Informatik IIa - Projektarbeit

System zur algebraischen Manipulation von Matrizen

Florian Hilbrecht

Diese Projektarbeit wurde erstellt von:

Jürgen Döffinger	M.-Nr.: 631551
Florian Hilbrecht	M.-Nr.: 631558
Hannes Kuschick	M.-Nr.: 631552
Christoph Dieck	M.-Nr.: 630033

Projektgruppe 1

Jena, 20.06.2010

Wesentliche Quellen:

- Bjarne Stroustrup
  Die C++ - Programmiersprache
  ISBN 0-201-70073-5
  Addision-Wesley Verlag

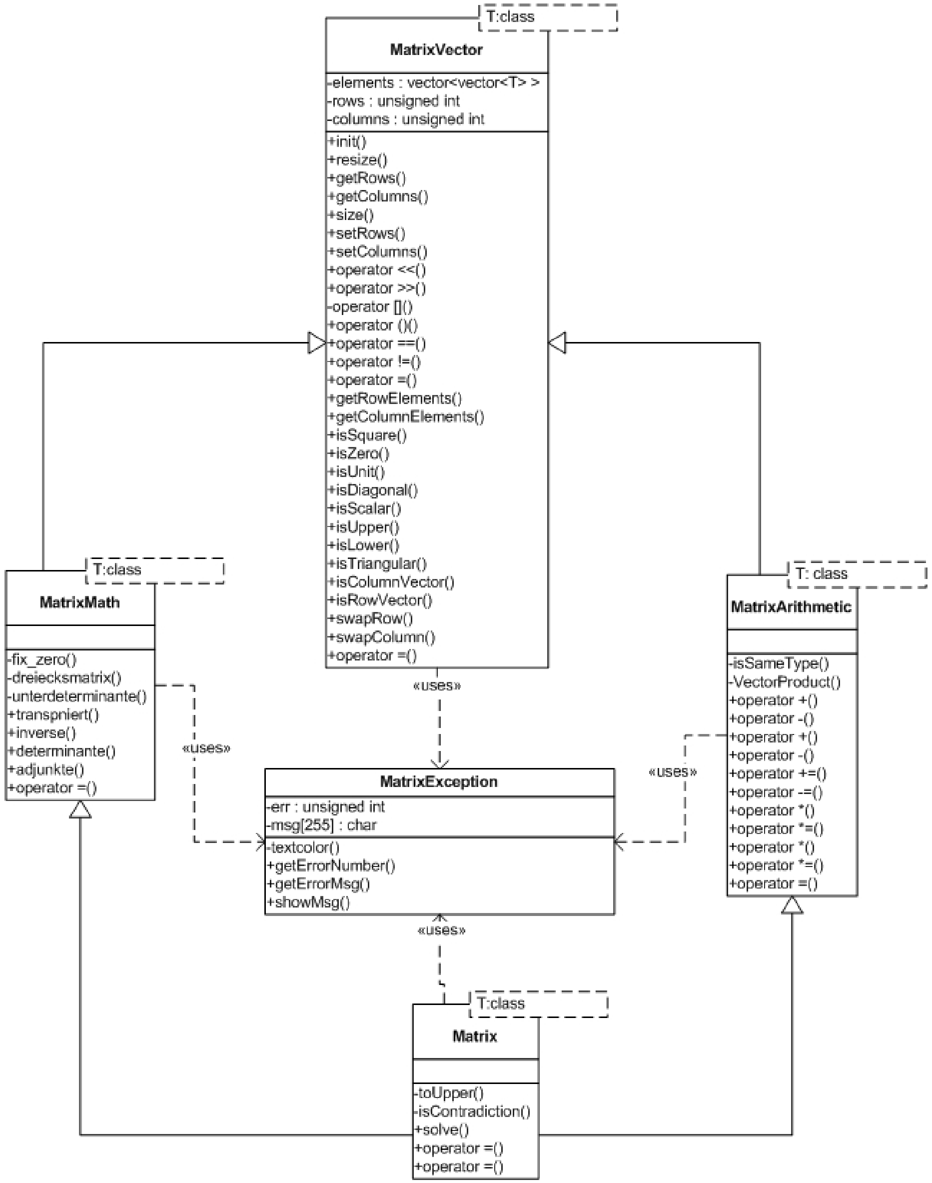
- Jürgen Wolf
  C++ von A bis Z – Das umfassende Handbuch
  2. Auflage, 2009
  ISBN 978-3-8362-1429-2
  Galileo Press

- Prof. Dr.-Ing. Jack
  Fachhochschule Jena
  Fachbereich Elektrotechnik/Informationstechnik
  Vorlesungsskripte Informatik IIa

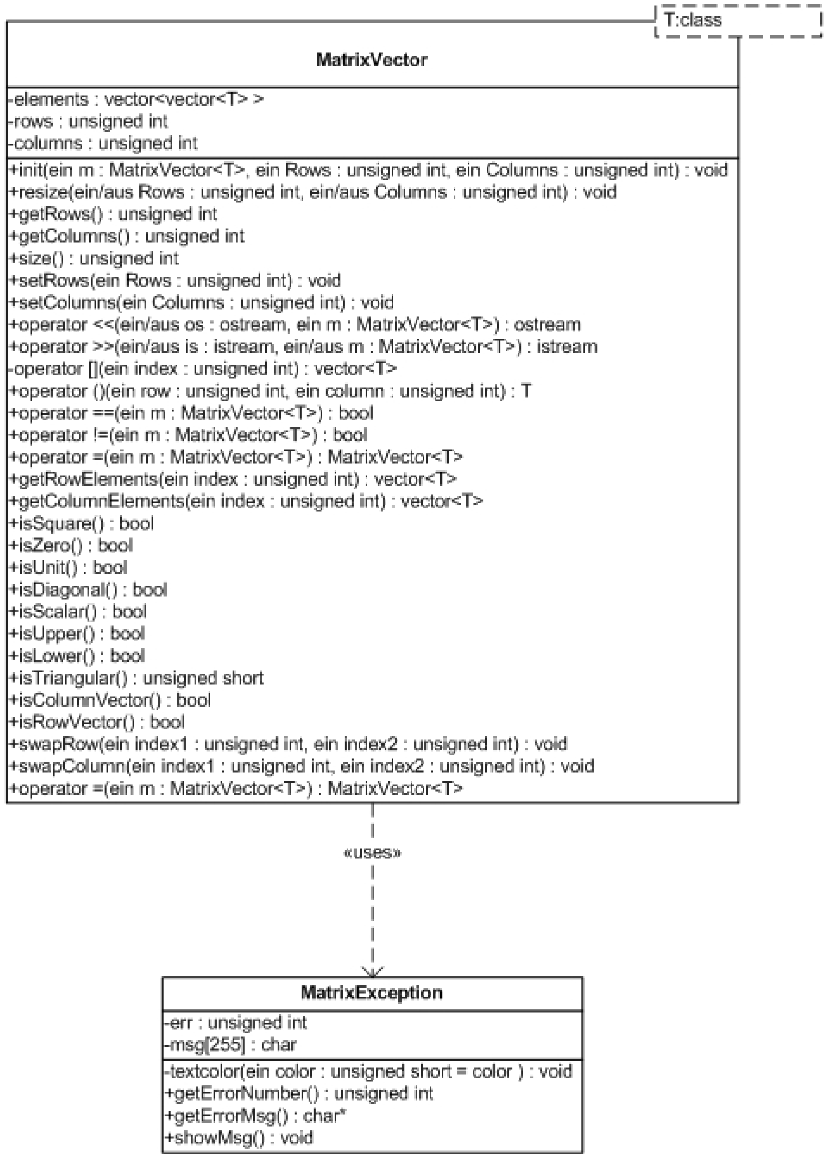
Das im folgenden präsentierte Projekt „Matrix“ wurde entsprechend der Aufgabenstellung
„System zur algebraischen Manipulation von Matrizen“ realisiert.

Das Projekt wurde für die geforderten Datentypen durch Template-Klassen (typenlose) umgesetzt.

Klassendiagramm:



Klassendiagramm
MatrixVector



MatrixVector greift auf die MatrixException zur Fehlerausgabe zu.

Klasse MatrixVector

Die Klasse MatrixVector ist eine Template-Klasse (Typenlos) und enthält die folgenden **Attribute**:

- elements — ist vom Typ vector<vector<T>>
  enthält alle Elemente der Matrix
- rows — entspricht der Zeilenanzahl der Matrix
- columns — entspricht der Spaltenanzahl der Matrix

Die Klasse MatrixVector enthält 28 **Methoden**. Zu den wesentlichsten zählen:

- Überladung des Shift-Operators **>>** zur Eingabe einer Matrix.

- Überladung des Shift-Operators **<<** zur Ausgabe einer Matrix.

- Überladung des Index-Operators **[]** zum Zugriff auf einzelne Elemente einer Matrix.
  Um den *unbefugten Zugriff* auf einzelne Elemente, Zeilen oder Spalten der Klasse MatrixVector zu verhindern, wurde diese Methode als „private“ deklariert (gekapselt)!

In der MatrixVector sind neben dem Copy-Konstruktor auch drei weitere **Konstruktoren**, zur Erstellung einer Matrix definiert.

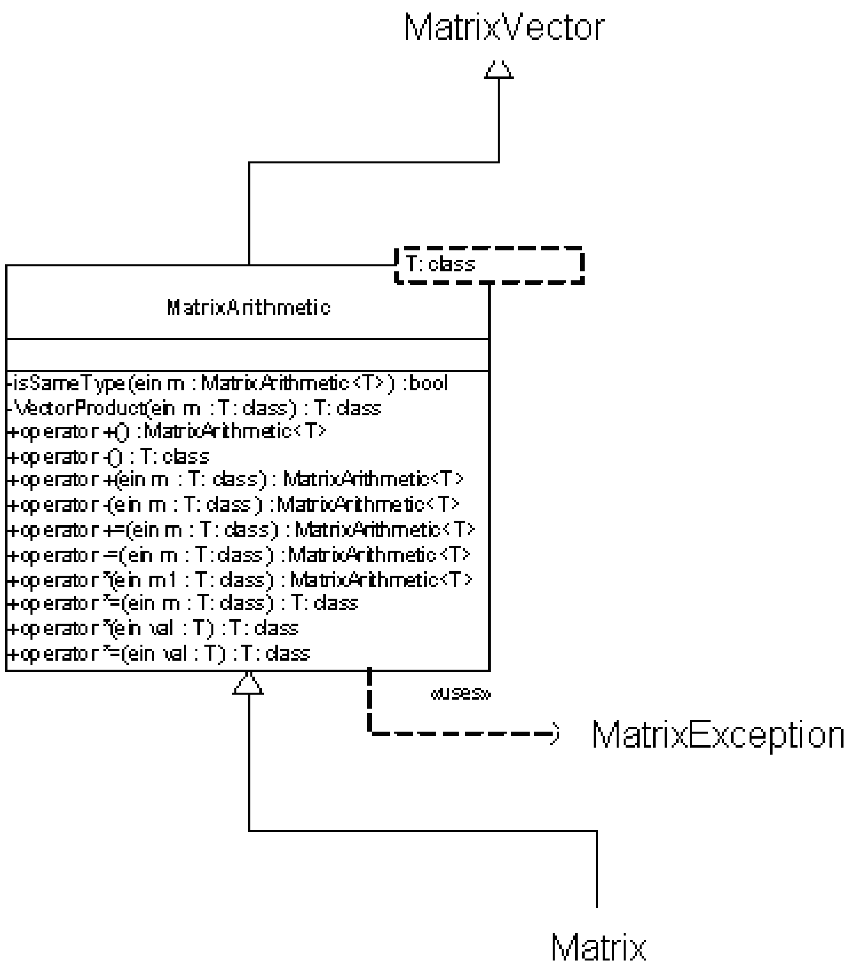
- Standardkonstruktor
  ohne Übergabe von Parametern wird automatisch eine 3x3 Matrix erstellt

- mit Übergabe *eines* Parameters wird eine quadratische Matrix entsprechend des übergebenen Parameters erstellt

- mit Übergabe von *zwei* Parametern wird eine Matrix entsprechend der übergebenen Parameter erstellt (Zeilenanzahl, Spaltenanzahl)

Die Konstruktoren rufen jeweils die Funktion resize() auf.

MatrixArithmetic



Überladene Operatoren:

“+“
- Addition zweier Matrizen mit gleicher Dimension
- Komponentenweise Addition

„-“
- Subtraktion zweier Matrizen mit gleicher Dimension
- Komponentenweise Subtraktion

Überladene Operatoren:

“+=“ - Addition zweier Matrizen mit gleicher Dimension

- Komponentenweise Addition

„-=“ - Subtraktion zweier Matrizen mit gleicher Dimension

- Komponentenweise Subtraktion

Überladene Operatoren:

“*“ - Multiplikation zweier Matrizen

→ Bedingung: - Spaltenanzahl M1 = Zeilenanzahl M2

→ Falk-Schema

- Multiplikation zweier Spaltenvektoren (Kreuzprodukt)

→ Bedingung: - Zeilenanzahl M1 = Zeilenanzahl M2

- beide Matrizen nur 1 Spalte

Überladene Operatoren:

“*“
- Multiplikation Matrix mit Skalar
- Komponentenweise Multiplikation

“*=“
- Multiplikation zweier Matrizen
- Multiplikation Matrix mit Skalar

Fehlermeldungen:

Fehlernummer	Fehlermeldung
200	Die Spaltenanzahl der Matrix 1 stimmt nicht mit der Zeilenanzahl der Matrix 2 ueberein.
201	Bei den Spaltenvektoren liegt keine gleiche Anzahl an Zeilen vor, daher kann das Vektorprodukt nicht gebildet werden.
202	Die beiden zu addierenden Matrizen stimmen nicht in der Anzahl der Zeilen und Spalten ueberein.
203	Die beiden zu subtrahierenden Matrizen stimmen nicht in der Anzahl der Zeilen und Spalten ueberein.
204	Es wurde versucht ein Vektorprodukt zu berechnen, obwohl keine Spaltenvektoren vorlagen.

Klasse MatrixMath

Die Klasse MatrixMath

Funktionen:

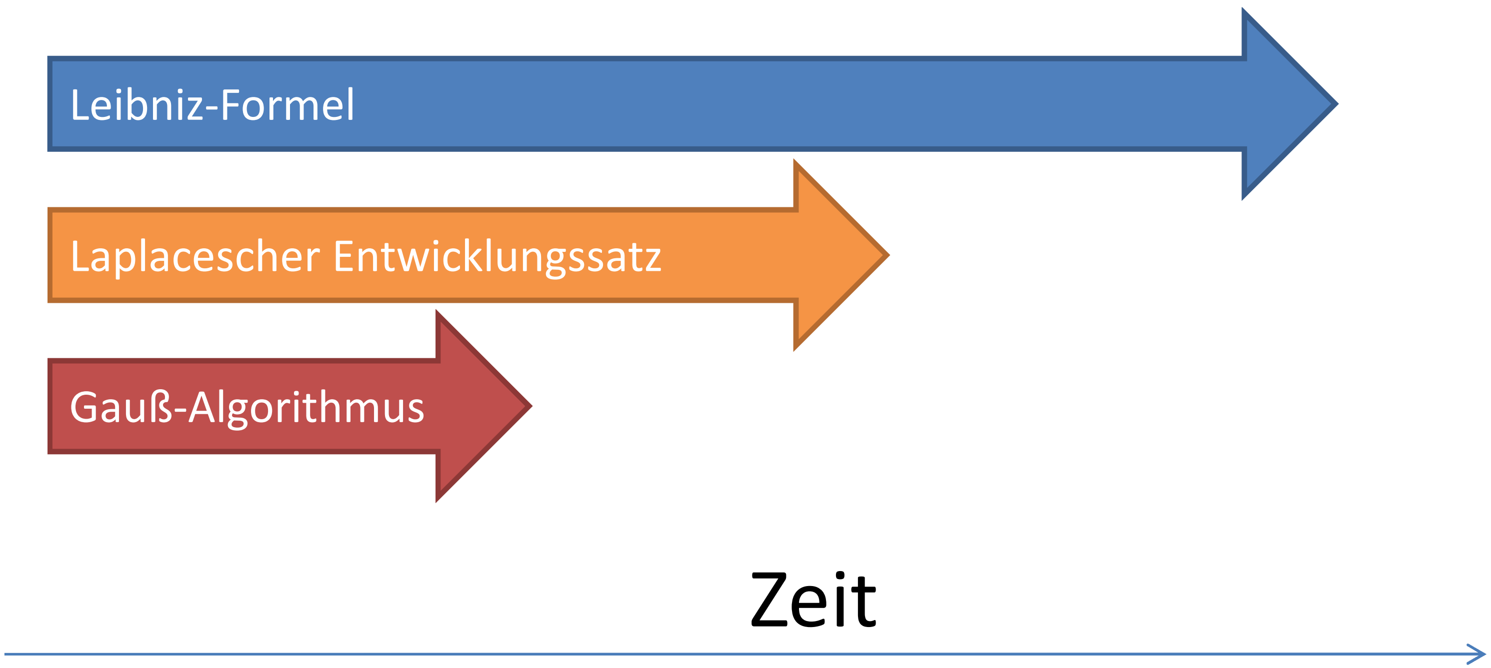
- Berechnung der Determinante
- Adjunkte einer Matrix
- Inverse einer Matrix
- Transponieren einer Matrix

Berechnung einer Determinante

Möglichkeiten:

- Leibniz-Formel
- Laplacescher Entwicklungssatz
- Gauß-Algorithmus

Effizienz


Leibniz-Formel
Laplacescher Entwicklungssatz
Gauß-Algorithmus
Zeit

Gauß-Algorithmus

2-1*2 2-2*2 1-2*2

2/1=2

$$\begin{matrix}1 & 2 & 2\\ 2 & 2 & 1\\ 1 & 1 & 2\end{matrix}$$

$$\begin{bmatrix}1 & 2 & 2\\ 0 & -2 & -3\\ 0 & 0 & 1.5\end{bmatrix}$$

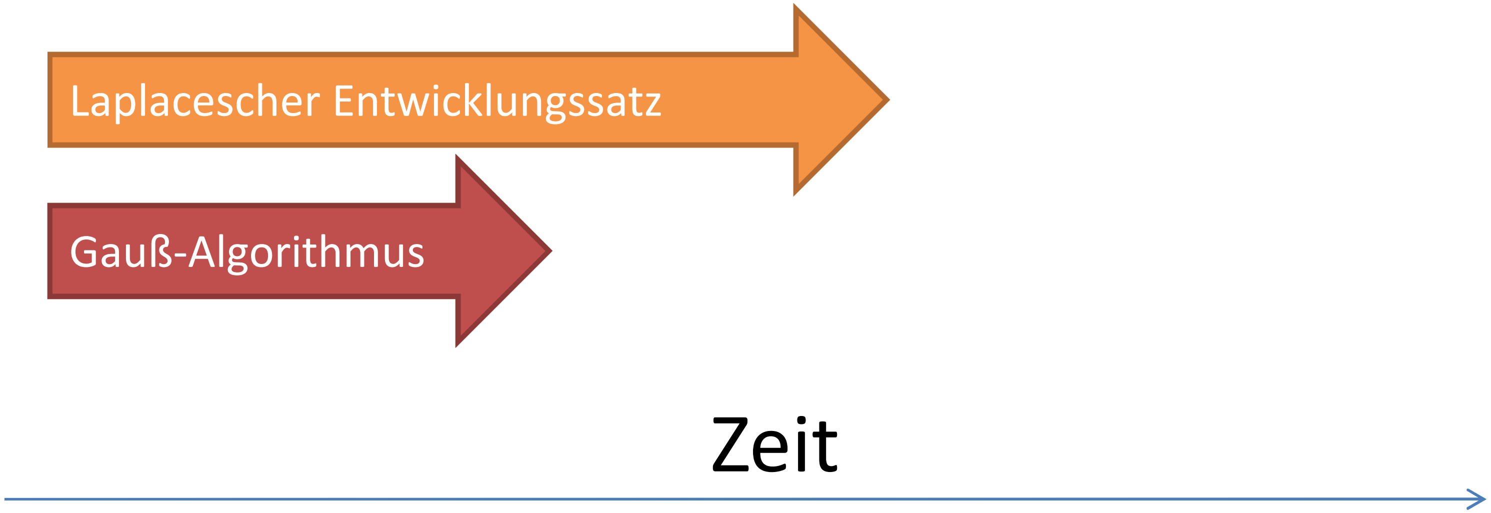
Det(A)=1*-2*1.5=-3

Berechnung der Inversen

Möglichkeiten:

- Laplacescher Entwicklungssatz
- Gauß-Algorithmus

Effizienz



Laplacescher Entwicklungssatz

$$\begin{matrix} 1 & 2 & 2 \\ 2 & 2 & 1 \\ 1 & 1 & 2 \end{matrix} \rightarrow 2*2-1*1=3$$

transponieren

$$\begin{bmatrix} 3 & -2 & -2 \\ -3 & 0 & 3 \\ 0 & 1 & -2 \end{bmatrix} \rightarrow -\frac{1}{3}\begin{bmatrix} 3 & -2 & -2 \\ -3 & 0 & 3 \\ 0 & 1 & -2 \end{bmatrix} \rightarrow \begin{bmatrix} -1 & \frac{2}{3} & \frac{2}{3} \\ 1 & 0 & -1 \\ 0 & -\frac{1}{3} & \frac{2}{3} \end{bmatrix}$$

Gleichungslöser

Informatik IIa – Projektarbeit

Jürgen Döffinger

20.06.2010

nach

Gauß – Jordan - Verfahren

Gliederung

- Was war die Aufgabenstellung?
- Wie wurde der Gleichungslöser umgesetzt?
- Programmbeispiel ?

- **Was war die Aufgabenstellung?**
- Wie wurde der Gleichungslöser umgesetzt?
- Programmbeispiel ?

Was war die Aufgabenstellung

Implementieren Sie ein generisches System zur algebraischen Behandlung von Matrizen. Folgende Operationen sollen realisiert werden:

- Matrix-Addition, -Subtraktion, -Multiplikation
- Skalarmultiplikation
- Vektor-Multiplikation
- Gauß-Elimination (Lösung von linearen Gleichungssystemen)
- Determinantenberechnung
- Matrix-Inversion.

Was war die Aufgabenstellung

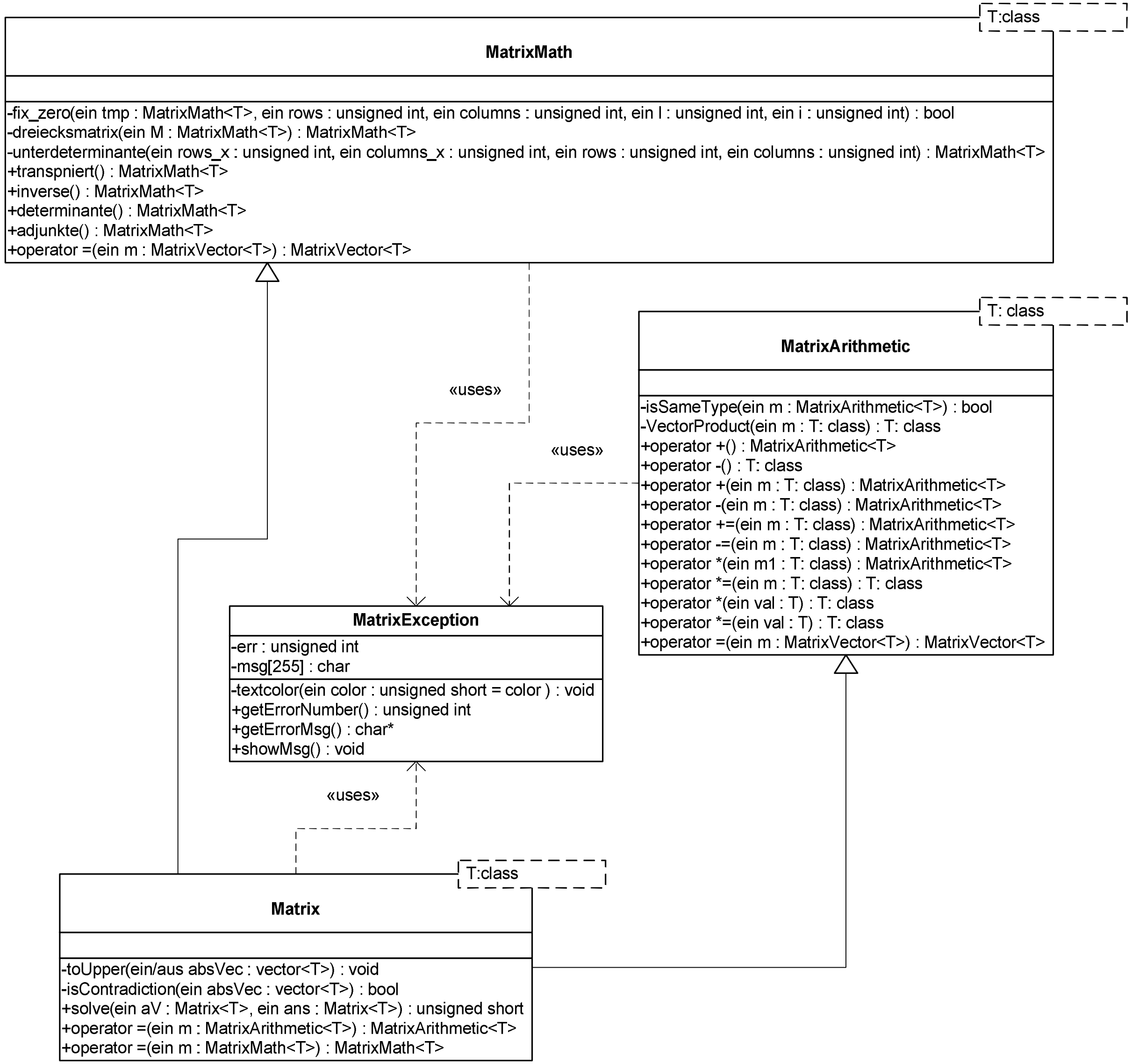
Implementieren Sie ein generisches System zur algebraischen Behandlung von Matrizen. Folgende Operationen sollen realisiert werden:

- Matrix-Addition, -Subtraktion, -Multiplikation
- Skalarmultiplikation
- Vektor-Multiplikation
- **Gauß-Elimination (Lösung von linearen Gleichungssystemen)**
- Determinantenberechnung
- Matrix-Inversion.

- **Was war die Aufgabenstellung?**
- Wie wurde der Gleichungslöser umgesetzt?
- Programmbeispiel ?

- Was war die Aufgabenstellung?
- **Wie wurde der Gleichungslöser umgesetzt?**
- Programmbeispiel ?

Wie wurde der Gleichungslöser umgesetzt?



Wie wurde der Gleichungslöser umgesetzt?

Bezeichnungen

A: Koeffizientenmatrix des Systems

x: Lösungsvektor

c: Spaltenvektor aus den *absoluten* Gliedern des Systems

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix}, \qquad \mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, \qquad \mathbf{c} = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_m \end{pmatrix}$$

Erweiterte Koeffizientenmatrix $(\mathbf{A} \mid \mathbf{c})$

$$(\mathbf{A} \mid \mathbf{c}) = \left(\begin{array}{c|c} \underbrace{\begin{matrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{matrix}}_{\mathbf{A}} & \underbrace{\begin{matrix} c_1 \\ c_2 \\ \vdots \\ c_m \end{matrix}}_{\mathbf{c}} \end{array}\right)$$

Quelle: Lothar Papula – Mathematische Formelsammlung – 9. Auflage – Vieweg Verlag

Wie wurde der Gleichungslöser umgesetzt?

Beispiel:

$$\begin{aligned} 2x_1 - 4x_2 - 10x_3 &= -38 \\ -1x_1 + 3x_2 - 2x_3 &= -1 \\ -3x_1 - 1x_2 + 3x_3 &= 14 \end{aligned} \quad \Rightarrow \quad \left.\begin{array}{rrr|r} 2 & -4 & -10 & -38 \\ -1 & 3 & -2 & -1 \\ -3 & -1 & 3 & 14 \end{array}\right. \quad \Rightarrow \quad \begin{array}{r} -2 \\ 1 \\ 3 \end{array}$$

Ablauf innerhalb der Methode:

- Parameter überprüfen
- obere Dreiecksmatrix bilden
- Determinante bilde und prüfen
  - det(A) = 0 → unendlich viele oder keine Lösungen
    - Prüfung auf Wiederspruch
      - Kein Wiederspruch → unendlich viele Lösungen
      - Wiederspruch → keine Lösung
  - det(A) ≠ 0 → nur eine Lösung
    - Lösungsvektor bilden

Wie wurde der Gleichungslöser umgesetzt?

Ablauf innerhalb der Methode:

- **Parameter überprüfen**
- obere Dreiecksmatrix bilden
- Determinante bilde und prüfen
  - det(A) = 0 → unendlich viele oder keine Lösungen
    - Prüfung auf Wiederspruch
      - Kein Wiederspruch → unendlich viele Lösungen
      - Wiederspruch → keine Lösung
  - det(A) ≠ 0 → nur eine Lösung
    - Lösungsvektor bilden

✓ Ist die Matrix mit absoluten Glieder ein Spaltenvektor ?
✓ Ist die Lösungsmatrix ein Spaltenvektor
✓ Stimmt die Zeilenanzahl der Matrix mit den absoluten Gliedern mit der des Lösungsvektors überein ?
✓ Stimmt die Anzahl absoluter Glieder mit der Anzahl Gleichungen in der Koeffizientenmatrix überein ?

Wie wurde der Gleichungslöser umgesetzt?

Ablauf innerhalb der Methode:

- Parameter überprüfen
- **obere Dreiecksmatrix bilden**
- Determinante bilde und prüfen
    - det(A) = 0 → unendlich viele oder keine Lösungen
        - Prüfung auf Wiederspruch
            - Kein Wiederspruch → unendlich viele Lösungen
            - Wiederspruch → keine Lösung
    - det(A) ≠ 0 → nur eine Lösung
        - Lösungsvektor bilden

$$\left(\begin{array}{ccc|c} 2 & -4 & -10 & -38 \\ -1 & 3 & -2 & -1 \\ -3 & -1 & 3 & 14 \end{array}\right) \Rightarrow \left(\begin{array}{ccc|c} 2 & -4 & -10 & -38 \\ 0 & 1 & -7 & -20 \\ 0 & 0 & -61 & -183 \end{array}\right)$$

Ablauf innerhalb der Methode:

- Parameter überprüfen
- obere Dreiecksmatrix bilden
- **Determinante bilde und prüfen**
    - **det(A) = 0 → unendlich viele oder keine Lösungen**
        - **Prüfung auf Wiederspruch**
            - **Kein Wiederspruch → unendlich viele Lösungen**
            - **Wiederspruch → keine Lösung**
    - **det(A) ≠ 0 → nur eine Lösung**
        - Lösungsvektor bilden

$$\left(\begin{array}{ccc|c} 2 & -4 & -10 & -38 \\ 0 & 1 & -7 & -20 \\ 0 & 0 & -61 & -183 \end{array}\right) \Rightarrow 2 * 1 * -61 = -122$$

Ablauf innerhalb der Methode:

- Parameter überprüfen
- obere Dreiecksmatrix bilden
- **Determinante bilde und prüfen**
  - **det(A) = 0 → unendlich viele oder keine Lösungen**
    - **Prüfung auf Wiederspruch**
      - **Kein Wiederspruch → unendlich viele Lösungen**
      - **Wiederspruch → keine Lösung**
  - **det(A) ≠ 0 → nur eine Lösung**
    - **Lösungsvektor bilden**

$$\left(\begin{array}{ccc|c} 2 & -4 & -10 & -38 \\ 0 & 1 & -7 & -20 \\ 0 & 0 & -61 & -183 \end{array}\right) \Rightarrow \left(\begin{array}{ccc|c} 1 & 0 & 0 & -2 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 3 \end{array}\right)$$

Gliederung

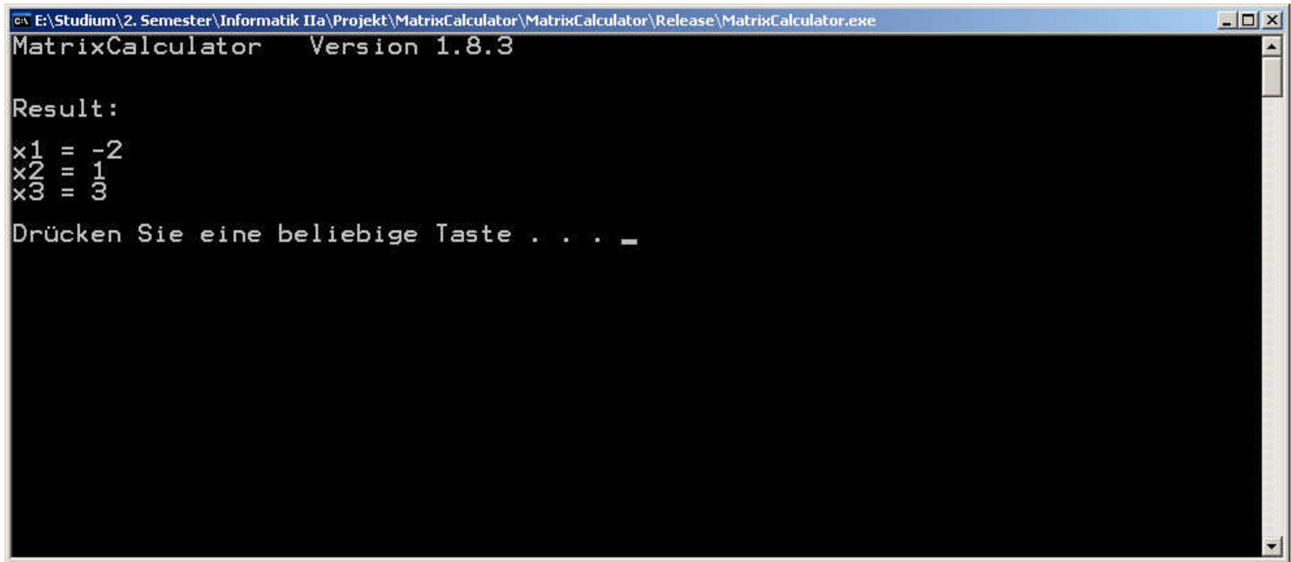
- Was war die Aufgabenstellung?
- **Wie wurde der Gleichungslöser umgesetzt?**
- Programmbeispiel ?

Gliederung

- Was war die Aufgabenstellung?
- Wie wurde der Gleichungslöser umgesetzt?
- **Programmbeispiel ?**

Programmbeispiel

$$\left(\begin{array}{rrr|r} 2 & -4 & -10 & -38 \\ -1 & 3 & -2 & -1 \\ -3 & -1 & 3 & 14 \end{array}\right) \Rightarrow \left(\begin{array}{rrr|r} 1 & 0 & 0 & -2 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 3 \end{array}\right)$$



Fragen ???