# MATRIX MULTIPLICATION

**Input:** $A = [a_{ij}],\ B = [b_{ij}].$

**Output:** $C = [c_{ij}] = A \cdot B.$

$i, j = 1, 2, \ldots, n.$

$$\begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{nn} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \cdot \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{nn} \end{bmatrix}$$

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

# Standard algorithm

**for** $i \leftarrow 1$ **to** $n$
    **do for** $j \leftarrow 1$ **to** $n$
        **do** $c_{ij} \leftarrow 0$
            **for** $k \leftarrow 1$ **to** $n$
                **do** $c_{ij} \leftarrow c_{ij} + a_{ik} \cdot b_{kj}$

GALGOTIAS UNIVERSITY

# Standard algorithm

**for** $i \leftarrow 1$ **to** $n$
  **do for** $j \leftarrow 1$ **to** $n$
    **do** $c_{ij} \leftarrow 0$
      **for** $k \leftarrow 1$ **to** $n$
        **do** $c_{ij} \leftarrow c_{ij} + a_{ik} \cdot b_{kj}$

Running time $= \Theta(n^3)$

# Divide-and-conquer algorithm

**IDEA:**

$n \times n$ matrix $=$ $2\times2$ matrix of $(n/2)\times(n/2)$ submatrices:

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

$$C \quad = \quad A \quad \cdot \quad B$$

$$\left.\begin{aligned} r &= ae + bg \\ s &= af + bh \\ t &= ce + dg \\ u &= cf + dh \end{aligned}\right\}$$

8 mults of $(n/2)\times(n/2)$ submatrices
4 adds of $(n/2)\times(n/2)$ submatrices

# Divide-and-conquer algorithm

**IDEA:**

$n \times n$ matrix = $2 \times 2$ matrix of $(n/2) \times (n/2)$ submatrices:

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

$$C = A \cdot B$$

$$r = ae + bg$$
$$s = af + bh$$
$$t = ce + dh$$
$$u = cf + dg$$

*recursive*

8 mults of $(n/2) \times (n/2)$ submatrices

4 adds of $(n/2) \times (n/2)$ submatrices

# Analysis of D&C algorithm

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

*# submatrices*

*submatrix size*

*work adding submatrices*

# Analysis of D&C algorithm

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

*# submatrices* (the 8)

*submatrix size* ($n/2$)

*work adding submatrices* ($\Theta(n^2)$)

$n^{\log_b a} = n^{\log_2 8} = n^3 \;\Rightarrow\;$ CASE 1 $\;\Rightarrow\; T(n) = \Theta(n^3).$

GALGOTIAS UNIVERSITY

# Analysis of D&C algorithm

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

*# submatrices*

*submatrix size*

*work adding submatrices*

$n^{\log_b a} = n^{\log_2 8} = n^3 \Rightarrow$ CASE 1 $\Rightarrow T(n) = \Theta(n^3)$.

***No better than the ordinary algorithm.***

GALGOTIAS UNIVERSITY

# Thank You