

Recurrence Relations

CPE348

Theresa Migler-VonDollen

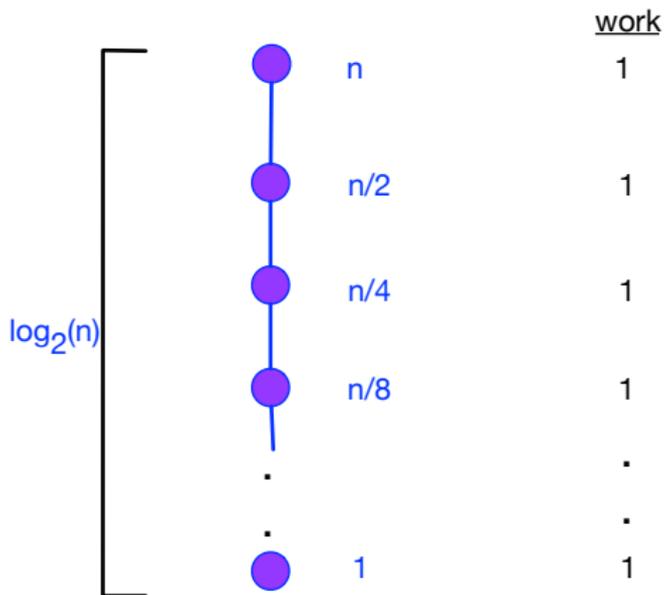
Recurrence Relations

You have two main choices when it comes to solving recurrence relations:

- The tree method (my favorite)
- The Master Theorem
 - If $T(n) = aT(\lceil n/b \rceil) + O(n^d)$ for $a > 0, b > 1, d \geq 0$ then:
 - $T(n) = O(n^d)$ if $d > \log_b a$
 - $T(n) = O(n^d \log n)$ if $d = \log_b a$
 - $T(n) = O(n^{\log_b a})$ if $d < \log_b a$

Recurrence Relations - Binary Search

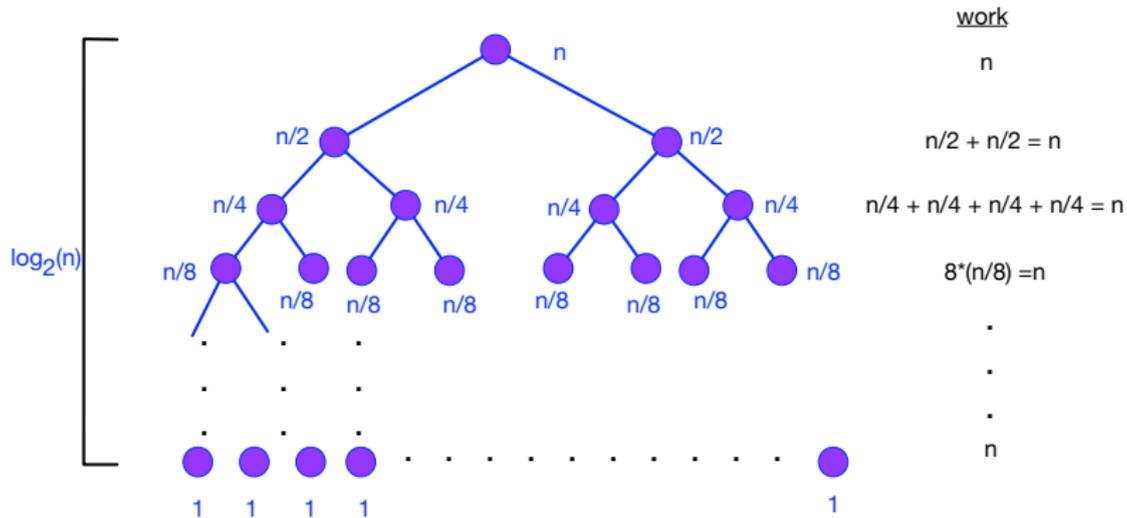
$$T(n) = T(n/2) + O(1)$$



$$= O(1 + 1 + \dots + 1) = O(\log_2(n))$$

Recurrence Relations - Merge Sort

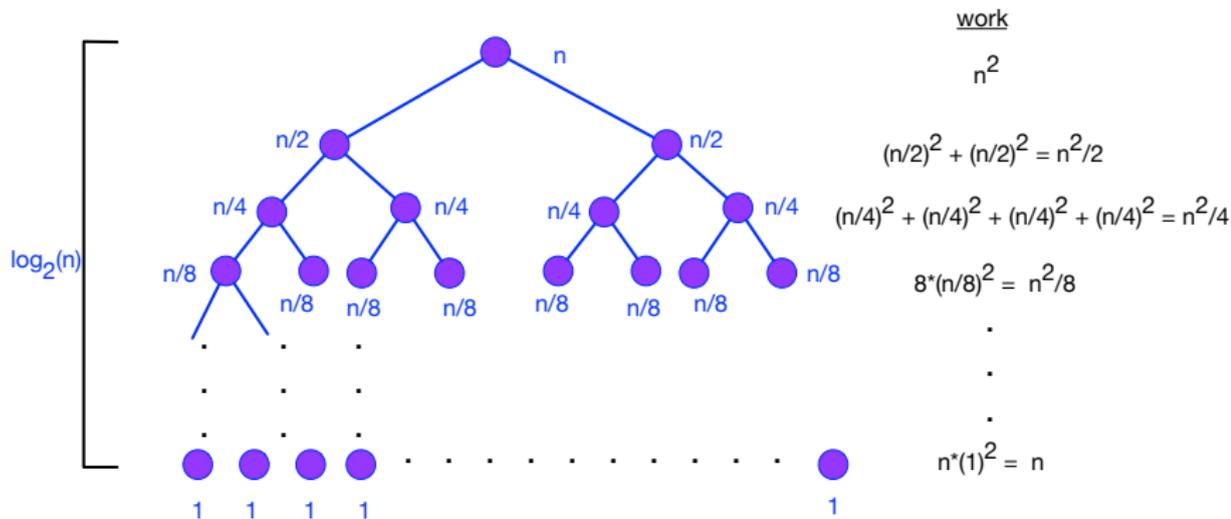
$$T(n) = 2T(n/2) + O(n)$$



$$= O(n + n + n + \dots + n) = O(n \log_2(n))$$

Recurrence Relations - More Practice

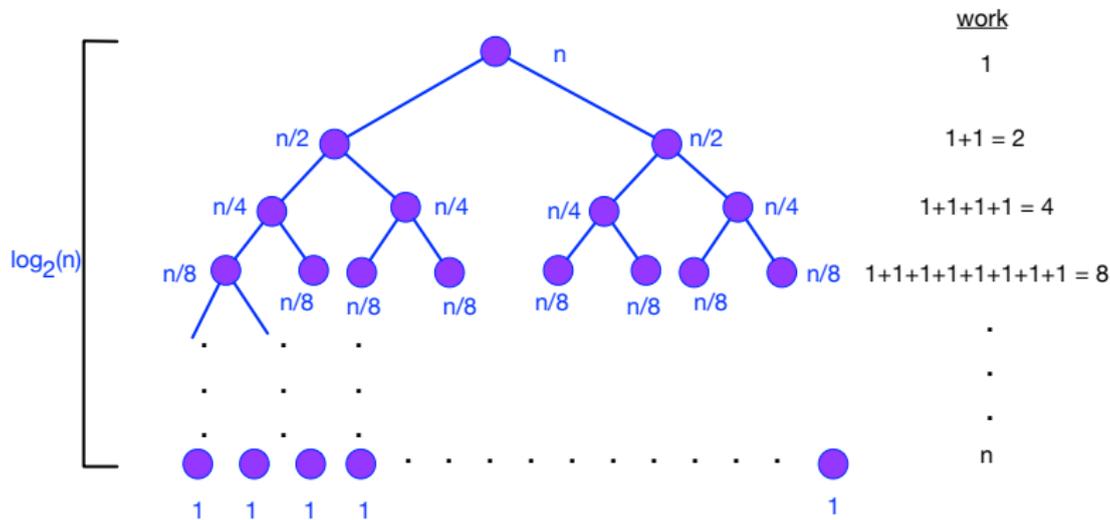
$$T(n) = 2T(n/2) + O(n^2)$$



$$= O(n^2 + n^2/2 + n^2/4 + n^2/8 \cdots + n^2/n) \leq O(2n^2) = O(n^2)$$

Recurrence Relations - More Practice

$$T(n) = 2T(n/2) + O(1)$$

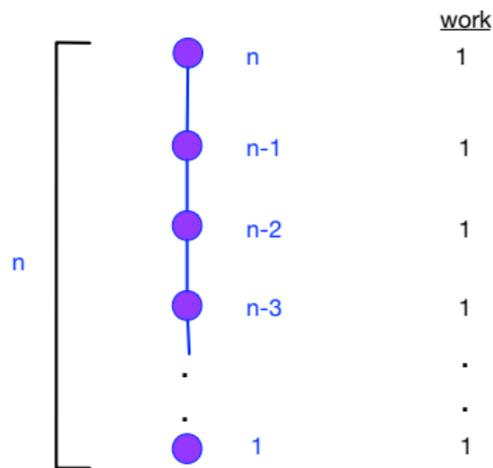


$$= O(1 + 2 + 4 + \dots + n) =$$

$$O(n + n/2 + n/4 + n/8 + \dots + n/2^{\log_2 n}) \leq O(2n) = O(n)$$

Recurrence Relations - Reduce by One

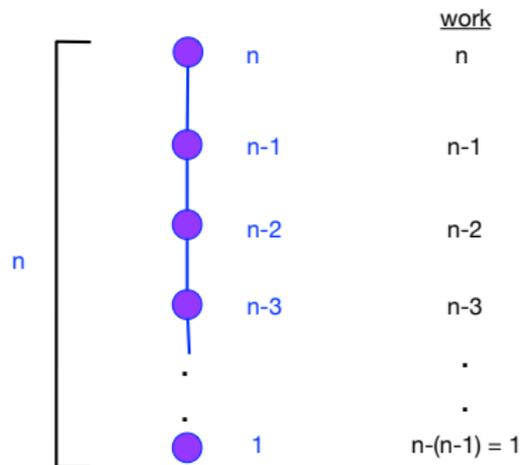
$$T(n) = T(n-1) + O(1)$$



$$= O(1 + 1 + 1 + \dots + 1) = O(n)$$

Recurrence Relations - Reduce by One

$$T(n) = T(n-1) + O(n)$$



$$= O(n + (n-1) + (n-2) + (n-3) + \dots + (n - (n-2)) + (n - (n-1))) \approx O(n^2/2) = O(n^2)$$