CS 721

HW#1 Solution.

(1) (a) $$T(n) = T(n/3) + 2$$

$a = 1,\ b = 3,\quad f(n) = 2 = \Theta(1)$

$n^{\log_b a} = n^0 = 1 = \Theta(1)$

$\therefore$ Using master theorum (case 2), $T(n) = \Theta(n^{\log_b a} \log n)$
$= \Theta(\log n)$

(b) (i) $$T(n) = T(n/3) + 1$$

$a = 1,\ b = 3,\quad f(n) = 1 = \Theta(1)$

Same as above $T(n) = \Theta(\log n)$

(ii) $$T(n) = T\left(\frac{n}{3/2}\right) + 1$$

$a = 1,\ b = 3/2,\ f(n) = 1 = \Theta(1).$

Same as before $T(n) = \Theta(\log n)$.

(2) (a) $$T(n) = T\left(\frac{n}{\sqrt{n}}\right) + (\sqrt{n} - 1)$$

$\Rightarrow$ $$T(n) = T(\sqrt{n}) + (\sqrt{n} - 1)$$

Set $n = 2^m \Rightarrow \sqrt{n} = 2^{m/2}$

$\therefore$ $$T(2^m) = T(2^{m/2}) + (2^{m/2} - 1) \quad \text{—— (1)}$$

Set $S(m) = T(2^m)$, then $S(m/2) = T(2^{m/2})$

$\therefore$ Eq (1) can be written as

$$S(m) = S(m/2) + (2^{m/2} - 1)$$

Now, $a = 1,\ b = 2,\ f(m) = 2^{m/2} - 1$

$m^{\log_b a} = m^0 = 1$.

$\therefore \quad f(m) = 2^{m/2} - 1 = \Omega(m^{0+\varepsilon})$ for say $\varepsilon = 1$.

Checking of regularity condition

$$a f(m/b) = 1 \cdot 2^{(m/2)/2}$$
$$= 2^{m/4} \le 2^{m/2} = f(m)$$
(here $c = 1$).

$\therefore$ Using master theorem (case 3).

$$S(m) = \Theta(f(m)) = \Theta(2^{m/2})$$

$$T(2^m) = \Theta(2^{m/2})$$

$$\therefore T(n) = T(2^m) = \Theta(2^{m/2}) = \Theta(\sqrt{n}).$$

(b) $\quad T(n) = T\left(\frac{n}{\log n}\right) + \log n - 1$

Here $a = 1$, $b = \log n$.
Since b is not constant but depends on n, we can not use master theorem.

(c) $\quad T_1(n) = T_1\left(\frac{n}{2}\right) + \log n$

$a = 1$, $b = 2$, $f(n) = \log n$.

$n^{\log_b a} = n^0 = 1$. However $f(n) \ne \Omega(n^{\log_b a + \varepsilon})$

Thus master theorem does not apply.

One possibility is to use Extended master theorem.

$f(n) = \Theta(n^{\log_b a} \log n)$

Thus $T_1(n) = \Theta(n^{\log_b a} \log^2 n) = \Theta(\log^2 n)$

Otherwise do it by change of variable.

Let $n = 2^m$. then $\frac{n}{2} = 2^{m-1}$. , $m = \log_2 n$

$\therefore T_1(2^m) = T_1(2^{m-1}) + m$

Let $S_1(m) = T_1(2^m)$

Then $S_1(m) = S_1(m-1) + m$

$$\therefore \quad S_1(m) = m + S_1(m-1)$$
$$= m + (m-1) + S_1(m-2)$$
$$= m + (m-1) + (m-2) + S_1(m-3)$$
$$\vdots$$
$$= m + (m-1) + \cdots + 1$$
$$= \frac{m(m+1)}{2}$$

$$\therefore \quad S_1(m) = \Theta(m^2)$$

$$\therefore \quad T_1(2^m) = \Theta(m^2)$$

$$\Rightarrow \quad T_1(n) = T_1(2^m) = \Theta(m^2) = \Theta(\log^2 n)$$

(d) $T_2(n) = T_2\left(\frac{n}{\sqrt{n}}\right) + \log n - 1$

$\Rightarrow \quad T_2(n) = T_2(\sqrt{n}) + \log n - 1$

Set $n = 2^m$, $\Rightarrow \sqrt{n} = 2^{m/2}$ and $\log n = m$.

$\therefore \quad T_2(2^m) = T_2(2^{m/2}) + m - 1$ —— (1)

Set $S_2(m) = T_2(2^m)$

$\therefore$ Eq (1) becomes

$S_2(m) = S_2(m/2) + m - 1$

$a = 1$, $b = 2$, $f(m) = m - 1$

$m^{\log_b a} = m^0 = 1$

$\therefore \quad f(m) = m - 1 = \Omega\left(m^{\log_b a + \epsilon}\right)$ where $\epsilon = 0.5$

$\therefore$ Using case 3 to master theorem

$$S_2(m) = \Theta(m)$$

$$\therefore \quad T_2(2^m) = S_2(m) = \Theta(m)$$

$$\therefore \quad T_2(n) = T_2(2^m) = S_2(m) = \Theta(m) = \Theta(\log n)$$

Thus summations of our original problem $T(n)$ has the following ~~bound~~ bound.

$$c_2 \log n \leq T(n) \leq c_1 \log^2 n$$

3 (a) <u>false</u>

let $f(n) = \frac{1}{n}$ then $(f(n))^2 = \frac{1}{n^2}$

$$\frac{1}{n} \neq O\left(\frac{1}{n^2}\right)$$

(b) <u>false</u>

If $f(n) = \Theta(f(n/2))$ it must be the case that $f(n) = O\, f(n/2)$. We show one counter example where that is not the case.

let $f(n) = 2^n$. $\therefore f(n/2) = 2^{n/2}$.

clearly $2^n \leq c\, 2^{n/2}$ does not hold. because then c have to be larger than $2^{n/2}$.

(c) <u>false</u>

let $f(n) = \frac{1}{n^2}$, $\sqrt{f(n)} = \frac{1}{n}$.

clearly $\frac{1}{n^2} \neq \Omega\left(\frac{1}{n}\right)$

(d) Let $h(n) = \max\{f(n), g(n)\}$

therefore $h(n) \leq f(n) + g(n)$

$\therefore h(n) = O(f(n) + g(n))$

Now $h(n) \geq f(n)$ and $h(n) \geq g(n)$

$\therefore h(n) + h(n) \geq f(n) + g(n)$

$\Rightarrow h(n) \geq \frac{1}{2}(f(n) + g(n))$

$\therefore h(n) = \Omega(f(n) + g(n))$

$\therefore h(n) = \Theta(f(n) + g(n))$

4. Need to prove $T(n) \leq cn^2$

Assume this holds for all sub problem of ~~of~~ size $n/2$ or smaller, i.e. $T(n/2) \leq c(n/2)^2$

Plugging in the recurrence relation

$$T(n) = 4T(n/2) + n$$
$$\leq 4 \cdot c(n/2)^2 + n$$
$$= cn^2 + n.$$

This is not what we want!

Instead we will try to prove $T(n) \leq c(n^2 - n)$, which will imply $T(n) \leq cn^2 \Rightarrow T(n) = O(n^2)$

Assume this holds for all sub problem of size $n/2$ or smaller.

$\therefore T(n/2) \leq c((n/2)^2 - n/2)$

Plugging back into recurrence relation gives us

$$T(n) = 4T(n/2) + n$$

$$\leq 4c\left((n/2)^2 - n/2\right) + n$$

$$= 4c(n^2/4 - n/2) + n$$

$$= cn^2 - 2cn + n$$

$$= cn^2 - cn + cn - 2cn + n$$

$$= cn^2 - cn - cn + n$$

$$= c(n^2 - n) - n\underbrace{(c-1)}_{\geq 0}$$

$$\leq c(n^2 - 1) \qquad \text{If we chose } c \geq 1$$