Análise de Algoritmos
Aula 11

Prof. Murilo V. G. da Silva

DINF/UFPR

(material da disciplina: André Guedes, Renato Carmo, Murilo da Silva)

Multiplicação de Matrizes Quadradas

Problema: Multiplicação de Matrizes Quadradas (MMQ)

- Duas matrizes quadradas X e Y com n linhas/colunas.
- Obter A matriz $X \times Y$.

Multiplicação de Matrizes Quadradas

Problema: Multiplicação de Matrizes Quadradas (MMQ)

- Duas matrizes quadradas X e Y com n linhas/colunas.
- Obter A matriz $X \times Y$.

Elemento na linha i e coluna j de X: $X[i, j]$.

Multiplicação de Matrizes Quadradas

Problema: Multiplicação de Matrizes Quadradas (MMQ)

- Duas matrizes quadradas X e Y com n linhas/colunas.
- Obter A matriz $X \times Y$.

Elemento na linha i e coluna j de X: $X[i, j]$.

Computando $R = X \times Y$:

Multiplicação de Matrizes Quadradas

Problema: Multiplicação de Matrizes Quadradas (MMQ)

- Duas matrizes quadradas X e Y com n linhas/colunas.
- Obter A matriz $X \times Y$.

Elemento na linha i e coluna j de X: $X[i,j]$.

Computando $R = X \times Y$: Cada $R[i,j]$: $1 \leq i,j \leq n$ é o produto interno da linha i de X com a coluna j de Y:

Multiplicação de Matrizes Quadradas

Problema: Multiplicação de Matrizes Quadradas (MMQ)

- Duas matrizes quadradas X e Y com n linhas/colunas.
- Obter A matriz $X \times Y$.

Elemento na linha i e coluna j de X: $X[i,j]$.

Computando $R = X \times Y$: Cada $R[i,j]$: $1 \leq i,j \leq n$ é o produto interno da linha i de X com a coluna j de Y:

$$R[i,j] = \sum_{k=1}^{n} X[i,k]Y[k,j], \text{ para todo } 1 \leq i,j \leq n,$$

Multiplicação de Matrizes Quadradas

Problema: Multiplicação de Matrizes Quadradas (MMQ)

- Duas matrizes quadradas X e Y com n linhas/colunas.
- Obter A matriz $X \times Y$.

Elemento na linha i e coluna j de X: $X[i,j]$.

Computando $R = X \times Y$: Cada $R[i,j]$: $1 \leq i,j \leq n$ é o produto interno da linha i de X com a coluna j de Y:

$$R[i,j] = \sum_{k=1}^{n} X[i,k]Y[k,j], \text{ para todo } 1 \leq i,j \leq n,$$

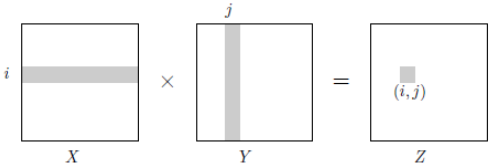

Figura: Algorithms, p. 62

Algoritmo visto no ensino médio:

Multiplicação de Matrizes Quadradas

Algoritmo visto no ensino médio:

Quanto custa?

Algoritmo visto no ensino médio:

Quanto custa?

Teorema 57

O algoritmo do ensino médio para MMQ toma tempo $\Theta(n^3)$, onde $n = \dim(X)$.

Multiplicação de Matrizes Quadradas

Algoritmo visto no ensino médio:

Quanto custa?

Teorema 57

O algoritmo do ensino médio para MMQ toma tempo $\Theta(n^3)$, onde $n = \dim(X)$.

É possível fazer melhor?

Algoritmo visto no ensino médio:

Quanto custa?

Teorema 57

O algoritmo do ensino médio para MMQ toma tempo $\Theta(n^3)$, onde $n = \dim(X)$.

É possível fazer melhor?

Teorema 58

Todo algoritmo para MMQ toma tempo $\Omega(n^2)$, onde $n = \dim(X)$.

Algoritmo Recursivo

Seja X uma matriz quadrada de ordem n com n par.

Algoritmo Recursivo

Seja X uma matriz quadrada de ordem n com n par.

- Seja $X_{1,1}, X_{1,2}, X_{2,1}, X_{2,2}$ as submatrizes de X de ordem $n/2$ conforme:

$$X = \begin{bmatrix} X_{1,1} & X_{1,2} \\ X_{2,1} & X_{2,2} \end{bmatrix}$$

Algoritmo Recursivo

Seja X uma matriz quadrada de ordem n com n par.

- Seja $X_{1,1}, X_{1,2}, X_{2,1}, X_{2,2}$ as submatrizes de X de ordem $n/2$ conforme:

$$X = \begin{bmatrix} X_{1,1} & X_{1,2} \\ X_{2,1} & X_{2,2} \end{bmatrix}$$

Isto é, para todo $1 \leq i, j \leq n/2$, pause

$$\begin{aligned} X_{1,1}[i,j] &= X[i,j], \\ X_{1,2}[i,j] &= X[i, n/2 + j], \\ X_{2,1}[i,j] &= X[n/2 + i, j], \text{ e} \\ X_{2,2}[i,j] &= X[n/2 + i, n/2 + j]. \end{aligned}$$

Algoritmo Recursivo

Seja X uma matriz quadrada de ordem n com n par.

- Seja $X_{1,1}, X_{1,2}, X_{2,1}, X_{2,2}$ as submatrizes de X de ordem $n/2$ conforme:

$$X = \begin{bmatrix} X_{1,1} & X_{1,2} \\ X_{2,1} & X_{2,2} \end{bmatrix}$$

Isto é, para todo $1 \leq i, j \leq n/2$, pause

$$\begin{array}{rcl} X_{1,1}[i,j] & = & X[i,j], \\ X_{1,2}[i,j] & = & X[i, n/2 + j], \\ X_{2,1}[i,j] & = & X[n/2 + i, j], \text{ e} \\ X_{2,2}[i,j] & = & X[n/2 + i, n/2 + j]. \end{array}$$

Caso $n = 2$:

Algoritmo Recursivo

Seja X uma matriz quadrada de ordem n com n par.

- Seja $X_{1,1}, X_{1,2}, X_{2,1}, X_{2,2}$ as submatrizes de X de ordem $n/2$ conforme:

$$X = \begin{bmatrix} X_{1,1} & X_{1,2} \\ X_{2,1} & X_{2,2} \end{bmatrix}$$

Isto é, para todo $1 \leq i, j \leq n/2$, pause

$$\begin{aligned} X_{1,1}[i,j] &= X[i,j], \\ X_{1,2}[i,j] &= X[i, n/2 + j], \\ X_{2,1}[i,j] &= X[n/2 + i, j], \text{ e} \\ X_{2,2}[i,j] &= X[n/2 + i, n/2 + j]. \end{aligned}$$

Caso $n = 2$: Veja que cada $X_{i,j} : 1 \leq j \leq 2$ é a matriz 1×1 dada por $X_{i,j} = \big[X[i,j]\big]$

Algoritmo Recursivo

Seja X uma matriz quadrada de ordem n com n par.

- Seja $X_{1,1}, X_{1,2}, X_{2,1}, X_{2,2}$ as submatrizes de X de ordem $n/2$ conforme:

$$X = \begin{bmatrix} X_{1,1} & X_{1,2} \\ X_{2,1} & X_{2,2} \end{bmatrix}$$

Isto é, para todo $1 \leq i, j \leq n/2$, pause

$$\begin{aligned} X_{1,1}[i,j] &= X[i,j], \\ X_{1,2}[i,j] &= X[i, n/2+j], \\ X_{2,1}[i,j] &= X[n/2+i, j], \text{ e} \\ X_{2,2}[i,j] &= X[n/2+i, n/2+j]. \end{aligned}$$

Caso $n = 2$: Veja que cada $X_{i,j} : 1 \leq j \leq 2$ é a matriz 1×1 dada por $X_{i,j} = \left[X[i,j]\right]$

Assim:

Algoritmo Recursivo

Seja X uma matriz quadrada de ordem n com n par.

- Seja $X_{1,1}, X_{1,2}, X_{2,1}, X_{2,2}$ as submatrizes de X de ordem $n/2$ conforme:

$$X = \begin{bmatrix} X_{1,1} & X_{1,2} \\ X_{2,1} & X_{2,2} \end{bmatrix}$$

Isto é, para todo $1 \leq i, j \leq n/2$, pause

$$\begin{aligned} X_{1,1}[i,j] &= X[i,j], \\ X_{1,2}[i,j] &= X[i, n/2 + j], \\ X_{2,1}[i,j] &= X[n/2 + i, j], \text{ e} \\ X_{2,2}[i,j] &= X[n/2 + i, n/2 + j]. \end{aligned}$$

Caso $n = 2$: Veja que cada $X_{i,j} \colon 1 \leq j \leq 2$ é a matriz 1×1 dada por $X_{i,j} = \big[X[i,j]\big]$

Assim:

$$XY = \begin{bmatrix} X_{1,1} & X_{1,2} \\ X_{2,1} & X_{2,2} \end{bmatrix} \begin{bmatrix} Y_{1,1} & Y_{1,2} \\ Y_{2,1} & Y_{2,2} \end{bmatrix} = \begin{bmatrix} X_{1,1}Y_{1,1} + X_{1,2}Y_{2,1} & X_{1,1}Y_{1,2} + X_{1,2}Y_{2,2} \\ X_{2,1}Y_{1,1} + X_{2,2}Y_{2,1} & X_{2,1}Y_{1,2} + X_{2,2}Y_{2,2} \end{bmatrix}$$

Algoritmo Recursivo

No caso $n = 4$, se

$$X = \begin{pmatrix} x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \\ x_{41} & x_{42} & x_{43} & x_{44} \end{pmatrix}, \text{ e } Y = \begin{pmatrix} y_{11} & y_{12} & y_{13} & y_{14} \\ y_{21} & y_{22} & y_{23} & y_{24} \\ y_{31} & y_{32} & y_{33} & y_{34} \\ y_{41} & y_{42} & y_{43} & y_{44} \end{pmatrix},$$

Algoritmo Recursivo

No caso $n = 4$, se

$$X = \begin{pmatrix} x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \\ x_{41} & x_{42} & x_{43} & x_{44} \end{pmatrix}, \text{ e } Y = \begin{pmatrix} y_{11} & y_{12} & y_{13} & y_{14} \\ y_{21} & y_{22} & y_{23} & y_{24} \\ y_{31} & y_{32} & y_{33} & y_{34} \\ y_{41} & y_{42} & y_{43} & y_{44} \end{pmatrix},$$

Então

Algoritmo Recursivo

No caso $n = 4$, se

$$X = \begin{pmatrix} x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \\ x_{41} & x_{42} & x_{43} & x_{44} \end{pmatrix}, \text{ e } Y = \begin{pmatrix} y_{11} & y_{12} & y_{13} & y_{14} \\ y_{21} & y_{22} & y_{23} & y_{24} \\ y_{31} & y_{32} & y_{33} & y_{34} \\ y_{41} & y_{42} & y_{43} & y_{44} \end{pmatrix},$$

Então

$$X_{1,1} = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix}, X_{1,2} = \begin{bmatrix} x_{13} & x_{14} \\ x_{23} & x_{24} \end{bmatrix}, X_{2,1} = \begin{bmatrix} x_{31} & x_{32} \\ x_{41} & x_{42} \end{bmatrix}, X_{2,2} = \begin{bmatrix} x_{33} & x_{34} \\ x_{43} & x_{44} \end{bmatrix}$$

$$Y_{1,1} = \begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{bmatrix}, Y_{1,2} = \begin{bmatrix} y_{13} & y_{14} \\ y_{23} & y_{24} \end{bmatrix}, Y_{2,1} = \begin{bmatrix} y_{31} & y_{32} \\ y_{41} & y_{42} \end{bmatrix}, Y_{2,2} = \begin{bmatrix} y_{33} & y_{34} \\ y_{43} & y_{44} \end{bmatrix}$$

Algoritmo Recursivo

No caso $n = 4$, se

$$X = \begin{pmatrix} x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \\ x_{41} & x_{42} & x_{43} & x_{44} \end{pmatrix}, \text{ e } Y = \begin{pmatrix} y_{11} & y_{12} & y_{13} & y_{14} \\ y_{21} & y_{22} & y_{23} & y_{24} \\ y_{31} & y_{32} & y_{33} & y_{34} \\ y_{41} & y_{42} & y_{43} & y_{44} \end{pmatrix},$$

Então

$$X_{1,1} = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix}, X_{1,2} = \begin{bmatrix} x_{13} & x_{14} \\ x_{23} & x_{24} \end{bmatrix}, X_{2,1} = \begin{bmatrix} x_{31} & x_{32} \\ x_{41} & x_{42} \end{bmatrix}, X_{2,2} = \begin{bmatrix} x_{33} & x_{34} \\ x_{43} & x_{44} \end{bmatrix}$$

$$Y_{1,1} = \begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{bmatrix}, Y_{1,2} = \begin{bmatrix} y_{13} & y_{14} \\ y_{23} & y_{24} \end{bmatrix}, Y_{2,1} = \begin{bmatrix} y_{31} & y_{32} \\ y_{41} & y_{42} \end{bmatrix}, Y_{2,2} = \begin{bmatrix} y_{33} & y_{34} \\ y_{43} & y_{44} \end{bmatrix}$$

De forma que

No caso $n = 4$, se

$$X = \begin{pmatrix} x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \\ x_{41} & x_{42} & x_{43} & x_{44} \end{pmatrix}, \text{ e } Y = \begin{pmatrix} y_{11} & y_{12} & y_{13} & y_{14} \\ y_{21} & y_{22} & y_{23} & y_{24} \\ y_{31} & y_{32} & y_{33} & y_{34} \\ y_{41} & y_{42} & y_{43} & y_{44} \end{pmatrix},$$

Então

$$X_{1,1} = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix}, X_{1,2} = \begin{bmatrix} x_{13} & x_{14} \\ x_{23} & x_{24} \end{bmatrix}, X_{2,1} = \begin{bmatrix} x_{31} & x_{32} \\ x_{41} & x_{42} \end{bmatrix}, X_{2,2} = \begin{bmatrix} x_{33} & x_{34} \\ x_{43} & x_{44} \end{bmatrix}$$

$$Y_{1,1} = \begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{bmatrix}, Y_{1,2} = \begin{bmatrix} y_{13} & y_{14} \\ y_{23} & y_{24} \end{bmatrix}, Y_{2,1} = \begin{bmatrix} y_{31} & y_{32} \\ y_{41} & y_{42} \end{bmatrix}, Y_{2,2} = \begin{bmatrix} y_{33} & y_{34} \\ y_{43} & y_{44} \end{bmatrix}$$

De forma que

$$X = \begin{bmatrix} X_{1,1} & X_{1,2} \\ X_{2,1} & X_{2,2} \end{bmatrix}, \text{ e } Y = \begin{bmatrix} Y_{1,1} & Y_{1,2} \\ Y_{2,1} & Y_{2,2} \end{bmatrix}$$

Algoritmo Recursivo

No caso $n = 4$, se

$$X = \begin{pmatrix} x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \\ x_{41} & x_{42} & x_{43} & x_{44} \end{pmatrix}, \text{ e } Y = \begin{pmatrix} y_{11} & y_{12} & y_{13} & y_{14} \\ y_{21} & y_{22} & y_{23} & y_{24} \\ y_{31} & y_{32} & y_{33} & y_{34} \\ y_{41} & y_{42} & y_{43} & y_{44} \end{pmatrix},$$

Então

$$X_{1,1} = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix}, X_{1,2} = \begin{bmatrix} x_{13} & x_{14} \\ x_{23} & x_{24} \end{bmatrix}, X_{2,1} = \begin{bmatrix} x_{31} & x_{32} \\ x_{41} & x_{42} \end{bmatrix}, X_{2,2} = \begin{bmatrix} x_{33} & x_{34} \\ x_{43} & x_{44} \end{bmatrix}$$

$$Y_{1,1} = \begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{bmatrix}, Y_{1,2} = \begin{bmatrix} y_{13} & y_{14} \\ y_{23} & y_{24} \end{bmatrix}, Y_{2,1} = \begin{bmatrix} y_{31} & y_{32} \\ y_{41} & y_{42} \end{bmatrix}, Y_{2,2} = \begin{bmatrix} y_{33} & y_{34} \\ y_{43} & y_{44} \end{bmatrix}$$

De forma que

$$X = \begin{bmatrix} X_{1,1} & X_{1,2} \\ X_{2,1} & X_{2,2} \end{bmatrix}, \text{ e } Y = \begin{bmatrix} Y_{1,1} & Y_{1,2} \\ Y_{2,1} & Y_{2,2} \end{bmatrix}$$

Repetindo a mesma ideia:

Algoritmo Recursivo

No caso $n = 4$, se

$$X = \begin{pmatrix} x_{11} & x_{12} & x_{13} & x_{14} \\ x_{21} & x_{22} & x_{23} & x_{24} \\ x_{31} & x_{32} & x_{33} & x_{34} \\ x_{41} & x_{42} & x_{43} & x_{44} \end{pmatrix}, \text{ e } Y = \begin{pmatrix} y_{11} & y_{12} & y_{13} & y_{14} \\ y_{21} & y_{22} & y_{23} & y_{24} \\ y_{31} & y_{32} & y_{33} & y_{34} \\ y_{41} & y_{42} & y_{43} & y_{44} \end{pmatrix},$$

Então

$$X_{1,1} = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix}, X_{1,2} = \begin{bmatrix} x_{13} & x_{14} \\ x_{23} & x_{24} \end{bmatrix}, X_{2,1} = \begin{bmatrix} x_{31} & x_{32} \\ x_{41} & x_{42} \end{bmatrix}, X_{2,2} = \begin{bmatrix} x_{33} & x_{34} \\ x_{43} & x_{44} \end{bmatrix}$$

$$Y_{1,1} = \begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{bmatrix}, Y_{1,2} = \begin{bmatrix} y_{13} & y_{14} \\ y_{23} & y_{24} \end{bmatrix}, Y_{2,1} = \begin{bmatrix} y_{31} & y_{32} \\ y_{41} & y_{42} \end{bmatrix}, Y_{2,2} = \begin{bmatrix} y_{33} & y_{34} \\ y_{43} & y_{44} \end{bmatrix}$$

De forma que

$$X = \begin{bmatrix} X_{1,1} & X_{1,2} \\ X_{2,1} & X_{2,2} \end{bmatrix}, \text{ e } Y = \begin{bmatrix} Y_{1,1} & Y_{1,2} \\ Y_{2,1} & Y_{2,2} \end{bmatrix}$$

Repetindo a mesma ideia:

$$\begin{bmatrix} X_{1,1} & X_{1,2} \\ X_{2,1} & X_{2,2} \end{bmatrix} \begin{bmatrix} Y_{1,1} & Y_{1,2} \\ Y_{2,1} & Y_{2,2} \end{bmatrix} = \begin{bmatrix} X_{1,1}Y_{1,1} + X_{1,2}Y_{2,1} & X_{1,1}Y_{1,2} + X_{1,2}Y_{2,2} \\ X_{2,1}Y_{1,1} + X_{2,2}Y_{2,1} & X_{2,1}Y_{1,2} + X_{2,2}Y_{2,2} \end{bmatrix}$$

Algoritmo Recursivo

Teorema 59

Sejam X, Y matrizes de dimensão n com n par. Então

$$XY = \begin{bmatrix} X_{1,1}Y_{1,1} + X_{1,2}Y_{2,1} & X_{1,1}Y_{1,2} + X_{1,2}Y_{2,2} \\ X_{2,1}Y_{1,1} + X_{2,2}Y_{2,1} & X_{2,1}Y_{1,2} + X_{2,2}Y_{2,2} \end{bmatrix}$$

Teorema 59

Sejam X, Y matrizes de dimensão n com n par. Então

$$XY = \begin{bmatrix} X_{1,1}Y_{1,1} + X_{1,2}Y_{2,1} & X_{1,1}Y_{1,2} + X_{1,2}Y_{2,2} \\ X_{2,1}Y_{1,1} + X_{2,2}Y_{2,1} & X_{2,1}Y_{1,2} + X_{2,2}Y_{2,2} \end{bmatrix}$$

Multiplica(X, Y)

$n \leftarrow \dim(X)$
Se $n = 1$
  Devolva $\begin{bmatrix} X[1,1] \times Y[1,1] \end{bmatrix}$

Se $n = 2$
  Devolva $\begin{bmatrix} X[1,1] \times Y[1,1] + X[1,2] \times Y[2,1] & X[1,1] \times Y[1,2] + X[1,2] \times Y[2,2] \\ X[2,1] \times Y[1,1] + X[2,2] \times Y[2,1] & X[2,1] \times Y[1,2] + X[2,2] \times Y[2,2] \end{bmatrix}$

Se n *é ímpar*
  acrescente uma linha e uma coluna de 0s a X e a Y

$R \leftarrow$ matriz $n \times n$
$R_{1,1} \leftarrow$ Multiplica($X_{1,1}$, $Y_{1,1}$) + Multiplica($X_{1,2}$, $Y_{2,1}$)
$R_{1,2} \leftarrow$ Multiplica($X_{1,1}$, $Y_{1,2}$) + Multiplica($X_{1,2}$, $Y_{2,2}$)
$R_{2,1} \leftarrow$ Multiplica($X_{2,1}$, $Y_{1,1}$) + Multiplica($X_{2,2}$, $Y_{2,1}$)
$R_{2,2} \leftarrow$ Multiplica($X_{2,1}$, $Y_{1,2}$) + Multiplica($X_{2,2}$, $Y_{2,2}$)

Devolva R

Algoritmo Recursivo

- Que problema resolve?

Algoritmo Recursivo

- Que problema resolve? MMQ

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto?

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto? Teorema 59

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto? Teorema 59
- Quanto custa?

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto? Teorema 59
- Quanto custa?

Teorema 60

O tempo de execução de Multiplica(X, Y) é $\Theta(n^3)$, onde $n = \dim(X)$.

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto? Teorema 59
- Quanto custa?

Teorema 60

O tempo de execução de Multiplica(X, Y) é $\Theta(n^3)$, onde $n = \dim(X)$.

Prova:

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto? Teorema 59
- Quanto custa?

Teorema 60

O tempo de execução de Multiplica(X, Y) é $\Theta(n^3)$, onde $n = \dim(X)$.

Prova:

Seja (X, Y) uma instância de MMQ e seja $n = \dim(X)$.

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto? Teorema 59
- Quanto custa?

Teorema 60

O tempo de execução de Multiplica(X, Y) é $\Theta(n^3)$, onde $n = \dim(X)$.

Prova:

Seja (X, Y) uma instância de MMQ e seja $n = \dim(X)$.
O tempo consumido fora das chamadas recursivas é $\Theta(n^2)$.

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto? Teorema 59
- Quanto custa?

Teorema 60

O tempo de execução de Multiplica(X, Y) é $\Theta(n^3)$, onde $n = \dim(X)$.

Prova:

Seja (X, Y) uma instância de MMQ e seja $n = \dim(X)$.
O tempo consumido fora das chamadas recursivas é $\Theta(n^2)$. Na notação do T.43

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto? Teorema 59
- Quanto custa?

Teorema 60

O tempo de execução de Multiplica(X, Y) é $\Theta(n^3)$, onde $n = \dim(X)$.

Prova:

Seja (X, Y) uma instância de MMQ e seja $n = \dim(X)$.
O tempo consumido fora das chamadas recursivas é $\Theta(n^2)$. Na notação do T.43

$$T(n) = 8T\left(\frac{n}{2}\right) + \Theta(n^2),$$

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto? Teorema 59
- Quanto custa?

Teorema 60

O tempo de execução de Multiplica(X, Y) é $\Theta(n^3)$, onde $n = \dim(X)$.

Prova:

Seja (X, Y) uma instância de MMQ e seja $n = \dim(X)$.
O tempo consumido fora das chamadas recursivas é $\Theta(n^2)$. Na notação do T.43

$$T(n) = 8T\left(\frac{n}{2}\right) + \Theta(n^2),$$

com

$$\begin{aligned} a &= 8, \\ b &= 2, \\ f(n) &= \Theta(n^2), \end{aligned}$$

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto? Teorema 59
- Quanto custa?

Teorema 60

O tempo de execução de Multiplica(X, Y) é $\Theta(n^3)$, onde $n = \dim(X)$.

Prova:

Seja (X, Y) uma instância de MMQ e seja $n = \dim(X)$.
O tempo consumido fora das chamadas recursivas é $\Theta(n^2)$. Na notação do T.43

$$T(n) = 8T\left(\frac{n}{2}\right) + \Theta(n^2),$$

com

$$\begin{aligned} a &= 8, \\ b &= 2, \\ f(n) &= \Theta(n^2), \end{aligned}$$

A partir disso, $n^{\log_b a}$

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto? Teorema 59
- Quanto custa?

Teorema 60

O tempo de execução de Multiplica(X, Y) é $\Theta(n^3)$, onde $n = \dim(X)$.

Prova:

Seja (X, Y) uma instância de MMQ e seja $n = \dim(X)$.
O tempo consumido fora das chamadas recursivas é $\Theta(n^2)$. Na notação do T.43

$$T(n) = 8T\left(\frac{n}{2}\right) + \Theta(n^2),$$

com

$$\begin{aligned} a &= 8, \\ b &= 2, \\ f(n) &= \Theta(n^2), \end{aligned}$$

A partir disso, $n^{\log_b a} = n^{\log_2 8}$

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto? Teorema 59
- Quanto custa?

Teorema 60

O tempo de execução de Multiplica(X, Y) é $\Theta(n^3)$, onde $n = \dim(X)$.

Prova:

Seja (X, Y) uma instância de MMQ e seja $n = \dim(X)$.
O tempo consumido fora das chamadas recursivas é $\Theta(n^2)$. Na notação do T.43

$$T(n) = 8T\left(\frac{n}{2}\right) + \Theta(n^2),$$

com

$$\begin{aligned} a &= 8, \\ b &= 2, \\ f(n) &= \Theta(n^2), \end{aligned}$$

A partir disso, $n^{\log_b a} = n^{\log_2 8} = n^3$

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto? Teorema 59
- Quanto custa?

Teorema 60

O tempo de execução de Multiplica(X, Y) é $\Theta(n^3)$, onde $n = \dim(X)$.

Prova:

Seja (X, Y) uma instância de MMQ e seja $n = \dim(X)$.
O tempo consumido fora das chamadas recursivas é $\Theta(n^2)$. Na notação do T.43

$$T(n) = 8T\left(\frac{n}{2}\right) + \Theta(n^2),$$

com

$$\begin{aligned} a &= 8, \\ b &= 2, \\ f(n) &= \Theta(n^2), \end{aligned}$$

A partir disso, $n^{\log_b a} = n^{\log_2 8} = n^3$, e como $f(n) = \Omega(n^{3-1})$

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto? Teorema 59
- Quanto custa?

Teorema 60

O tempo de execução de Multiplica(X, Y) é $\Theta(n^3)$, onde $n = \dim(X)$.

Prova:

Seja (X, Y) uma instância de MMQ e seja $n = \dim(X)$.
O tempo consumido fora das chamadas recursivas é $\Theta(n^2)$. Na notação do T.43

$$T(n) = 8T\left(\frac{n}{2}\right) + \Theta(n^2),$$

com

$$\begin{aligned} a &= 8, \\ b &= 2, \\ f(n) &= \Theta(n^2), \end{aligned}$$

A partir disso, $n^{\log_b a} = n^{\log_2 8} = n^3$, e como $f(n) = \Omega(n^{3-1})$, então (T.43):

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto? Teorema 59
- Quanto custa?

Teorema 60

O tempo de execução de Multiplica(X, Y) é $\Theta(n^3)$, onde $n = \dim(X)$.

Prova:

Seja (X, Y) uma instância de MMQ e seja $n = \dim(X)$.
O tempo consumido fora das chamadas recursivas é $\Theta(n^2)$. Na notação do T.43

$$T(n) = 8T\left(\frac{n}{2}\right) + \Theta(n^2),$$

com

$$\begin{aligned} a &= 8, \\ b &= 2, \\ f(n) &= \Theta(n^2), \end{aligned}$$

A partir disso, $n^{\log_b a} = n^{\log_2 8} = n^3$, e como $f(n) = \Omega(n^{3-1})$, então (T.43):

$$T(n) = \Theta(n^{\log_b a}) = \Theta(n^3)$$

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto? Teorema 59
- Quanto custa?

Teorema 60

O tempo de execução de Multiplica(X, Y) é $\Theta(n^3)$, onde $n = \dim(X)$.

Prova:

Seja (X, Y) uma instância de MMQ e seja $n = \dim(X)$.
O tempo consumido fora das chamadas recursivas é $\Theta(n^2)$. Na notação do T.43

$$T(n) = 8T\left(\frac{n}{2}\right) + \Theta(n^2),$$

com

$$\begin{aligned} a &= 8, \\ b &= 2, \\ f(n) &= \Theta(n^2), \end{aligned}$$

A partir disso, $n^{\log_b a} = n^{\log_2 8} = n^3$, e como $f(n) = \Omega(n^{3-1})$, então (T.43):

$$T(n) = \Theta(n^{\log_b a}) = \Theta(n^3)$$

- É possível fazer melhor?

Algoritmo Recursivo

- Que problema resolve? MMQ
- Está correto? Teorema 59
- Quanto custa?

Teorema 60

O tempo de execução de Multiplica(X, Y) é $\Theta(n^3)$, onde $n = \dim(X)$.

Prova:

Seja (X, Y) uma instância de MMQ e seja $n = \dim(X)$.
O tempo consumido fora das chamadas recursivas é $\Theta(n^2)$. Na notação do T.43

$$T(n) = 8T\left(\frac{n}{2}\right) + \Theta(n^2),$$

com

$$\begin{aligned} a &= 8, \\ b &= 2, \\ f(n) &= \Theta(n^2), \end{aligned}$$

A partir disso, $n^{\log_b a} = n^{\log_2 8} = n^3$, e como $f(n) = \Omega(n^{3-1})$, então (T.43):

$$T(n) = \Theta(n^{\log_b a}) = \Theta(n^3)$$

- É possível fazer melhor?Sim

Teorema 61

Sejam X, Y matrizes de dimensão n com n par e sejam

$$
\begin{aligned}
P_1 &= X_{1,1}(Y_{1,2} - Y_{2,2}),\\
P_2 &= (X_{1,1} + X_{1,2})Y_{2,2},\\
P_3 &= (X_{2,1} + X_{2,2})Y_{1,1},\\
P_4 &= X_{2,2}(Y_{2,1} - Y_{1,1}),\\
P_5 &= (X_{1,1} + X_{2,2})(Y_{1,1} + Y_{2,2}),\\
P_6 &= (X_{1,2} - X_{2,2})(Y_{2,1} + Y_{2,2}),\\
P_7 &= (X_{1,1} - X_{2,1})(Y_{1,1} + Y_{1,2}).
\end{aligned}
$$

Então

$$
\begin{aligned}
P_5 + P_4 - P_2 + P_6 &= X_{1,1}Y_{1,1} + X_{1,2}Y_{2,1} = (XY)_{1,1},\\
P_1 + P_2 &= X_{1,1}Y_{1,2} + X_{1,2}Y_{2,2} = (XY)_{1,2},\\
P_3 + P_4 &= X_{2,1}Y_{1,1} + X_{2,2}Y_{2,1} = (XY)_{2,1},\\
P_1 + P_5 - P_3 - P_7 &= X_{2,1}Y_{1,2} + X_{2,2}Y_{2,2} = (XY)_{2,2}.
\end{aligned}
$$

Algoritmo de Strassen (1969)

Multiplica(X, Y)

$n \leftarrow \dim(X)$

Se $n = 1$
  Devolva $\left[X[1,1] \times Y[1,1]\right]$

Se $n = 2$
  Devolva $\begin{bmatrix} X[1,1] \times Y[1,1] + X[1,2] \times Y[2,1] & X[1,1] \times Y[1,2] + X[1,2] \times Y[2,2] \\ X[2,1] \times Y[1,1] + X[2,2] \times Y[2,1] & X[2,1] \times Y[1,2] + X[2,2] \times Y[2,2] \end{bmatrix}$

Se n *é ímpar*
  acrescente uma linha e uma coluna de 0s a X e a Y
  $n \leftarrow n + 1$

$P_1 \leftarrow$ Multiplica($X_{1,1}$, $Y_{1,2} - Y_{2,2}$)
$P_2 \leftarrow$ Multiplica($X_{1,1} + X_{1,2}$, $Y_{2,2}$)
$P_3 \leftarrow$ Multiplica($X_{2,1} + X_{2,2}$, $Y_{1,1}$)
$P_4 \leftarrow$ Multiplica($X_{2,2}$, $Y_{2,1} - Y_{1,1}$)
$P_5 \leftarrow$ Multiplica($X_{1,1} + X_{2,2}$, $Y_{1,1} + Y_{2,2}$)
$P_6 \leftarrow$ Multiplica($X_{1,2} - X_{2,2}$, $Y_{2,1} + Y_{2,2}$)
$P_7 \leftarrow$ Multiplica($X_{1,1} - X_{2,1}$, $Y_{1,1} + Y_{1,2}$)

$R_{1,1} \leftarrow P_5 + P_4 - P_2 + P_6$
$R_{1,2} \leftarrow P_1 + P_2$
$R_{2,1} \leftarrow P_3 + P_4$
$R_{2,2} \leftarrow P_1 + P_5 - P_3 - P_7$

remova a linha e a coluna de 0s de R resultantes das filas de 0s acrescentadas a X e a Y
Devolva R

Algoritmo de Strassen (1969)

Teorema 62

O tempo de execução de Multiplica(X, Y) é $\Theta(n^{\lg 7})$, onde $n = \dim(X)$.

Prova:

Seja (X, Y) uma instância de MMQ e seja $n = \dim(X)$.
O tempo consumido fora das chamadas recursivas é $\Theta(n^2)$.
Na notação do T.43, temos

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

com

$$\begin{aligned} a &= 7, \\ b &= 2, \\ f(n) &= \Theta(n^2), \end{aligned}$$

e daí

$$n^{\log_b a} = n^{\log_2 7} = n^{\lg 7}$$

e como

$$f(n) = \Omega(n^{\lg 7 - (\lg 7 - 2)}),$$

então (T.43)

$$T(n) = \Theta(n^{\log_b a}) = \Theta(n^{\lg 7}).$$

Algoritmo de Strassen (1969)

Corolario 63

O tempo de execução de Strassen(X, Y) é $\Omega((n)^{2.807})$ e $\mathcal{O}((n)^{2.808})$, onde $n = \dim(X)$.

Algoritmo de Strassen (1969)

Corolario 63

O tempo de execução de Strassen(X, Y) é $\Omega((n)^{2.807})$ e $\mathcal{O}((n)^{2.808})$, onde $n = \dim(X)$.

- Está correto?

Algoritmo de Strassen (1969)

Corolario 63

O tempo de execução de Strassen(X, Y) é $\Omega((n)^{2.807})$ e $\mathcal{O}((n)^{2.808})$, onde $n = \dim(X)$.

- Está correto? Teorema 61

Algoritmo de Strassen (1969)

Corolario 63

O tempo de execução de Strassen(X, Y) é $\Omega((n)^{2.807})$ e $\mathcal{O}((n)^{2.808})$, onde $n = \dim(X)$.

- Está correto? Teorema 61
- É possível fazer melhor?

Algoritmo de Strassen (1969)

Corolario 63

O tempo de execução de Strassen(X, Y) é $\Omega((n)^{2.807})$ e $\mathcal{O}((n)^{2.808})$, onde $n = \dim(X)$.

- Está correto? Teorema 61
- É possível fazer melhor? $\mathcal{O}(n^{2.3728639})$.