CIS507: Design & Analysis of Algorithms
Tutorial: Asymptotics and Recurrences

Cheat Sheet: Master Method

For $T(n) = aT(n/b) + f(n)$, with $a \geq 1$, $b \geq 1$, compare $f(n)$ with $n^{\log_b a}$.

Case	Condition, for $\epsilon > 0$	Solution
1	$f(n) = \mathcal{O}(n^{\log_b a - \epsilon})$	$T(n) = \Theta(n^{\log_b a})$ Number of leafs dominates
2	$f(n) = \Theta(n^{\log_b a})$	$T(n) = \Theta(n^{\log_b a} \lg n)$ All rows have same asymptotic sum
3	$f(n) = \Omega(n^{\log_b a + \epsilon})$	$T(n) = \Theta(f(n))$ provided that $af(n/b) \leq cf(n)$ for some $c < 1$
2 (general)	$f(n) = \Theta(n^{\log_b a} \lg^k n)$ or some constant $k \geq 0$	$T(n) = \Theta(n^{\log_b a} \lg^{k+1} n)$ They grow at 'similar' rate

Note that the last row includes the more general version of case 2, which isn't in the textbook.

1 Asymptotics (Textbook Exercise 3.1-1)

Let $f(n)$ and $g(n)$ be asymptotically nonnegative functions. Using the basic definition of Θ-notation, prove that $\max(f(n), g(n)) = \Theta(f(n) + g(n))$.

2 Asymptotics (Textbook Exercise 3.1-2)

Show that for any real constants a and b, where $b > 0$:
$(n + a)^b = \Theta(n^b)$.

3 Math (Textbook Exercise 3.2-2)

Prove that $a^{\log_b c} = c^{\log_b a}$.

4 Asymptotics (Textbook Exercise 3.1-4)

Is $2^{n+1} = \mathcal{O}(2^n)$? Is $2^{2n} = \mathcal{O}(2^n)$? Explain.

5 Asymptotics

Rank the following functions by increasing order of growth. That is, find any arrangement g_1; g_2; g_3; g_4; g_5; g_6; g_7 of the functions satisfying $g_1 = \mathcal{O}(g_2)$, $g_2 = \mathcal{O}(g_3)$, $g_3 = \mathcal{O}(g_4)$, $g_4 = \mathcal{O}(g_5)$, $g_5 = \mathcal{O}(g_6)$, $g_6 = \mathcal{O}(g_7)$.

- $f_1(n) = n^4 + \log n$
- $f_2(n) = n + \log^4 n$ (note that $\log^4 n$ is shorthand for $(\log n)^4$)
- $f_3(n) = n \log n$
- $f_4(n) = \binom{n}{3}$
- $f_5(n) = \binom{n}{n/2}$
- $f_6(n) = 2^n$
- $f_7(n) = n^{\log n}$

6 Substitution (Textbook Exercise 4.3-1)

Show that the solution of $T(n) = T(n-1) + n$ is $\mathcal{O}(n^2)$. Use the substitution method.

7 Substitution (Textbook Exercise 4.3-7)

Using the master method, you can show that the solution to the recurrence $T(n) = 4T(n/3) + n$ is $T(n) = \Theta(n^{\log_3 4})$. Show that a substitution proof with the assumption $T(n) \leq cn^{log_3 4}$ fails. Then show how to subtract off a lower-order term to make a substitution proof work.

8 Master Method

Find an asymptotic solution of the following functional recurrences.

1. $T(n) = 9T(n/3) + n^3$
2. $T(n) = 2T(n/4) + \sqrt{n}$
3. $T(n) = 3T(n/4) + n \lg n$
4. $T(n) = 4T(n/2) + n^2 \log n$

9 Trick of Changing the Variable

Solve the recurrence $T(n) = 2T(\sqrt{n}) + 1$ using the master method. *Hint:* To make this possible, you can define your own variables and a new recurrence function to make the recurrence look like the form in the master method.