$n^{0.5}$ $n^{0.5+\epsilon}$ $\frac{1}{2}$ 0.6 $\frac{1}{2}$

1 0.5 1

Cosc 336 Brandon Richards
Assignment 02

A1. $T(n) = 2T\left(\frac{n}{4}\right) + 3$

$a = 2,\ b = 4,\ f(n) = 3$

Test Case 1 $\quad f(n) = O(n^{\log_b a - \epsilon})$ $\quad \epsilon > 0$ $\quad O(n^{\log_b a})$

$3 = O(n^{\log_4 2 - \epsilon})$ $\quad 0.5-$ $\quad 0.5-1$ $\quad -0.5$

$3 = O(n^{0.5-\epsilon})$

$3 = O(n^{0.5+\epsilon})$

$T(n) = \Theta(n \quad)$

Iteration: $T(n) = 2T\left(\frac{n}{4}\right) + 3$

$\cancel{2T\left(\frac{n}{4}\right)} = 2^2 T\left(\frac{n}{4^2}\right) + 3 \cdot 2$

$\cancel{2^2 T\left(\frac{n}{4^2}\right)} = 2^3 T\left(\frac{n}{4^3}\right) + 4 \cdot 2^2$

$2^3 T\left(\cancel{\frac{n}{4^3}}\right) = 2^4 T\left(\frac{n}{4^4}\right) + 4 \cdot 2^3$

$2^{k-1} T\left(\frac{n}{4^{k-1}}\right) = 2^k T\left(\frac{n}{4^k}\right) + 4 \cdot 2^{k-1}$

$2^k T\left(\frac{n}{4^k}\right) = 1 * 2^k$

$\frac{n}{4^k} = 1$ $\quad T(n) = 4(2^0 + 2^1 + 2^2 + \dots 2^{k-1}) + 2^k$

$n = 4^k$ $\quad = 4\left[\frac{a^{m+1}-1}{a-1}\right] \approx 4\left[\frac{2^{(k-1)+1}-1}{2-1}\right] + 2^k$

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

$= 4 \cdot 2^k - 4 + 2^k = 5 \cdot 2^k - 4$

$T(n) = 5 \cdot 2^{\log_4 n} - 7$

A2. $T(n) = 3T\left(\frac{n}{4}\right) + 2n$

$a = 3 \quad b = 4 \quad f(n) = 2n$

Test Case 1: $\Theta(n^{\log_b a - \epsilon})$ $\quad f(n) = O(n^{\log_4 3 - \epsilon})$

$2n = O(n^{0.79 - \epsilon})$

$\epsilon > 0 \quad 1 \geq 0.79 - \epsilon$

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

$2n = \Theta(n^{\log_4 3})$

$2n = \Theta(n^{0.78})$

$1 > 0.78$

Test Case 3 $\Theta(f(n))$ $\quad f(n) = \Omega(n^{\log_b a + \epsilon})$ & $\epsilon > 0$, $c < 1$

$af(n/b) \leq cf(n)$ for large n

Is there $\epsilon > 0$, $c < 1$ such that

$f(n) = 2n = \Omega(n^{\log_4 3 + \epsilon})$

$3 \times 2\left(\frac{n}{4}\right) < c \times 2n$

$0 \quad 0.5 \quad \log_4 3 \quad \log_4 3 + \epsilon \quad 1$

$\epsilon = \frac{1 - \log_4 3}{2}$

$\frac{3n}{2} < c2n$

$c = \frac{4}{5}$

$\frac{3n}{2} \cdot \frac{1}{2n} < c$

$\frac{3}{4} < c$

$T(n) = \Theta(n)$

Iteration: $T(n) = 3T\left(\frac{n}{4}\right) + 2n$

$3T\left(\frac{n}{4}\right) = 3^2 T\left(\frac{n}{4^2}\right) + 2n \cdot 3$

$3^2 T\left(\frac{n}{4^2}\right) = 3^3 T\left(\frac{n}{4^3}\right) + 2n \cdot 3^2$

$\frac{3}{4} \cdot \frac{4}{4}$

$\frac{3}{4} + \frac{1}{10}$

$\frac{30 + 4}{40} = \frac{34}{40}$

$$\cancel{3^3 T\left(\frac{n}{4^3}\right)} = 3^4 T\left(\frac{n}{4^4}\right) + 2n \cdot 3^3$$

$$3^{k-1} T\left(\frac{n}{4^{k-1}}\right) = 3^k \left(\frac{n}{4^k}\right) + 2n \cdot 3^{k-1}$$

$$3^k T\left(\frac{n}{4^k}\right) = 1 \cdot 3^{k-1}$$

$k = \log_4 n$

$$T(n) = 2\left(n + n + n \cdots n\right) \cdot \left(3^0 + 3^1 + 3^2 + \cdots 3^{k-1}\right) + 3^k$$

$$= 2n \cdot \left[\frac{3^{(k-1)+1} - 1}{3-1}\right] + 3^k$$

$$T(n) = 2n \cdot \left[\frac{3^k - 1}{2}\right] + 3^k$$

$$T(n) = 2n\left[\frac{3^{\log_4 n} - 1}{2}\right] + 3^{\log_4 n}$$

A3. $T(n) = T(n-2) + 3$

Iteration: $T(n) = T(n-2) + 3$

$T(n-2) = T(n-2\cdot 2) + 3$

$T(n-2\cdot 2) = T(n-2\cdot 3) + 3$

$T(n-2\cdot 3) = T(n-2\cdot 4) + 3$

$T(n-2\cdot(k-1)) = T(n-2\cdot k) + 3$ — k^{th} 2nd last

$T(n-2\cdot k) = 1 + 3$ — last

$k = \frac{n-1}{2}$ — $T(n) = 4$

A4. $T(n) = 2T(n-1) + 1$

$2T(n-1) = 2^2 T(n-2) + 2 \cdot 1$

$2^2 T(n-2) = 2^3 T(n-3) + 2^2 \cdot 1$

$2^3 T(n-3) = 2^4 T(n-4) + 2^3 \cdot 1$

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

$2^k T(n-k) = 1 \cdot 2^k$

$n = k$
$k = n$

$T(n) = 2^k + 1(2^0 + 2^1 + 2^2 \ldots + 2^{k-1})$

$T(n) = 2^k + \left[\dfrac{2^{(k-1)+1} - 1}{2 - 1}\right]$

$= 2^k + \dfrac{2^k - 1}{1}$

$= 2^k + 2^k - 1$

$= 2^{1+k} - 1$

$T(n) = 2^{1+n} - 1$

A5. $T(n) = 4T\left(\frac{n}{2}\right) + n \log n$

Iteration: $4T\left(\frac{n}{2}\right) = 4^2 T\left(\frac{n}{4}\right) + n \log n \cdot 4$

$4^2 T\left(\frac{n}{4}\right) = 4^3 T\left(\frac{n}{6}\right) + n \log n \cdot 4^2$

$4^3 T\left(\frac{n}{6}\right) = 4^4 T\left(\frac{n}{8}\right) + n \log n \cdot 4^3$

$4^{k-1} T\left(\frac{n}{2^{k-1}}\right) = 4^k T\left(\frac{n}{2^k}\right) + n \log n \cdot 4^{k-1}$ ~~k~~ $+h$ $\log$

$4^k T\left(\frac{n}{2^k}\right) = 1 \cdot 4^k + n \log n \cdot 4^k$ Last

$n = 2^k$

$k = \log_2 n$

$T(n) = (4^0 + 4^1 + 4^2 \dots 4^{k-1})$
$\times (n + n + n \dots n)$
$\times (\log n + \log n + 1 + \dots \log(n+h))$
$+ 4^k$

$T(n) =$

Master Theorem:

$a = 4$, $b = 2$, $f(n) = n \log n$

Case 1: $\Theta(n^{\log_b a - \epsilon}) = f(n) = n \log n$

$n \log n = \Theta(n^{\log_2 4 - \epsilon})$

yes

$n \log n < \Theta(n^{\log_2 4 - \epsilon})$

$\epsilon = \frac{\log_2 4 - 1}{2}$

$\epsilon \geq 0$

n 2

0 1 2

A6. $T(n) = 3T\left(\frac{n}{5}\right) + n \log n$

Master theorem

$a = 3$, $b = 5$, $f(n) = n \log n$

$\epsilon > 0 \log_5 3$

0 0.68260 1

Case 1: $\Theta(n^{\log_b a - \epsilon})$ $= f(n) = (n^{\log_5 3 - \epsilon})$

$n \log n = (n^{0.68260 - \epsilon})$

No $\quad 1 > 0.68260 > \epsilon$

Case 2: $\Theta(n^{\log_b a} \log n)$ $\quad f(n) = \Theta(n^{\log_b a})$

$n \log n = \Theta(n^{\log_5 3})$

$n^1 \log n = \Theta(n^{0.68})$

No $\quad 1 > 0.68$

Case 3: $\Theta(f(n))$ $\quad f(n) = \Omega(n^{\log_b a + \epsilon})$ and $\quad \epsilon > 0$, $c < 1$

$af(n/b) < cf(n)$

$\log_5 3 - 1$

$\log_5 3$ $\quad f(n) = \Omega(n^{\log_5 3 + \epsilon})$ and $3\left(\frac{n}{5} \log\left(\frac{n}{5}\right)\right) < c \times n \log n$

0 0.5 0 1

$\log_5 3 - 1$

$3\left(\frac{n}{5} \log n - \log 5\right) < c \times n \log n$

$\log_5 3 + \epsilon \leq 1$

$\frac{3n(\log n - \log 5)}{5} < c \times n \log n$

$\epsilon = \frac{1 - \log_5 3}{2}$ $\quad T(n) = \Omega(n \log n)$

$c = \frac{6}{10}$

Iteration: $3T\left(\frac{n}{5}\right) + n\log n$

1st ..... $3T\left(\frac{n}{5}\right) = 3^2 T\left(\frac{n}{5^2}\right) + n\log n$

2nd ..... $3^2 T\left(\frac{n}{5^2}\right) = 3^3 T\left(\frac{n}{5^3}\right) + n\log n$

3rd ..... $3^3 T\left(\frac{n}{5^3}\right) = 3^4 T\left(\frac{n}{5^4}\right) + n\log n$

kths ..... $3^{k-1} T\left(\frac{n}{5^{k-1}}\right) = 3^k T\left(\frac{n}{5^k}\right) + n\log n$

$\frac{n}{5^k} = 1$

$T(1)$

last step $3^k T\left(\frac{n}{5^k}\right) = 1 * 3^k$ $\quad 3^{n-1} T\left(\frac{n}{5^{n-1}}\right) = 3^n(1)$

$T(n) = (n\log n + n\log n + n\log n \ldots n\log n) + 3^k$

$n = 5^k$

$k = ?$ $\quad k = \log_5 n$

$=$

A7. $T(n) = 2T\left(\frac{n}{3}\right) + n^2$

Master Theorem:

$a = 2,\ b = 3,\ f(n) = n^2$

Case 1: $\Theta(n^{\log_b a - \epsilon}) \rightarrow f(n) = \Theta(n^{\log_b a - \epsilon})$

$n^2 = (n^{\log_3 2 - \epsilon})$

$n^2 = n^{0.63092 - \epsilon}$

$2 > 0.63092$

no

Case 2: $\Theta(n^{\log_b a}) \rightarrow f(n) = \Theta(n^{\log_b a})$

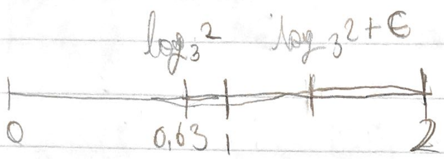
$n^2 = \Theta(n^{\log_3 2}) \rightarrow n^2 = \Theta(n^{0.63092})$

me

Case 3: $\Theta(f(n))$

$$f(n) = \Omega(n^{\log_b a + \epsilon}) \text{ and } af(n/b) < cf(n)$$

$$n^2 = \Omega(n^{0.63 + \epsilon}) \text{ and } 2\left(\frac{n}{3}\right)^2 < c \cdot n^2$$



$$2\left(\frac{n}{3}\right)^2 < cn^2$$

$$\frac{1}{n^2} \cdot \frac{2n^2}{9} < \frac{c\cancel{n^2}}{\cancel{n^2}}$$

$$\frac{2}{9} < c$$

$$c = \frac{1}{9}$$

$$\log_3 2 + \epsilon \leq 2$$

$$\epsilon = \frac{2 - \log_3 2}{2}$$

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

Iteration: $T(n) = 2T\left(\frac{n}{3}\right) + n^2$

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

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

$$2^3 T\left(\frac{n}{3^3}\right) = 2^4 T\left(\frac{n}{3^4}\right) + n^2 \cdot 2^3$$

k^{th} step. $2^{k-1} T\left(\frac{n}{3^{k-1}}\right) = 2^k T\left(\frac{n}{3^k}\right) + \boxed{n^2 \cdot 2^{k-1}}$

last $2^k T\left(\frac{n}{3^k}\right) = 1 * 2^k$

$n = 3^k$

$k = \log_3 n$

$$T(n) = (n^2 + n^2 + n^2 + n^2 \ldots n^2) * (2^0 + 2^1 + 2^2 + \ldots 2^{k-1}) + 2^k$$

$$T(n) = n^2 \cdot \left[\frac{2^{(k-1)+1} - 1}{2-1}\right] + 2^k$$

$$= n^2 \cdot \left[\frac{2^k - 1}{1}\right] + 2^k$$

$$= n^2 \cdot [2^k - 1] + 2^k$$

$$= 2^k n^2 - n^2 + 2^k \qquad 2^k(n^2+1) - n^2$$

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

Part B

2.3-4 (p 39) & Prob 2-1 (p 39)

We can express insertion sort as a recursive procedure as follows. In order to sort A[1...n], we recursively sort A[1...n-1] and then insert A[n] into the sorted array A[1...n-1]. Write a recurrence for the running time of this recursive version of insertion sort.

$$T(n) = \begin{cases} 1 & n = 1 \\ T(n-1) + n & n > 1 \end{cases}$$

2.1 Insertion sort

merge sort worst case $\Theta(n \log n)$

Insertion sort $\Theta(n^2)$ - worst case

a. $\Theta(n^2)$ n/k sublist k = length so $\Theta(k^2)$ time
to do $\frac{n}{k}$ sublists, $\frac{n}{k} \cdot k^2 = \Theta(nk)$ time

b. Merge sublists in $O(n \lg(\frac{n}{k}))$
Each sublist will be sorted in a larger list below, so and so on.
Eventually there will be two large sublists to compare which will then be sorted into the final sorted list.

c. $O(n \log n)$
$O(nk + n \log(\frac{n}{k})$
$n \log n - 1)$
$(nk + n\log n - n)$

$k \leq \log n$

(d) Since k is the length & $k \leq \log n$. k should be determined by the running time of the insertion list.