CSCI 260 Test Prep

Midterm 1

Master Theo

(1)

The Master Theorem

For a function $T(n)$ defined on positive integers, where $T(n) = aT(\frac{n}{b}) + f(n)$ and $f(n)$ is valued function, and $a \geq 1$ and $b > 1$, then:

1. $f(n) \in O(n^{\log_b a - \varepsilon})$ for some const

$$\Rightarrow T(n) \in \Theta(n^{\log_b a})$$

2. $f(n) \in \Theta(n^{\log_b a})$

$$\Rightarrow T(n) \in \Theta(n^{\log_1 a} \lg n)$$

3. $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ for some cons

regularity condition { AND $\exists$ constant c s.t. $0 < c < 1$ and $af(n/b) < cf(n)$ when n sufficiently

$$\Rightarrow T(n) \in \Theta(f(n)).$$

When we are analyzing the running times of Divide & Conquer - style recursive algorithms, we often arrive at an "intermediate" expression that is self-referential, called a "Recurrence Relation"

E.g. $T(1) = 1$

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

The above recurrence is the result of analyzing Mergesort, which can be expressed schematically as:

mergesort($A[1..n]$)

if $n \leq 1$ do $O(1)$ work. $O(1)$

mergesort($A[1..n/2]$) $T(n/2)$

mergesort($A[n/2+1..n]$) $T(n/2)$

Do $O(n)$ work merging the two lists. $O(n)$

Hence the running time, in Big O, is expressed as

$T(1) = 1$ ← This is assumed, and does not appear in the Master Theorem. It really means $O(1)$

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

To put these into a reasonable Big-O form to report as running times, we use the Master Theorem.

(4)

Q 1. Use Master Theorem to give a Θ - expression for the following recurrence relations:

a) $T(n) = 3T(n/4) + n\sqrt{n}$

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

c) $T(n) = 16T(n/4) + n^2 \log n$

a) $T(n) = 3T(n/4) + n^{1.5}$

(5)

(6)

a) $T(n) = 3T(n/4) + n^{1.5}$

$n^{1.5}$? $\in$ $O(\quad)$

$\Theta(\quad)$

$\Omega(\quad)$

(7)

a) $T(n) = 3T(n/4) + n^{1.5}$

$n^{1.5} \overset{?}{\in} O(n^{\log_4 3})$

$\Theta(\quad)$

$\Omega(\quad)$

8

a) $T(n) = 3T(n/4) + n^{1.5}$

$n^{1.5} \overset{?}{\in} O(n^{\log_4 3 - \varepsilon})$ for some $\varepsilon > 0$

$n^{1.5} \overset{?}{\in} \Theta(n^{\log_4 3})$

$n^{1.5} \overset{?}{\in} \Omega(n^{\log_4 3 + \varepsilon})$ for some $\varepsilon > 0$

(9)

a) $T(n) = 3T(n/4) + n^{1.5}$

$n^{1.5} \overset{?}{\in} \begin{cases} O(n^{\log_4 3 - \varepsilon}) & \text{for some } \varepsilon > 0 \\ \Theta(n^{\log_4 3}) & \\ \Omega(n^{\log_4 3 + \varepsilon}) & \text{for some } \varepsilon > 0 \end{cases}$

$1 < 3 < 4$

hence $\log_4 1 < \log_4 3 < \log_4 4$

hence $0 < \log_4 3 < 1 \quad < 1.5$

∴ it might be case 3, Ω. Lets check the regularity condition.

" AND $\exists$ constant c s.t. $0 < c < 1$ and

$a f(n/b) < c f(n)$ when n sufficiently large

$3\left(\frac{n}{4}\right)^{1.5} < c \cdot n^{1.5}$ for some $0 < c < 1$

(10)

a) $T(n) = 3T(n/4) + n^{1.5}$

$n^{1.5} \overset{?}{\in} O(n^{\log_4 3 - \varepsilon})$ for some $\varepsilon > 0$

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

$\Omega(n^{\log_4 3 + \varepsilon})$ for some $\varepsilon > 0$

$1 < 3 < 4$

hence $\log_4 1 < \log_4 3 < \log_4 4$

hence $0 < \log_4 3 < 1 < 1.5$

∴ it might be case 3, Ω. Lets check the regularity condition.

"AND $\exists$ constant c s.t. $0 < c < 1$ and

$af(n/b) < cf(n)$ when n sufficiently large

$3\left(\frac{n}{4}\right)^{1.5} < c\, n^{1.5}$ for some $0 < c < 1$

$\frac{3 \cdot n^{1.5}}{4 \cdot 2} < c \cdot n^{1.5}$

divide both sides by $n^{1.5}$

$\frac{3}{8} < c$ Yes, $\exists$ such a c

eg $c = \frac{4}{8}$ ie $\frac{1}{2}$

∴ By Master Theorem case (3), $T(n) \in \Theta(\quad)$ ▨

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

(11)

$$n^2 \in \begin{cases} O(\quad) \\ \Theta(\quad) \\ \Omega(\quad) \end{cases}$$

12

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

$O(n^{\log_3 9 - \epsilon})$ for some $\epsilon > 0$

$n^2 \in \Theta(\quad)$

$\Omega(\quad)$

13

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

$O(n^{2-\varepsilon})$ for some $\varepsilon > 0$

$n^2 \in \Theta(n^2)$

$\Omega(n^{2+\varepsilon})$ for some $\varepsilon > 0$

$T(n) \in \Theta(n^{\log_b a})$

$T(n) \in \Theta(n^{\log_b a} \lg n)$

$T(n) \in \Theta(f(n))$

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

$n^2 \in O(n^{2-\varepsilon})$ for some $\varepsilon > 0$ — $T(n) \in \Theta(n^{\log_b a})$

$n^2 \in \Theta(n^2)$ — $T(n) \in \Theta(n^{\log_b a} \lg n)$

$n^2 \in \Omega(n^{2+\varepsilon})$ for some $\varepsilon > 0$ — $T(n) \in \Theta(f(n))$

$n^2 \in \Theta(n^{\log_3 9})$,

$\therefore$ by Master Theorem case (2), $T(n) \in O(n^2 \lg n)$

c) $T(n) = 16T(n/2) + n^2 \log n$ (15)

$$n^2 \log n \in O(n^{\log_2 16 - \varepsilon}) \quad \varepsilon > 0$$

ie. $n^2 \log n \in O(n^{4 - \varepsilon})$ where $\varepsilon = 1$

$\therefore$ By MT case ①, $T(n) \in \Theta(n^4)$.

Homework:

1. Give the Recurrence Relation for the running time of the following program:

```
recurse(A[1...n])
    if n ≤ 1 return
    recurse(A[1... n/2])
    recurse(A[2... n/2 + 1])
    recurse(A[3.... n/2 + 2])
    k = 0
    for (i = 1 to n)
        for (j = 1 to n)
            k++
```

"Do $O(n^2)$ work"

2. Give the Θ-expression for the following recurrence relations, by applying the Master Theorem.

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

b) $T(n) = 25T(n/5) + n^2$

c) $T(n) = 5T(n/5) + n^2 \log n$

d) $T(n) = 5T(n/5) + n \lg n$

CSCI 260 Midterm 1 Test Preparation: Master Theorem

The Master Theorem provides a method to transform certain recurrence relations into asymptotic Θ running times for algorithms, particularly those following a Divide and Conquer paradigm. If none of its specific cases apply, the theorem is considered "silent."

The Master Theorem

The Master Theorem applies to recurrence relations of the form: $$ T(n) = aT\left(\frac{n}{b}\right) + f(n) $$ where:

- $a \ge 1$ (number of subproblems)
- $b > 1$ (factor by which input size is divided)
- $f(n)$ is a positive-valued function (cost of dividing/combining).

The theorem compares $f(n)$ with $n^{\log_b a}$ to determine the asymptotic running time:

Case 1

If $f(n) \in O(n^{\log_b a - \varepsilon})$ for some constant $\varepsilon > 0$, then $T(n) \in \Theta(n^{\log_b a})$.

- This means $f(n)$ grows polynomially *slower* than $n^{\log_b a}$.

Case 2

If $f(n) \in \Theta(n^{\log_b a})$, then $T(n) \in \Theta(n^{\log_b a} \log n)$.

- This means $f(n)$ grows asymptotically *at the same rate* as $n^{\log_b a}$.

Case 3

If $f(n) \in \Omega(n^{\log_b a + \varepsilon})$ for some constant $\varepsilon > 0$, AND a *regularity condition* holds: $a f(n/b) \le c f(n)$ for some constant $0 < c < 1$ when n is sufficiently large. Then $T(n) \in \Theta(f(n))$.

- This means $f(n)$ grows polynomially *faster* than $n^{\log_b a}$.

Recurrence Relations and Merge