Aufgabe 4
Matrixmultiplikation
PdP WS 2005/2006

Maximilian Störzer, Daniel Wasserrab, Dennis Giffhorn

LS Softwaresysteme

11. Mai 2006

Motivation

Schulmethode für Matrixmultiplikation
zweier $n \times n$ Matrizen A und B:

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

Beispiel: multipliziere zwei 2×2 Matrizen

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} * \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix} =$$

$$\begin{pmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{pmatrix}$$

Motivation

Es gibt aber bessere Algorithmen zur Matrixmultiplikation:

- Schulmethode für Matrixmultiplikation hat Aufwand $O(n^3)$.
- Die Matrixmultiplikation nach **Schönhage und Strassen** hat "nur" den Aufwand $O(n^{2,81})$!

$\Rightarrow$ D.h. diese Art der Multiplikation ist gut für grosse $n \times n$ Matrizen

Grundidee: Addieren und Subtrahieren ist billiger als Multiplizieren.

Motivation

- Methode von Schönhage und Strassen zur Multiplikation zweier 2×2 Matrizen A und B:

$$
\begin{aligned}
i &= (a_{12} - a_{22}) * (b_{21} + b_{22}) \\
ii &= (a_{11} + a_{22}) * (b_{11} + b_{22}) \\
iii &= (a_{11} - a_{21}) * (b_{11} + b_{12}) \\
iv &= (a_{11} + a_{12}) * b_{22} \\
v &= a_{11} * (b_{12} - b_{22}) \\
vi &= a_{22} * (b_{21} - b_{11}) \\
vii &= (a_{21} + a_{22}) * b_{11}
\end{aligned}
$$

$$
\begin{aligned}
c_{11} &= i + ii - iv + vi \\
c_{12} &= iv + v \\
c_{21} &= vi + vii \\
c_{22} &= ii - iii + v - vii
\end{aligned}
$$

Multiplikation von $n \times n$ Matrizen

Das ist auch auf große Matrizen anwendbar!

- Mit diesen Formeln können auch rekursiv zwei $n \times n$ Matrizen multipliziert werden
- Einschränkung: n muss eine Zweierpotenz sein
- Die Matrizen werden in jeweils vier $\frac{n}{2} \times \frac{n}{2}$ Matrizen zerlegt:

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} * \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix} = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix}$$

- Die Multiplikationen in den Formeln sind damit wiederum *Matrixmultiplikationen*, die Additionen/Subtraktionen *komponentenweise* Additionen/Subtraktionen

Multiplikation von $n \times n$ Matrizen

- Methode von Schönhage und Strassen:

$$\begin{aligned}
I &= (A_{12} - A_{22}) * (B_{21} + B_{22}) \\
II &= (A_{11} + A_{22}) * (B_{11} + B_{22}) \\
III &= (A_{11} - A_{21}) * (B_{11} + B_{12}) \\
IV &= (A_{11} + A_{12}) * B_{22} \\
V &= A_{11} * (B_{12} - B_{22}) \\
VI &= A_{22} * (B_{21} - B_{11}) \\
VII &= (A_{21} + A_{22}) * B_{11}
\end{aligned}$$

$$\begin{aligned}
C_{11} &= I + II - IV + VI \\
C_{12} &= IV + V \\
C_{21} &= VI + VII \\
C_{22} &= II - III + V - VII
\end{aligned}$$

Algorithmus

```
Matrix mult(Matrix A, Matrix B) {
  if (A.size() == 1) {
    return A.get(1,1) * B.get(1,1);
  } else {
    berechne Teilmatrizen $A_{11}, ..., A_{22}, B_{11}, ..., B_{22}$;
    berechne $I, ..., VII, C_{11}, ..., C_{22}$ nach obigen Formeln;
    for ($X * Y$ in den Formeln) {
      mult($X$, $Y$):
    }
    Matrix C = $\begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix}$;
    return C;
  }
}
```

Beispiel : Multiplikation von 4 × 4 Matrizen

- $$\begin{pmatrix} 0 & 1 & 2 & 3 \\ 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 5 \\ 3 & 4 & 5 & 6 \end{pmatrix} * \begin{pmatrix} 0 & 3 & 6 & 9 \\ 2 & 5 & 8 & 11 \\ 4 & 7 & 10 & 13 \\ 6 & 9 & 12 & 15 \end{pmatrix}$$

$$\begin{array}{ll}
A_{11} = \begin{pmatrix} 0 & 1 \\ 1 & 2 \end{pmatrix} & B_{11} = \begin{pmatrix} 0 & 3 \\ 2 & 5 \end{pmatrix} \\
A_{12} = \begin{pmatrix} 2 & 3 \\ 3 & 4 \end{pmatrix} & B_{12} = \begin{pmatrix} 6 & 9 \\ 8 & 11 \end{pmatrix} \\
A_{21} = \begin{pmatrix} 2 & 3 \\ 3 & 4 \end{pmatrix} & B_{21} = \begin{pmatrix} 4 & 7 \\ 6 & 9 \end{pmatrix} \\
A_{22} = \begin{pmatrix} 4 & 5 \\ 5 & 6 \end{pmatrix} & B_{22} = \begin{pmatrix} 10 & 13 \\ 12 & 15 \end{pmatrix}
\end{array}$$

Beispiel : Multiplikation von 4 × 4 Matrizen

$$
\begin{aligned}
I &= (A_{12} - A_{22}) * (B_{21} + B_{22}) &&= \begin{pmatrix} -64 & -88 \\ -64 & -88 \end{pmatrix} \\
II &= (A_{11} + A_{22}) * (B_{11} + B_{22}) &&= \begin{pmatrix} 124 & 184 \\ 172 & 256 \end{pmatrix} \\
III &= (A_{11} - A_{21}) * (B_{11} + B_{12}) &&= \begin{pmatrix} -32 & -56 \\ -32 & -56 \end{pmatrix} \\
IV &= (A_{11} + A_{12}) * B_{22} &&= \begin{pmatrix} 68 & 86 \\ 112 & 142 \end{pmatrix} \\
V &= A_{11} * (B_{12} - B_{22}) &&= \begin{pmatrix} -4 & -4 \\ -12 & -12 \end{pmatrix} \\
VI &= A_{22} * (B_{21} - B_{11}) &&= \begin{pmatrix} 36 & 36 \\ 44 & 44 \end{pmatrix} \\
VII &= (A_{21} + A_{22}) * B_{11} &&= \begin{pmatrix} 16 & 58 \\ 20 & 74 \end{pmatrix}
\end{aligned}
$$

Beispiel : Multiplikation von 4 × 4 Matrizen

$$
\begin{aligned}
C_{11} &= I + II - IV + VI &&= \begin{pmatrix} 28 & 46 \\ 40 & 70 \end{pmatrix} \\
C_{12} &= IV + V &&= \begin{pmatrix} 64 & 82 \\ 100 & 130 \end{pmatrix} \\
C_{21} &= VI + VII &&= \begin{pmatrix} 52 & 94 \\ 64 & 118 \end{pmatrix} \\
C_{22} &= II - III + V - VII &&= \begin{pmatrix} 136 & 178 \\ 172 & 226 \end{pmatrix}
\end{aligned}
$$

Beispiel : Multiplikation von 4 × 4 Matrizen

- also:

$$\begin{pmatrix} 0 & 1 & 2 & 3 \\ 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 5 \\ 3 & 4 & 5 & 6 \end{pmatrix} * \begin{pmatrix} 0 & 3 & 6 & 9 \\ 2 & 5 & 8 & 11 \\ 4 & 7 & 10 & 13 \\ 6 & 9 & 12 & 15 \end{pmatrix} =$$

$$\begin{pmatrix} 28 & 46 & 64 & 82 \\ 40 & 70 & 100 & 130 \\ 52 & 94 & 136 & 178 \\ 64 & 118 & 172 & 226 \end{pmatrix}$$

Multiplikation von $n \times n$ Matrizen

Ein paar Anmerkungen:

- Die Schulmethode ist in der Praxis für kleinere Matrizen schneller.
- Daher wird eine Grösse m angegeben,
  ab der die Rekursion abbrechen soll.
- $m \times m$ Matrizen oder noch kleinere Matrizen werden dann
  mittels der Schulmethode multipliziert.

Das ist ein allgemeines Verfahren—man verwendet immer den
für den konkreten Input besten Algorithmus.

Verbesserter Algorithmus

```
Matrix mult(Matrix A, Matrix B, int limit) {
  if (A.size() <= limit) {
    return schulmethode(A, B);
  } else {
    berechne Teilmatrizen $A_{11}, ..., A_{22}, B_{11}, ..., B_{22}$;
    berechne $I, ..., VII, C_{11}, ..., C_{22}$ nach obigen Formeln;
    for ($X * Y$ in den Formeln {
      mult($X$, $Y$, limit);
    }
    Matrix C = $\begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix}$;
    return C;
  }
}
```

Einschränkung?

Der Algorithmus funktioniert nur für zwei $n \times n$ Matrizen, mit $n = 2^k$—Heisst das er ist zu speziell für praktische Anwendungen?

Nein, da $m \times m$ Matrizen in $n \times n$ Matrizen mit $n =$ Zweierpotenz transformiert werden können:

$$\begin{pmatrix} 0 & 3 & 6 \\ 2 & 5 & 8 \\ 4 & 7 & 10 \end{pmatrix} \longrightarrow \begin{pmatrix} 0 & 3 & 6 & 0 \\ 2 & 5 & 8 & 0 \\ 4 & 7 & 10 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Für Matrixmultiplikation und -addition ist dieses *Auffüllen mit 0 ungefährlich*. Vor der Rückgabe des Ergebnisses muss das Ergebnis wieder in eine $m \times m$ Matrix umgeformt werden (durch *Abschneiden der hinzugefügten Zeilen und Spalten*).

Zur Implementierung

Dem Programm wird eine durchzuführende Matrixmultiplikation als Folge von 6 Integern $n\ n_0\ x_a\ x_b\ y_a\ y_b$ übergeben.

- n ist die Grösse der $n \times n$ Matrizen A und B, welche multipliziert werden sollen
- n_0 ist die Grösse, ab der die Schulmethode verwendet werden soll
- Matrix A berechnet sich durch $a_{ij} = x_a * i + x_b * j$, für $0 <= i, j < n$
- Matrix B berechnet sich durch $b_{ij} = y_a * i + y_b * j$, für $0 <= i, j < n$
- **Wichtig:** Die Indizes beginnen hier bei 0, während sie in den mathematischen Definitionen und Formeln bei 1 beginnen

Keine allgemeine Darstellungsform von Matrizen!
(Eingabe für die Beispielaufgabe ist 4 2 1 1 2 3)

Fragen?

Fragen?