Lecture16

CSE 417
Algorithms and Complexity

Winter 2023
Lecture 16
Divide and Conquer and Recurrences

2/13/2023 CSE 417 1

Divide and Conquer

- Recurrences, Sections 5.1 and 5.2
- Algorithms
  - Median (Selection)
  - Fast Matrix Multiplication
  - Counting Inversions (5.3)
  - Multiplication (5.5)

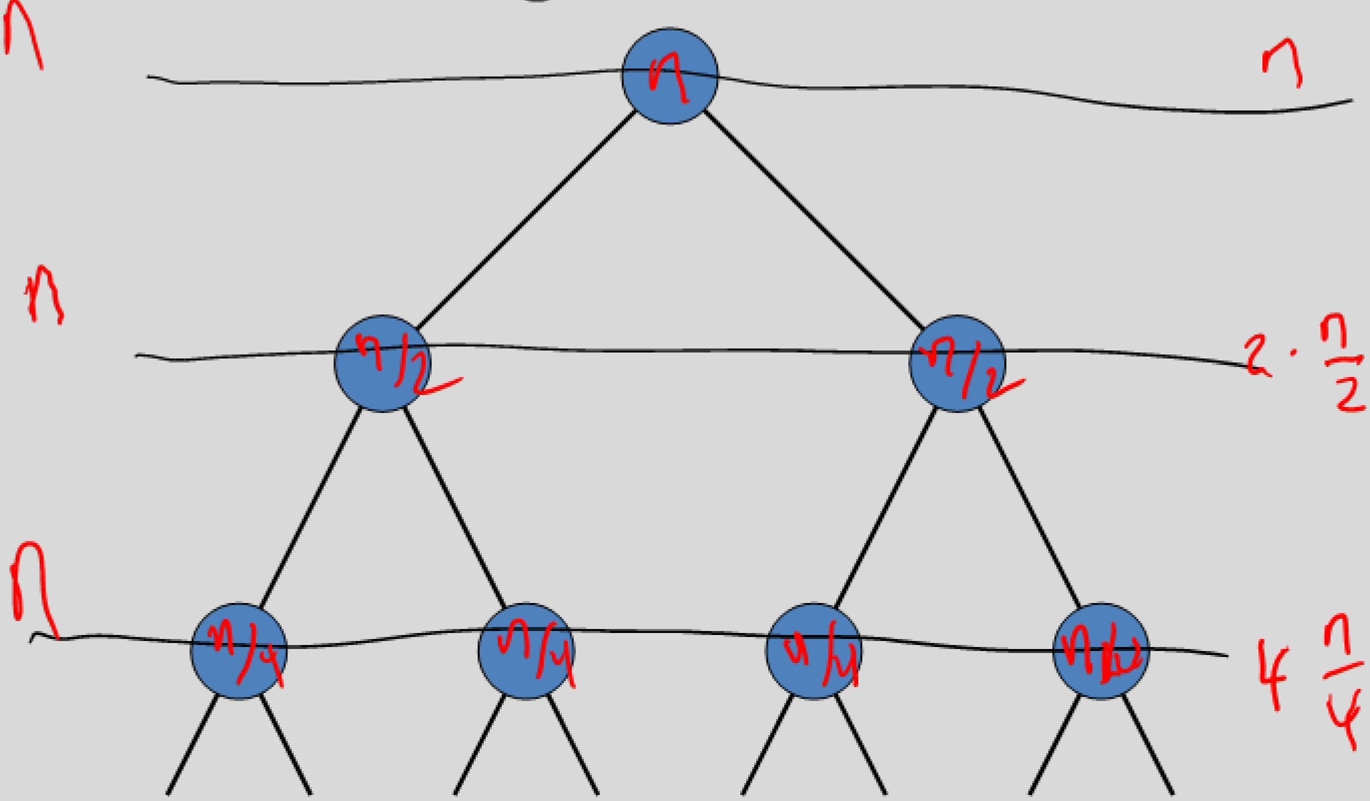
Divide and Conquer : Merge Sort

```
Array MSort(Array a, int n){
   if (n <= 1)  return a;
   return Merge(MSort(a[0 .. n/2], n/2),  MSort(a[n/2+1 .. n-1], n/2);
}
```

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

2/13/2023 CSE 417 3

Unrolling the recurrence



2/13/2023 CSE 417 4

A better mergesort (?)

3-merge $O(n)$

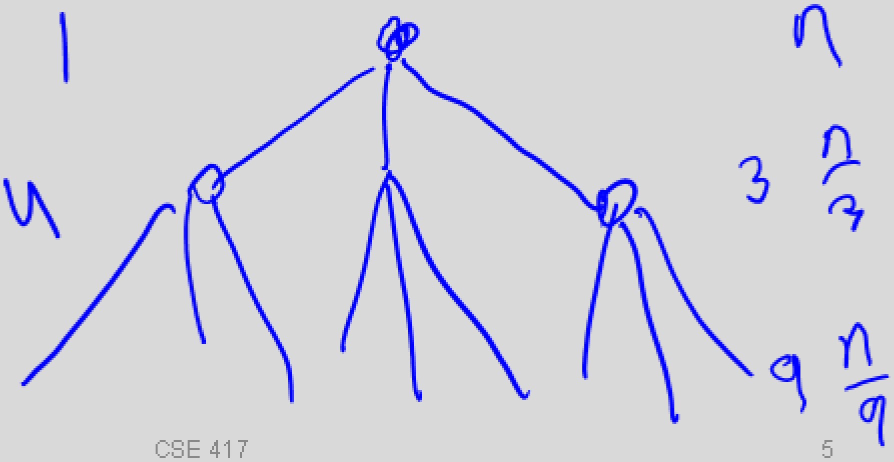
- Divide into 3 subarrays and recursively sort
- Apply 3-way merge

$n \log_3 n$

$$T(n) = 3T\left(\frac{n}{3}\right) + n$$

$$T(1) = 1$$

$ht = \log_3 n$



What is the recurrence?

Unroll recurrence for $T(n) = 3T(n/3) + n$

$T(n) = aT(n/b) + f(n)$

$T(n) = aT(n/b) + n^c$

Master Theorem

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

Where does this recurrence arise?

T(n) = games in tournament.

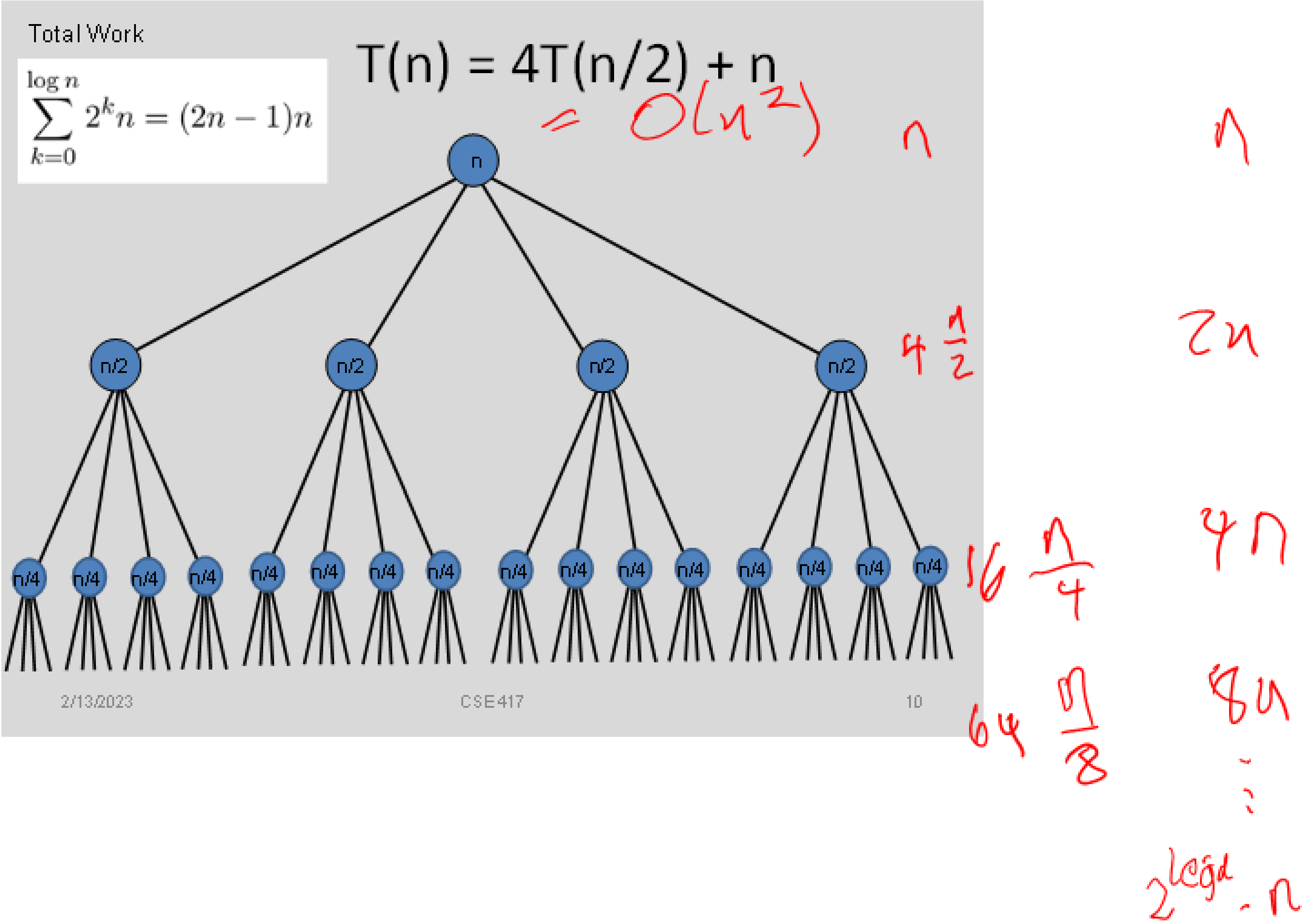
$$T(n) = \frac{n}{2} + T\left(\frac{n}{2}\right)$$

$$T(2) = 1, \quad T(1) = 0$$

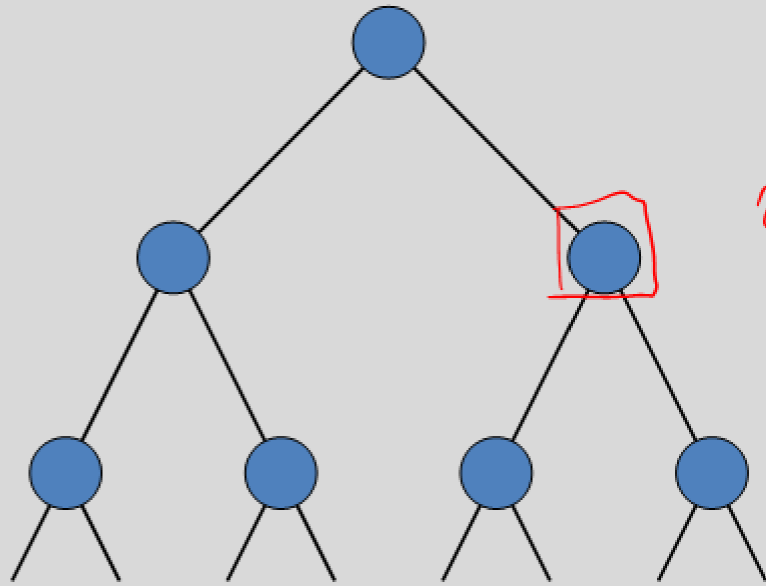
Solving the recurrence exactly

$$\sum_{i=0}^{n} x^i = \frac{x^{n+1}-1}{x-1} = \frac{1-x^{n+1}}{1-x}$$

$$T(n) = T\left(\frac{n}{2}\right) + n$$
$$= T\left(\frac{n}{4}\right) + \frac{n}{2} + n$$
$$= T\left(\frac{n}{8}\right) + \frac{n}{4} + \frac{n}{2} + n$$
$$\vdots$$
$$= 1 + 2 + 4 + \cdots + \frac{n}{4} + \frac{n}{2} + n$$
$$= 2n - 1$$


Total Work
$\sum_{k=0}^{\log n} 2^k n = (2n-1)n$
T(n) = 4T(n/2) + n
= O(n^2)
n
n/2
n/4
2/13/2023
CSE417
10
n
4 n/2
16 n/4
64 n/8
n
2n
4n
8n
2^{log d} · n

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



n^2 $\quad n^2$

$2\left(\frac{n}{2}\right)^2$ $\quad \frac{1}{2} n^2$

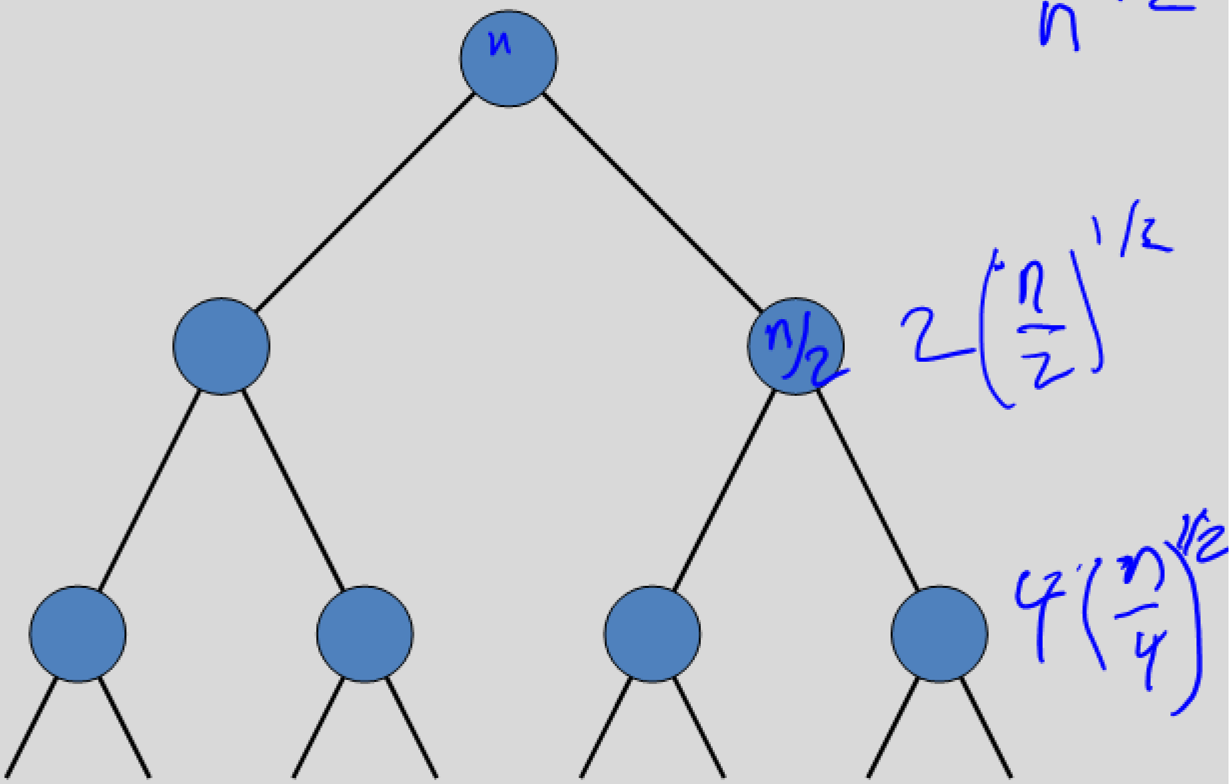
$4\left(\frac{n}{4}\right)^2$ $\quad \frac{1}{4} n^2$

$8\left(\frac{n}{8}\right)^2$ $\quad \frac{1}{8} n^2$

2/13/2023 CSE417 11

$$n^2\left(1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \cdots\right) \le 2n^2$$

$T(n) = 2T(n/2) + n^{1/2}$



Recurrences

- Three basic behaviors
  - Dominated by initial case
  - Dominated by base case
  - All cases equal – we care about the depth

What you really need to know about recurrences

- Work per level changes geometrically with the level
- Geometrically increasing ($x > 1$)
  - The bottom level wins
- Geometrically decreasing ($x < 1$)
  - The top level wins
- Balanced ($x = 1$)
  - Equal contribution

Classify the following recurrences (Increasing, Decreasing, Balanced)

- $T(n) = n + 5T(n/8)$ — n $\frac{5}{8}n$ decr
- $T(n) = n + 9T(n/8)$ — n $\frac{9}{8}n$ inc
- $T(n) = n^2 + 4T(n/2)$ — n^2 $4 \cdot \frac{n^2}{4}$ =
- $T(n) = n^3 + 7T(n/2)$ — n^3 $7\frac{n^3}{8}$ decr
- $T(n) = n^{1/2} + 3T(n/4)$ — $n^{1/2}$ $3\frac{n^{1/2}}{4^{1/2}}$ inc

Recursive Matrix Multiplication

Multiply 2 x 2 Matrices:

$$\begin{vmatrix} r & s \\ t & u \end{vmatrix} = \begin{vmatrix} a & b \\ c & d \end{vmatrix} \begin{vmatrix} e & g \\ f & h \end{vmatrix}$$

r = ae + bf

s = ag + bh

t = ce + df

u = cg + dh

A N x N matrix can be viewed as a 2 x 2 matrix with entries that are (N/2) x (N/2) matrices.

The recursive matrix multiplication algorithm recursively multiplies the (N/2) x (N/2) matrices and combines them using the equations for multiplying 2 x 2 matrices

Recursive Matrix Multiplication

- How many recursive calls are made at each level?
- How much work in combining the results?
- What is the recurrence?

What is the run time for the recursive Matrix Multiplication Algorithm?

- Recurrence:

Strassen's Algorithm

Multiply 2 x 2 Matrices:

$$\begin{vmatrix} r & s \\ t & u \end{vmatrix} = \begin{vmatrix} a & b \\ c & d \end{vmatrix} \begin{vmatrix} e & g \\ f & h \end{vmatrix}$$

$r = p_1 + p_2 - p_4 + p_6$

$s = p_4 + p_5$

$t = p_6 + p_7$

$u = p_2 - p_3 + p_5 - p_7$

Where:

$p_1 = (b - d)(f + h)$

$p_2 = (a + d)(e + h)$

$p_3 = (a - c)(e + g)$

$p_4 = (a + b)h$

$p_5 = a(g - h)$

$p_6 = d(f - e)$

$p_7 = (c + d)e$

From AHU 1974

Recurrence for Strassen's Algorithms

- $T(n) = 7\,T(n/2) + cn^2$
- What is the runtime?

$\log_2 7 = 2.8073549221$

BFPRT Recurrence

$$T(n) \le T(3n/4) + T(n/5) + 20\,n$$

What bound do you expect?