Aidan Schneider
10/06/25

Massachusetts Institute of Technology
6.046J/18.410J: Introduction to Algorithms
Professors Michel Goemans and Piotr Indyk

Handout 9
February 15, 2002

Master Theorem Worksheet

This is a worksheet to help you master solving recurrence relations using the Master Theorem. For each recurrence, either give the asympotic solution using the Master Theorem (state which case), or else state that the Master Theorem doesn't apply. You should be able to go through these **25** recurrences in **10** minutes.

Problem 1-1. $T(n) = 3T(n/2) + n^2$

$c_{crit} = \log_2 3 \approx 1.58$ $\quad f(n) = \Omega(n^c)$ $\quad c > c_{crit}$

$f(n) = n^2$ $\quad n^2 \stackrel{\checkmark}{=} \Omega(n^c)$

$T(n) = \Theta(n^2)$ case 3

Problem 1-2. $T(n) = 7T(n/2) + n^2$

$c_{crit} = \log_2 7 \approx 2.81$ $\quad f(n) = O(n^c)$ $\quad c < c_{crit}$

$f(n) = n^2$ $\quad n^2 \stackrel{\checkmark}{=} O(n^2)$

$T(n) = \Theta(n^{2.81})$ case 1

Problem 1-3. $T(n) = 4T(n/2) + n^2$

$c_{crit} = \log_2 4 = 2$ $\quad f(n) = \Theta(n^{c_{crit}}(\log n)^k)$ $\quad k \geq 0$

$f(n) = n^2$ $\quad n^2 \stackrel{\checkmark}{=} \Theta(n^2(\log n)^0)$ $k=0$

$T(n) = \Theta(n^2(\log n)^1)$ case 2

Problem 1-4. $T(n) = 3T(n/4) + n \lg n$

$c_{crit} = \log_4(3) \approx 0.79$ $\quad f(n) = \Omega(n^c)$ $\quad c > c_{crit}$

$f(n) = n \log n$ $\quad n\log(n) \stackrel{\checkmark}{=} \Omega(n^c)$

$T(n) = \Theta(n\log(n))$ case 3

Problem 1-5. $T(n) = 4T(n/2) + \lg n$

$c_{crit} = \log_2(4) = 2$ $\quad f(n) = O(n^c)$ $\quad c < c_{crit}$

$f(n) = \log(n)$ $\quad \log n \stackrel{\checkmark}{=} O(n^c)$

$T(n) = \Theta(n^2)$ case 1

Problem 1-6. $T(n) = T(n-1) + n$

$c_{crit} = \log_1 1 =$ undef

Master Theorem doesn't apply

Problem 1-7. $T(n) = 4T(n/2) + n^2 \lg n$

$c_{crit} = \log_2 4 = 2$

$f(n) = n^2 \log(n)$

$f(n) = \Theta(n^{c_{crit}} (\log n)^k)$ $k \geq 0$

$n^2 \log n \stackrel{\checkmark}{=} \Theta(n^2 (\log n)^1)$

$\underline{T(n) = \Theta(n^2 (\log n)^2) \text{ case } 2}$

Problem 1-8. $T(n) = 5T(n/2) + n^2 \lg n$

$c_{crit} = \log_2 5 \approx 2.32$

$n^{2.32} > n^2 \Rightarrow f(n) = O(n^c)$ $c << c_{crit}$

$f(n) = n^2 \log n$

$n^2 \log(n) \stackrel{\checkmark}{=} O(n^c)$

$\underline{T(n) = O(n^{2.32}) \text{ case } 1}$

Problem 1-9. $T(n) = 3T(n/3) + n/\lg n$

$n^{\log_3 3} = n^1$

$f(n) = \frac{n}{\log(n)}$

case 1: $\frac{n}{\log n} \neq O(n^1)$

case 2 & 3 don't apply either

<u>Master Theorem doesn't apply</u>

Problem 1-10. $T(n) = 2T(n/4) + c$

$n^{\log_4 2} = c$

$n^{\frac{1}{2}} = c$

case 1: $c = O(n^c)$ $c < c_{crit}$

$c \stackrel{\checkmark}{=} O(n^0)$

$\underline{T(n) = O(n^{\frac{1}{2}}) \text{ case } 1}$

Problem 1-11. $T(n) = T(n/4) + \lg n$

$n^{\log_4 1} = \log n$

$1 < \log n$

case 2: $\log n = \Theta(n^0 (\log n)^k)$

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

$\underline{T(n) = \Theta(\log^2 n) \text{ case } 2}$

Problem 1-12. $T(n) = T(n/2) + T(n/4) + n^2$

<u>Master theorem doesn't apply</u>

Problem 1-13. $T(n) = 2T(n/4) + \lg n$

$n^{\log_4 2} = \log n$

$n^{\frac{1}{2}} > \log n$

Case 1: $\log(n) = O(n^c)$

$\underline{T(n) = \Theta(n^{\frac{1}{2}}) \text{ case } 1}$

Problem 1-14. $T(n) = 3T(n/3) + n \lg n$

$n^{\log_3 3} = n \log n$

$n < n \log n$

case 2: $n \log n = \Theta(n^1 (\log n)^k)$

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

$\underline{T(n) = \Theta(n \log^2 n) \text{ case } 2}$

Problem 1-15. $T(n) = 8T((n - \sqrt{n})/4) + n^2$

Master Theorem doesn't apply

Problem 1-16. $T(n) = 2T(n/4) + \sqrt{n}$

$n^{\log_4 2} = \sqrt{n}$ Case 2! $\sqrt{n} = \Theta(n^{\frac{1}{2}}(\log n)^k)$ $k \geq 0$

$n^{\frac{1}{2}} = \sqrt{n}$ $\sqrt{n} = \Theta(n^{\frac{1}{2}}(\log n)^0) \stackrel{\checkmark}{=} \Theta(n^{\frac{1}{2}})$ $T(n) = \Theta(n^{\frac{1}{2}}(\log n)^1)$ case 2

Problem 1-17. $T(n) = 2T(n/4) + n^{0.51}$

$n^{\log_4 2} = n^{0.51}$ case 3: $n^{0.51} \stackrel{\checkmark}{=} \Omega(n^c)$ $c > c_{crit}$

$n^{\frac{1}{2}} < n^{0.51}$ $T(n) = \Theta(n^{0.51})$ case 3

Problem 1-18. $T(n) = 16T(n/4) + n!$

$n^{\log_4 16} = n!$ Case 3! $n! \stackrel{\checkmark}{=} \Omega(n^c)$ $c > c_{crit}$

$n^2 < n!$ $T(n) = \Theta(n!)$ case 3

Problem 1-19. $T(n) = 3T(n/2) + n$

$n^{\log_2 3} = n$ case 1: $n \stackrel{\checkmark}{=} O(n^c)$ $c < c_{crit}$

$n^{1.58} > n$ $T(n) = \Theta(n^{1.58})$ case 1

Problem 1-20. $T(n) = 4T(n/2) + cn$

$n^{\log_2 4} = cn$ case 1! $cn \stackrel{\checkmark}{=} O(n^c)$ $c < c_{crit}$

$n^2 > cn$ $T(n) = \Theta(n^2)$ case 1

Problem 1-21. $T(n) = 3T(n/3) + n/2$

$n^{\log_3 3} = \frac{n}{2}$ case 2: $\frac{n}{2} = \Theta(n^1(\log n)^k)$ $k \geq 0$

$n^1 = \frac{1}{2}n$ $\frac{n}{2} \stackrel{\checkmark}{=} \Theta(n(\log n)^0)$ $T(n) = \Theta(n \log n)$ case 2

Problem 1-22. $T(n) = 4T(n/2) + n/\lg n$

$n^{\log_2 4} = \frac{n}{\log(n)}$ case 1: $\frac{n}{\log(n)} \stackrel{\checkmark}{=} O(n^c)$ $c < c_{crit}$

$n^2 > \frac{n}{\log(n)}$ $T(n) = \Theta(n^2)$ case 1

Problem 1-23. $T(n) = 7T(n/3) + n^2$

$n^{\log_3 7} = n^2$ case 3: $n^2 \stackrel{\checkmark}{=} \Omega(n^c)$ $c > c_{crit}$

$n^{1.77} < n^2$

$\underline{T(n) = \Theta(n^2) \text{ case } 3}$

Problem 1-24. $T(n) = 8T(n/3) + 2^n$

$n^{\log_3 8} = 2^n$ case 3: $2^n \stackrel{\checkmark}{=} \Omega(n^c)$ $c > c_{crit}$

$n^{1.89} < 2^n$

$\underline{T(n) = \Theta(2^n) \text{ case } 3}$

Problem 1-25. $T(n) = 16T(n/4) + n$

$n^{\log_4 16} = n$ case 1: $n \stackrel{\checkmark}{=} O(n^c)$ $c < c_{crit}$

$n^2 > n$

$\underline{T(n) = \Theta(n^2) \text{ case } 1}$