to verify.

Assume $T(n) = \Theta(n^2)$

1) $T(n) = T(\frac{n}{2}) + T(\frac{n}{3}) + n \le C_1 n$
$$C_1 \frac{n}{2} + C_1 \frac{n}{3} + n \le C_1 n$$
$$\frac{5}{6} C_1 + 1 \le C_1$$
$$6 \le C_1$$

2) $T(n) = O(n^2)$ is proved

$T(n) = T(\frac{n}{2}) + T(\frac{n}{3}) + n \geq C_2 n$
$$C_2 \frac{n}{2} + C_2 \frac{n}{3} + n \geq C_2 n$$
$$\frac{5}{6} C_2 + 1 \geq C_2$$
$$6 \geq C_2$$

$T(n) = \Omega(n^2)$ is proved.

Q2.

Assume $T(n) = 2T(\frac{n}{2}) + cn\log_2 n$ is $O(n(\log_2 n)^2)$

Step1. Prove $T(n) = O(n(\log_2 n)^2)$

$$T(n) = 2C_1(\frac{n}{2} \cdot (\log_2 \frac{n}{2})^2) + cn\log_2 n \le C_1 n (\log_2 n)^2$$
$$C_1 \cdot (\log_2 \frac{n}{2})^2 + c\log_2 n \le C_1 (\log_2 n)^2$$
$$C_1 (\log_2 n - 1)^2 + C\log_2 n \le C_1 (\log_2 n)^2$$

$$
\begin{aligned}
C_1\cdot(\log_2\tfrac{n}{2})^2 + c\log_2 n &\le C_1(\log_2 n)^2 \\
C_1\cdot(\log_2 n - 1)^2 + c\log_2 n &\le C_1(\log_2 n)^2 \\
c\log_2 n &\le C_1\log_2^2 n - C_1\log_2^2 n + 2C_1\log_2 n - C_1 \\
c\log_2 n &\le 2C_1\log_2 n - C_1 \\
c\log_2 n &\le C_1(2\log_2 n - 1)
\end{aligned}
$$

We set $n \ge n_0 = 3$, $C_1 \ge \frac{c}{2 - 1/\log_2 3} = \frac{c}{2 - 1/\log_2 n} \le C_1$ $\Rightarrow$ Hence, $T(n) = O(n\log_2^2 n)$ is proved.

Step2: Prove $T(n) = \Omega(n\log_2^2 n)$

$$
\begin{aligned}
T(n) = 2C_2\left(\tfrac{n}{2}(\log_2\tfrac{n}{2})^2\right) + cn\log_2 n &\ge C_2 n\log_2^2 n \\
C_2 n(\log_2\tfrac{n}{2})^2 + cn\log_2 n &\ge C_2 n\log_2^2 n \\
C_2(\log_2 n - 1)^2 + c\log_2 n &\ge C_2\log_2^2 n \\
c\log_2 n &\ge 2C_2\log_2 n - C_2 \\
\frac{c}{2 - 1/\log_2 n} &\ge C_2
\end{aligned}
$$

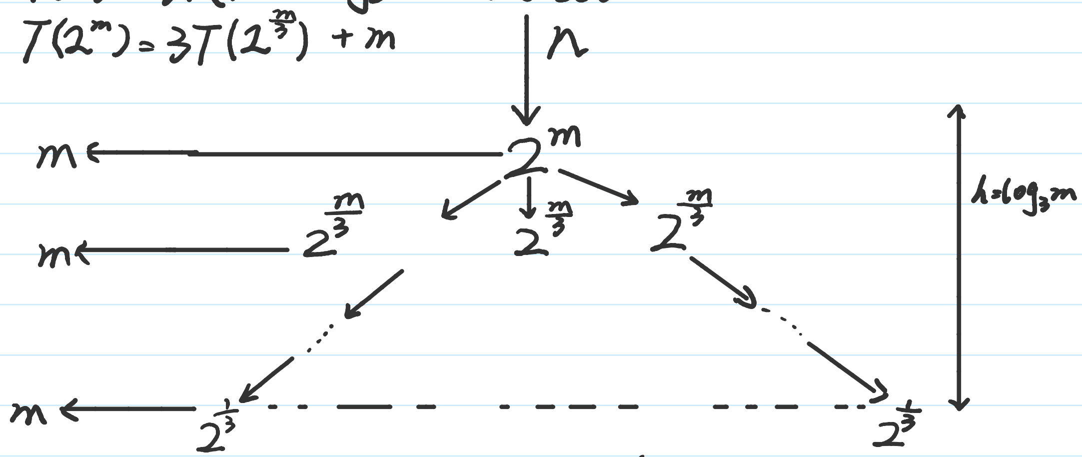
Since $\frac{c}{2 - 1/\log_2 n} > \frac{c}{2}$, for $\forall n \in R^+$ $n \ge 2$ $\Rightarrow$ Hence, $T(n) = \Omega(n\log_2^2 n)$ is proved.

Thus $T(n) = \Theta(n\log_2^2 n)$ is proved.

Q3.

$T(n) = 3T(n^{\frac{1}{3}}) + \log_2 n$: we set $n = 2^m$

$T(2^m) = 3T(2^{\frac{m}{3}}) + m$



$T(n) = O(m\cdot\log_3 m) = O(\log_2 n \cdot \log_3\log_2 n)$

Q4.

a)

According to master's method:

$T(n) = 3T(\frac{n}{3}) + \Theta(n)$. Compare $f(n)$ with $n^{\log_b a}$

① → $f(n) = n$

② → $n^{\log_b a} = n^{\log_3 3} = n$

Since ① = ② = n ⇒ case 2, the time complexity is $\Theta(n\lg n)$

b) According to master's method:

$T(n) = \ldots T(\frac{9n}{\ldots}) + \Theta\ldots$ Compare $f(n)$ with $n^{\log_b a}$

b) According to master's method:

$T(n) = 2T(\frac{9n}{10}) + \Theta(n)$. Compare $f(n)$ with $n^{\log_b a}$

① → $n^{\log_b a} = n^{\log_{\frac{10}{9}} 2} > n^{6}$

② → $f(n) = n$.

Since $n^{\log_b a} > n$. the time complexity is $\Theta(n^{\log_{10/9} 2})$

c) According to master's method:

$T(n) = 3T(\frac{n}{3}) + \Theta(n^2)$. Compare $f(n)$ with $n^{\log_b a}$

① → $a=3$. $b=3$. $n^{\log_b a} = n$

② → $f(n) = n^2$

Since $n^{\log_b a} < n^2$, the time complexity is $\Theta(n^2)$

△ Conclusion: algorithm a is the fastest one among them.

Q5. ① As for $T(n) = T(\frac{n}{2}) + n^2 \lg n$:

No, the master's method doesn't apply to this one.

Even though $f(n) = n^2 \lg n$ is asymptotically larger than $n^{\log_b a} = n^{\log_2 1} = 1$.

the problem is that it's not polynomially larger.

The ratio $f(n)/n^{\log_b a} = n^2 \lg n$ is asymptotically less than n^{ϵ} for $\forall \epsilon > 0$.

Thus, it falls into gap between Case 2 & 3.

② Provide the asymptotic upper bound:

$T(n) = T(\frac{n}{2}) + n^2 \lg n$,

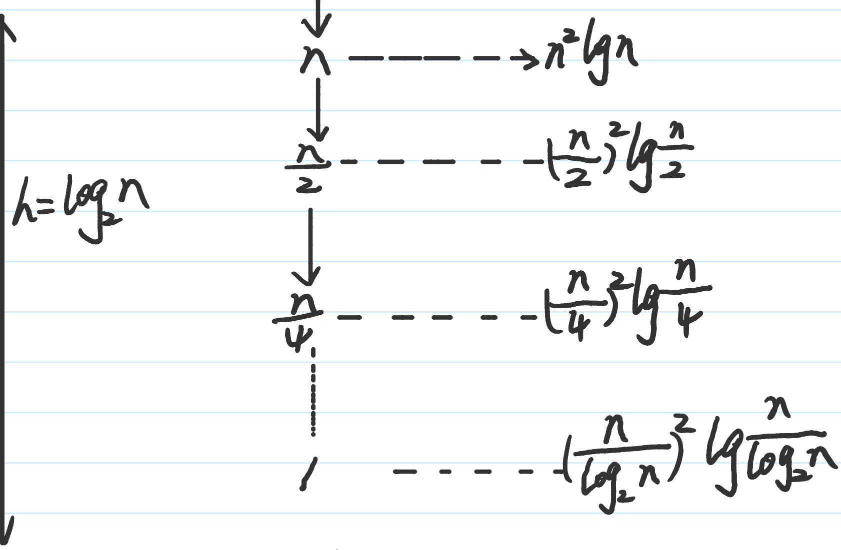
According to master's Theorem:

$a=1$, $b=2$, $f(n) = n^2 \lg n$

Hence:

$$T(n) = \Theta(n^{\log_2 1}) + \sum_{j=0}^{\lfloor \log_2 n - 1 \rfloor} a^j f(n_j)$$

$$= \Theta(1) + \sum_{j=0}^{\lfloor \log_2 n - 1 \rfloor} a_j f(n_j)$$



$$= n^2 \lg n + \frac{1}{2} n^2 \lg \frac{n}{2} + \cdots + \frac{1}{\log_2 n} n^2 \log \frac{n}{\log_2 n}$$

$$= n^2 \lg n + \frac{1}{2} n^2 (\lg n - \lg 2) + \cdots + \frac{1}{\log_2 n} n^2 (\lg n - \lg \lg n) \le (1 + \frac{1}{2} + \frac{1}{4^2} \cdots + \frac{1}{\log_2^2 n}) n^2 \lg n$$

$$\le \frac{4}{3} n^2 \lg n$$

Hence, the asymptotic upper bound is $g(n) = \frac{4}{3} n^2 \lg n$

$T(n) = \Theta(n^2 \lg n)$