Recurrences and solving them

9/16/24

Administrivia

- Due tomorrow:
  - HW 0: Submit clicker ID
  - HW 1: Array with fast initialization, asymptotic ordering, induction
- Expect reading for Wed/Thurs

Recall: Divide and conquer

- Algorithmic technique of splitting problem into smaller instances of the same problem, solving them recursively, and combining the results

- Examples: mergesort, quicksort, binary search

Recall: Recurrences

Representing running time w/ recursive expression

Ex: Mergesort

Let $T(n)$ be the time to sort n elements

The algorithm requires

- 2 calls to sort n/2 elements $\quad 2T(n/2)$
- The merge (and other stuff) $\quad n$

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

What is the recurrence for the following sorting algorithm?

Recursively sort the first 2/3rds
Recursively sort the last 2/3rds
Recursively sort the first 2/3rds

A. $T(n) = T(2n/3) + n$
B. $T(n) = T(3n) + 2n/3$
C. $T(n) = 3T(2n/3) + n$
D. $T(n) = 3T(n) + (2/3)$
E. None of the above

What is the recurrence for the following sorting algorithm?

> Recursively sort the first 2/3rds
> Recursively sort the last 2/3rds
> Recursively sort the first 2/3rds

A. $T(n) = T(2n/3) + n$

B. $T(n) = T(3n) + 2n/3$

C. $T(n) = 3T(2n/3) + n$

D. $T(n) = 3T(n) + (2/3)$

E. <u>None of the above</u> <u>$T(n) = 3T(2n/3) + 1$</u>

Solving recurrences: recursion trees

Solving recurrences: recursion trees

Level i: 2^i tasks perform work $n/2^i$ each

Total work per level: n

$\log_2 n = O(\log n)$ levels until base case

Total work: $O(n \log n)$

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

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

Level i: 1 task with work $n/2^i$

Total work: $n + n/2 + n/2^2 + n/2^3 + \ldots + n/2^{\log n}$

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

Level i: 1 task with work $n/2^i$

Total work: $n + n/2 + n/2^2 + n/2^3 + \ldots + n/2^{\log n}$

$= O(n)$

$1 + r + r^2 + r^3 + r^4 + \ldots = 1/(1-r)$

when $r < 1$

Solving $T(n) = 3T(n/2) + n$

Approach to solving a recurrence

- Draw recursion tree
- Determine the non-recursive work per level (often, find #tasks and work/task)
- Sum to get total work (often, find #levels)

- Very useful identity:

  $$1 + r + r^2 + r^3 + r^4 + \ldots = 1/(1-r) \text{ when } r < 1$$
- Other trick: switch order of series

Try it yourself!

Solve each of the following recurrences

a) $T(n) = T(n/3) + 1$

b) $T(n) = 3T(n/4) + n$

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

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