Justin Ciocoi

Oct 3, 2023

CSCI 377 Class Notes

Computer Algorithms

Recurrence of Dividing Functions

Review of Decreasing Function Recurrence

- Three methods to solve:
    - Recursion Trees
        - Break down function until you reach base case outlined in function definition
    - Substitution Method
        - Substitute in values into original $T(n)$ expression
    - Master Theorem
        - *For the General Form*
            - $T(n) = a * T(n-b) + f(n)$
            - where,
                - $a > 0, b > 0$, and $f(n) = \theta(n^d)$
        - Case 1: $a < 1$
            - $T(n) = O(n^d)$
        - Case 2: $a > 1$
            - $T(n) = O(n^k * a^{n/b})$
        - Case 3: $a = 1$
            - $T(n) = O(n * f(n))$

Dividing Functions

- Must always have your answer, $T(n)$ as a function of n
    - For example, if $\frac{n}{2^k} = 1$, and $T(n) = k$
    - We know, $n = 2^k$, so $k = log(n)$
    - Finally, $T(n) = log(n)$

Example

- $T(n) = 1$ if $n = 1$
- $T(n) = T(\frac{n}{2}) + n$ if $n > 1$
    - Goes from $T(n)$ to $T(\frac{n}{2^k})$
        - Result is:

$$T(n) = n + \frac{n}{2} + \frac{n}{2^2} + \frac{n}{2^3} + ... + \frac{n}{2^{k-1}} + \frac{n}{2^k}$$
$$T(n) = n(1 + (\frac{1}{2} + \frac{1}{2^2} + ... + \frac{1}{2^k}))$$
$$T(n) = n(1 + \sum_{i=0}^{k} \frac{1}{2^i})$$
$$T(n) = n(1 + 1) = 2n = O(n)$$

Master Theorem for Dividing Functions

- *For the General Form*
    - $T(n) = a * T(\frac{n}{b}) + \theta(n^d)$
    - where
        - $a > 0, b > 1, d \geq 0$ (all constant)
- Case 1: $d < \log_b(a)$
    - $T(n) = \theta(n^{log_b(a)})$
- Case 2: $d > \log_b(a)$
    - $T(n) = \theta(n^d)$
- Case 3: $d = \log_b(a)$
    - $T(n) = \theta(n^{log_b(a)} log(n)) = \theta(n^d log(n))$

Example 1

$$T(n) = 4T(\frac{n}{2}) + \theta(n^1)$$
$$a = 4, b = 2, d = 1$$
$$d < \log_b(a)$$
$$T(n) = \theta(n^{log_2(4)}) = \theta(n^2)$$

Example 2

$$T(n) = 2T(\frac{n}{2}) + \theta(n^1)$$
$$a = 2, b = 2, d = 1$$
$$\log_b(a) = \log_2(2) = 1$$
$$d = \log_b(a)$$

$$T(n) = \theta(n \log(n))$$

Example 3

$$T(n) = 2T(\frac{n}{2}) + \theta(n^2)$$
$$a = 2, b = 2, d = 2$$
$$\log_b(a) = \log_2(2) = 1$$
$$d > \log_b(a)$$
$$T(n) = \theta(n^2)$$