Announcements

- HW7 solution will be released tomorrow
- sample prelim question solutions will be released Saturday
- Prelim 2: Tuesday March 24th
  - Topics: stable matching, flows and applications and NP-completeness
  - Information sheet on topics and sample question in canvas

- Section next week prelim review: it is optional, and feel free to attend the section of your choice

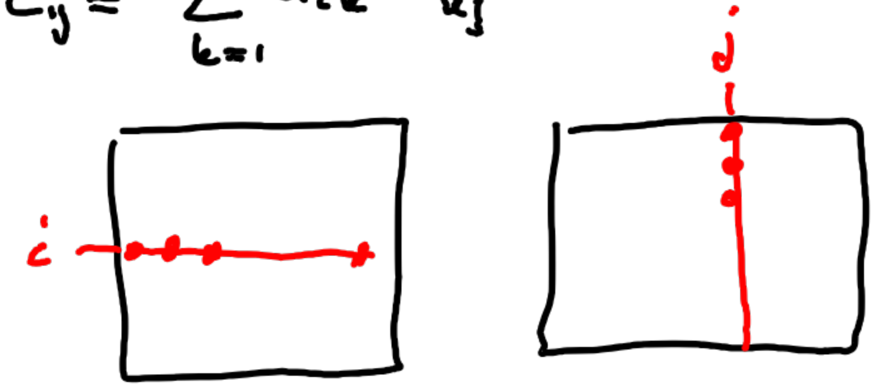
- HW8 will be divide and conquer, due Friday April 10

Divide and Conquer: Matrix multiplication

$$A = \begin{array}{cc} & \begin{array}{ccc} & j & \end{array} \\ i & \begin{bmatrix} a_{11} & & a_{1n} \\ & \boxed{} & \\ a_{n1} & & a_{nn} \end{bmatrix} \end{array}$$

n by n   entry in row i column j a_{ij}

A, B $n \times n$ matrices $A \cdot B = C$ $c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$



time to compute:

result n by n c_{ij} $i, j = 1, \dots, n$

$O(n)$ to compute each

$O(n^3)$ time

c_{ij}

goal today: can we do better?

Note $A + B$ result $= D$ $d_{ij} = a_{ij} + b_{ij}$

total time $O(n^2)$

Divide and Conquer for Matrix Multiplication

$$A \quad \begin{array}{cc} n/2 & n/2 \end{array}$$
$$\begin{array}{c} \\ \end{array}\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{array}{c} n/2 \\ n/2 \end{array}$$

A_{lk} is $n/2 \times n/2$ matrix
$l,k \in \{1,2\}$
B_{lk} is $n/2 \times n/2$ matrix

$$B \quad \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}$$

assume n even

$$c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$$

$$C = AB$$

$$\begin{bmatrix} A_{11}B_{11} + A_{12}B_{21} & A_{11}B_{12} + A_{12}B_{22} \\ A_{21}B_{11} + A_{22}B_{21} & A_{21}B_{12} + A_{22}B_{22} \end{bmatrix}$$

Resulting recurrence $T(n)$ time $n \times n$ matrix multiply (n power of 2)

$$T(n) = 8\,T(n/2) + c n^2$$

c constant

What is the running time of the recursive matrix multiplication defined above on $n \times n$ matrices

A. $O(n \log n)$
B. $O(n^2)$
C. $O(n^2 \log n)$
D. $O(n^{2.5})$
E. $O(n^3)$
F. None of these

time spent level 0

$cn^2 + 8c\cdot\left(\frac{n}{2}\right)^2 + 8^2 c\cdot\left(\frac{n}{4}\right)^2 + \dots$

$\sum_{i=0}^{\log_2 n} 8^i \cdot c \cdot \left(\frac{n}{2^i}\right)^2 = c \sum_{i=1}^{\log_2 n} 8^i \frac{n^2}{2^{2i}}$

$= cn^2 \cdot \sum_{i=1}^{\log_2 n} \frac{8^i}{4^i} = cn^2 \sum_{i=1}^{\log_2 n} 2^i$

$= c\cdot n^2 \left(\underbrace{2^{\log_2 n+1}}_{\approx 2n} - 1\right) = O(n^3)$

#	size
1	$n \times n$
8	$n/2 \times n/2$
⋮	
i … 8^i	$n/2^i \times n/2^i$

levels $\log_2 n$

Faster: Recursive Matrix Multiplication (Strassen)

- $M_1 = (A_{11} + A_{22})(B_{11} + B_{22})$
- $M_2 = (A_{21} + A_{22})B_{11}$
- $M_3 = A_{11}(B_{12} - B_{22})$
- $M_4 = A_{22}(B_{21} - B_{11})$
- $M_5 = (A_{11} + A_{12})B_{22}$
- $M_6 = (A_{21} - A_{11})(B_{11} + B_{12})$
- $M_7 = (A_{12} - A_{22})(B_{21} + B_{22})$

- $C_{11} = M_1 + M_4 - M_5 + M_7$
- $C_{12} = M_3 + M_5$
- $C_{21} = M_2 + M_4$
- $C_{22} = M_1 - M_2 + M_3 + M_6$

$$A_{11}B_{12} - \cancel{A_{11}B_{22}}$$
$$+ \cancel{A_{11}B_{22}} + A_{12}B_{22}$$

$$T(n) = 7 \cdot T\left(\frac{n}{2}\right) + c \cdot n^2$$

$$\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}$$

$$C_{12} = A_{11}B_{12} + A_{12}B_{22}$$

Analyzing Strassen's algorithm's running time

level $\qquad 7^i$ problems $\left(\frac{n}{2^i} \times n/2^i\right)$ matrix $\qquad$ # levels $\log_2 n$

$i = 0, 1, 2$

$$\sum_{i=0}^{\log_2 n} 7^i \, c \cdot \left(\frac{n}{2^i}\right)^2 = c \cdot n^2 \sum_{i=0}^{\log_2 n} \left(\frac{7}{4}\right)^i = O\left(n^2 \cdot \left(\frac{7}{4}\right)^{\log_2 n}\right)$$

$$= O\left(\cancel{n^2} \, \frac{7^{\log_2 n}}{\cancel{4^{\log_2 n}}}\right) =$$

$$= O\left(7^{\log_2 n}\right) = O\left(n^{\log_2 7}\right)$$

$$\sim n^{2.81}$$

Recall

$$\sum_{i=0}^{k} q^i = \frac{q^{k+1} - 1}{q - 1} = \frac{q \cdot q^k - 1}{q - 1}$$

only term with n in it

also $q^{\log_2 n} = n^{\log_2 q}$

Master Theorem for solving divide and conquer recurrences

$q \,?\, 2^\alpha$

Solving $T(n) = q\,T(n/2) + c \cdot n^\alpha$ c & α constants

time $\displaystyle\sum_{i=0}^{\log_2 n} q^i c \cdot \left(\frac{n}{2^i}\right)^\alpha = c n^\alpha \sum_{i=0}^{\log_2 n} \left(\frac{q}{2^\alpha}\right)^i$

level i: q^i problems

size $n/2^i$

time on each

$(n/2^i)^\alpha$

levels $\log_2 n$

Cases: 1: $q > 2^\alpha \Rightarrow$ sum $O\left(\dfrac{q^{\log_2 n}}{\underbrace{2^{\alpha \log_2 n}}_{n^\alpha}}\right)$

overall $O(q^{\log_2 n}) = O(n^{\log_2 q})$

Case 2: $q = 2^\alpha \Rightarrow O(n^\alpha \log_2 n)$

Case 3: $q < 2^\alpha = O(n^\alpha)$

$x < 1$

hint: $x + x^2 + \dots x^k \dots = \dfrac{1}{1-x}$