Lineare Gleichungssysteme

eine Unterrichtsreihe im Grundkurs Klasse 12 mit DERIVE

"Weniger rechnen - mehr verstehen"

war das Motto dieser Unterrichtseinheit. Bekanntlich wird gerade bei linearen Gleichungssystemen (LGS) häufig besonders stumpfsinnig gerechnet. Vernachlässigt werden Aspekte wie

- das Modellieren von Anwendungen (die auch auf LGS höherer Ordnung führen)
- Interpretieren von Lösungen
- Probleme im Umgang mit LGS, z.B. schlecht konditionierte LGS

Das Rechnen nimmt uns nun ein CAS (Computer-Algebra-System, hier DERIVE) ab, so daß die oben genannten und andere Aspekte weit mehr zu Tragen kommen können. Beim Einsatz von CAS ergeben sich allerdings neue Probleme, die vielfältig auf den Unterricht einwirken können:

- **Wie führt man ein CAS bei den Schülern ein?**
- **Wie ändern sich Aufgabenstellungen - bei Erarbeitungen, bei Hausaufgaben, bei Klausuren?**
- **Verstellt der Wunsch, mit einem CAS zu arbeiten, andere (bessere) Ansätze?**

- **Kann ein CAS auch beim Verständnis für die Algorithmen helfen?**
- **Wie werden Begriffsbildungen durch ein CAS beeinflußt? Wie weit wird der Lehrer dabei neuen Zwängen unterworfen?**
- **Kann man im Unterricht nun auf das (bisherige) Üben verzichten?**
- **Wie sichert man die bei der Arbeit am Computer entstehenden Ergebnisse?**
- **Wie kann der Lehrer den Erfolg der Arbeit kontrollieren?**

- **Gibt es Bezüge zum Informatikunterricht, die sich ausnutzen lassen?**

Einigen dieser Fragen soll im folgenden nachgegangen werden. Die Einbettung der CAS-Anwendung in die übrige Unterrichtseinheit kann in dem Beitrag nur angedeutet werden, jedoch sollen die Abhängigkeiten und Auswirkungen verdeutlicht werden. Außerdem werden Bezüge zur Informatik hergestellt.

Übersicht über die Unterrichtseinheit

- Probleme, die auf LGS führen
- Woher sind LGS schon bekannt? - Vektorrechnung!
- Was ist ein LGS? Was ist gesucht? Was ist bei anderen Gleichungsarten anders?
- Konstruiere ein (3,3)-LGS mit vorgegebener Lösungsmenge!
- LGS als Vektorsystem, Matrixschreibweise
- Ein günstiges LGS (in "Idealform") weist den Weg (z.B. Materialverflechtung)
- Erste Lösungsansätze von Hand bei einem LGS mit bekannter Lösungsmenge, einige Elementarumformungen
- **DERIVE hilft beim Verwirklichen der Lösungsideen, Umformungsregeln**
- das Ablesen der Lösungsmenge aus dem von DERIVE gelieferten Ergebnis
- weitere Anwendungen: Modellbilden - rechnen lassen - interpretieren
- schlecht konditionierte LGS

Geeignete Derive-Funktionen für die schrittweise Lösung von LGS

Denkt man an das für die Oberstufe relevante Gauß-Verfahren, so kommen in erster Linie folgende Funktionen in Frage:
SCALE_ELEMENT, SUBTRACT_ELEMENTS, FORCE0, PIVOT und für die schnelle Lösung: ROW_REDUCE. Diese Funktionen hat DERIVE in der Datei VECTOR.MTH gespeichert und müssen dazugeladen werden.

A sei eine Matrix

SCALE-ELEMENT(A,i,s) multipliziert die i-te Zeile von A mit dem Term s, alle anderen Elemente von A bleiben erhalten. *Zeile i:= s * Zeile i*

Beispiel:

	4	7	-1		4	7	-1
A =	5	7	8	SCALE-ELEMENT(#1,3,1/2) =	5	7	8
	2	6	3		1	3	3/2

SUBTRACT_ELEMENTS(A,i,j) berechnet die neue Zeile i durch Subtraktion der Zeile j von A von der i-ten Zeile . Alle anderen Elemente von A bleiben erhalten.

Zeile i:=Zeile i - Zeile j

Beispiel:

	4	7	-1		4	7	-1
A =	5	7	8	SUBTRACT_ELEMENT(A,2,3) =	3	1	5

2	6	3		2	6	3

FORCE0(A,i,j,p) bringt das Element A(i,j) auf 0 durch Subtraktion eines geeigneten Vielfachen der Pivotzeile p von der Zeile i. Alle anderen Elemente von A bleiben erhalten.
Zeile i:=Zeile i - Vielfaches von Pivotzeile p

Beispiel:

	1	7/4	-1/4		1	7/4	-1/4
A =	5	7	8	FORCE0(A,2,1,1) =	0	-7/4	37/4
	2	6	3		2	6	3

Es wird Element A(2,1) auf 0 gebracht. Pivotzeile war Zeile 1. Das Vielfache von Zeile 1 war 5, für das Element A(2,1) wurde gerechnet: A(2,1) = 5-5*1 =0 und außerdem:
A(2,2)=7-5*7/4 =28/4-35/4 =-7/4
A(2,3)=8-5*(-1/4) =32/4+5/4 =37/4

PIVOT(A,i,j) bringt das Element A(i+1,j) auf 0 durch Subtraktion eines geeigneten Vielfachen der Zeile i. Alle anderen Elemente von A bleiben erhalten.
Zeile(i+1):=Zeile(i+1) - Vielfaches von Zeile i

Beispiel:

	1	7/4	-1/4		1	7/4	-1/4
A =	5	7	8	PIVOT(A,2,1) =	5	7	8
	2	6	3		0	16/5	-1/5

Es wird Element A(3,1) auf 0 gebracht. Zeile i war Zeile 2. Das Vielfache von Zeile 2 war 2/5, für das Element A(3,1) wurde gerechnet: A(3,1) = 2-2/5*5 =0 und außerdem:
A(3,2)=6-7*2/5 =30/5-14/5 =16/5
A(3,3)=3-8*2/5 =15/5-16/5 = - 1/5

Hinweis: Die Funktionen FORCE0 und PIVOT unterscheiden sich geringfügig und können leicht miteinander verwechselt werden. Man sollte daher nicht beide im Unterricht verwenden! FORCE0 scheint mir für den Übergang von der Handrechnung zum Computereinsatz geeigneter.
PIVOT(A,i,j) bringt das Element A(i+1,j) auf 0 mittels der Hilfszeile i
FORCE0(A,i,j,p) bringt das Element A(i,j) auf 0 mittels der zusätzlich anzugebenden Hilfszeile p.
Es gilt also PIVOT(A,i,j) = FORCE(A,i+1,j,i).

ROW_REDUCE(A,B), dabei ist A die Koeffizientenmatrix des LGS, B ist die Matrix für die rechte Seite. Das LGS ist dabei in der Form Ax=B gegeben. - ROW_REDUCE fügt die beiden Matrizen aneinander und bestimmt die Lösungsmenge des LGS durch Umformung auf die Einheitsmatrix mittels elementarer Zeilenumformungen der erweiterten Matrix

- Zahl*Matrix,
- Addition oder Subtraktion zweier Zeilen

- dabei ändert sich die Lösungsmenge des LGS nicht!).

Die Lösung eines LGS mit der Funktion ROW_REDUCE

ROW_REDUCE ist zunächst ein nützliches Werkzeug für den Lehrer und bei der hier verfolten Konzeption erst in einer späteren Unterrichtsphase für die Arbeit mit den Schülern interessant.

Beispiel: Wir betrachten das LGS
4t+7s = -1
5t+7s = 8
2t+6s = 3, also

$$A = \begin{matrix} 4 & 7 \\ 5 & 7 \\ 2 & 6 \end{matrix} \qquad B = \begin{matrix} -1 \\ 8 \\ 3 \end{matrix}$$

$$\text{ROW\_REDUCE}(A,B) = \begin{matrix} 1 & 0 & 9 \\ 0 & 1 & -37/7 \\ 0 & 0 & 1 \end{matrix}$$

Das LGS wurde also umgeformt zu:
1t+0s = 9
0t+1s = -37/7
0t+0s = 1 Gleichung 3 ist unerfüllbar. Keine Lösung!

Die Lösung eines LGS über eine Liste und der Anweisung SOLVE

Unter Verwendung der Matrizenschreibweise kann man die Koeffizientenmatrix A und die Ergebnismatrix B eingeben und mit der Funktione ROW_REDUCE(A,B) bearbeiten:

$$\text{ROW\_REDUCE}\left(\begin{bmatrix} 4 & 7 & -1 \\ 5 & 7 & 8 \\ 2 & 6 & 3 \end{bmatrix}, \begin{bmatrix} 2 \\ 3 \\ 4 \end{bmatrix}\right)$$

$$\begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & \frac{8}{9} \\ 0 & 0 & 1 & \frac{2}{9} \end{bmatrix}$$

Man kann das LGS jedoch auch als Liste eingeben:

[4x+7y-z=2, 5x+7y+8z=3, 2x+6y+3z=4]   und die Lösung mit SOLVE ermitteln:
[x=-1, y=8/9, z=2/9].

Allerdings treten hier Schwierigkeiten auf, wenn keine eindeutige Lösung vorliegt. Zur Entwicklung des Gauß-Verfahrens ist dieser Weg nicht geeignet. Als Black-Box zur Lösung kleiner LGS kann man den Weg benutzen.

Methodisch-didaktische Überlegungen zum Einsatz von DERIVE
Anforderungen an Lehrer und Schüler

Zum Unterricht über lineare Gleichungssysteme (LGS) und deren Bearbeitung mit dem Gauss-Verfahren

(1) Verschiedene Anwendungen sollen zunächst zeigen, daß LGS in der Praxis häufig auftreten. Dabei geht es nicht nur um LGS mit zwei oder drei Variablen, vielmehr sollen die Beispiele bewußt auch auf LGS mit mehr als drei Variablen und solche mit unterschiedlicher Anzahl von Variablen und Gleichungen führen. Hauptziel ist hier das Modellieren von Anwendungssituationen (Materialverflechtung, Populationsdynamik, magische Quadrate usw.) mit dem Aufstellen der LGS, die alle die Form Ax=B haben. Die jeweilige Lösung wird aufgeschoben oder sogar vorgegeben (der Lehrer benutzt ROW_REDUCE; ggf. Kontrollrechnung der Schüler). Vermutlich bringen die Schüler

nur Beispiele aus der Analytischen Geometrie, wie etwa die Schnittpunktberechnung zweier Geraden oder einer Geraden mit einer Ebene.

In dieser Phase erhalten die Schüler auch die Aufgabe, ein LGS zu vorgegebener Lösungsmenge zu konstruieren, z.B. zu L={(1,2,3)} (Beitrag zu **"LGS verstehen"** !).

Begriffe: Matrix als rechteckiges Zahlenschema, Koeffizientenmatrix $A_{(m,n)}$, erweiterte Koeffizientenmatrix ($A_{(m,n)}$, $B_{(m,1)}$), Zeilen, Spalten, Zeilenvektor, Spaltenvektor, Doppelindizes für Zeilen- und Spaltenpositionen, Anzahl der Variablen n, Anzahl der Gleichungen m, Lösung eines LGS.
Declare - Matrix, Row - Column

(2) Unter den Beispielen ist auch mindestens ein LGS, das von vornherein eine günstige Gestalt hat: In der Hauptdiagonalen lauter 1-en und darunter (möglicherweise auch darüber!) lauter 0-en. Hierfür eignet sich z.B. ein Problem zur Materialverflechtung, nämlich die Bereitsstellung von Rohstoffen aufgrund einer Bestellung von Endprodukten.
Damit ist das Umformungsziel für LGS vorgegeben - nämlich gerade eine so günstige Gestalt des LGS (Endform) herbeizuführen! Die Darstellungen der LGS erfolgen ausführlich und in abkürzender Matrizenschreibweise.

(3) Welche Umformungen mit den einzelnen Gleichungen sind geeignet, um die günstige Endform zu erzeugen? *Wir kommen auf die zulässigen Elementarumformungen* - im Grundkurs ohne Beweis.

- Multiplikation einer Gleichung mit einer Zahl ungleich Null,
- Ersetzen einer Gleichung durch Addition (Subtraktion) einer anderen Gleichung.

Auf diese Weise lassen sich die gewünschten 1-en und 0-en erzeugen. Die jeweils neuen LGS werden in Matrizenform dargestellt, gelegentlich wird an die ausführliche Schreibweise erinnert.

Zu diesem Zeitpunkt stellt sich die Frage des Einsatzes von CAS (DERIVE) mit folgenden Möglichkeiten:
(a) Sofortige Bestimmung der Endform mit Hilfe der Funktion ROW_REDUCE
(b) Schrittweise elementare Umformungen mit Hilfe der Funktionen
- (b1) SCALE_ELEMENT und SUBTRACT_ELEMENTS oder
  (etwas schneller) mit
- (b2) SCALE_ELEMENT und FORCE0 oder
  (noch schneller) mit
- (b3) PIVOT

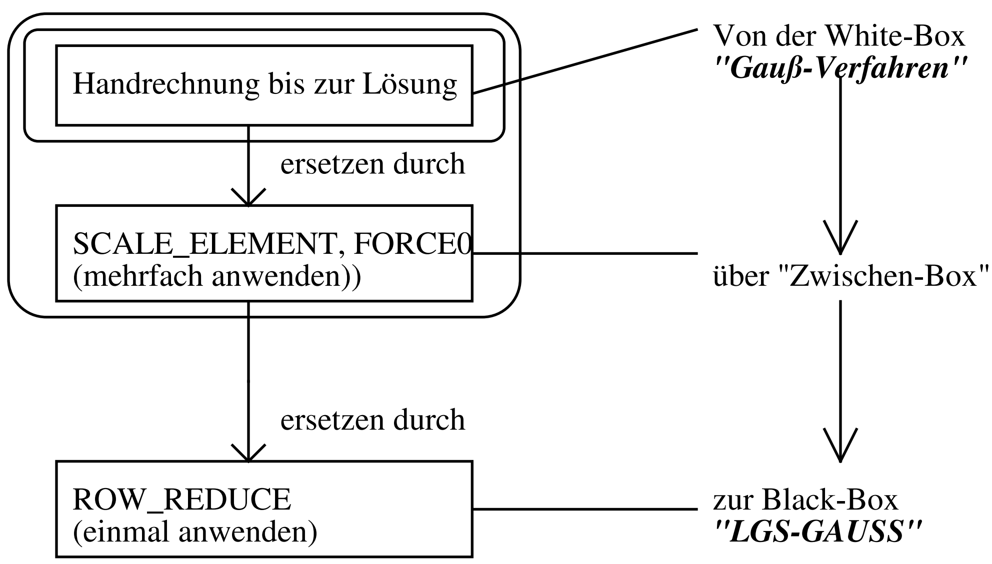
(c) Lösung des LGS durch Eingabe als Liste und Verwendung von SOLVE.

Dieser Weg kommt jedoch hier nicht in Frage, da es um das Gauß-Verfahren geht.

Weiterhin ergibt sich die Frage, wieweit man den Gauß-Algorithmus durch Handrechnungen vorbereitet werden soll., etwa
(d) Durchrechnung eines oder mehrerer Beispiels mit Handrechnung

Wenn ein Verständnis für den Gauß-Algorithmus gefordert wird, man die Vorteile des Computereinsatzes nutzen und schnell zum Rechnereinsatz vordringen will, empfiehlt sich die Reihenfolge • Handrechnung an einem Beispiel (weiteres Beispiel als Hausarbeit) • (b2) SCALE_ELEMENT und FORCE0, • (a) ROW_REDUCE

Diese Reihenfolge läßt sich entsprechend dem White-Box / Black-Box -Prinzip so veranschaulichen:



Figur Die Black-Box "LGS-GAUSS"

Die mathematischen Grundlagen und das erste Verständnis für das Gauß-Verfahren werden in der White-Box "Gauß-Verfahren" erzeugt. Dabei bezieht sich das Verstehen zunächst auf den

- Ablauf des Verfahrens unter Anwendung der Elementarumformungen, dann
- auf die Zusammenfassung von Arbeitsschritten (in 2 Stufen) und schließlich
- auf das Verstecken all dieser Abläufe in einem Baustein, der Black-Box "LGS-GAUSS".

Dabei wird gelernt, wie man die Zwischenergebnisse und schließlich das Endschema (letzte Matrix) mit den 3 grundsätzlich möglichen Lösungsfällen interpretiert (keine

Lösung, genau eine Lösung, unendlich viele Lösungen). Der Fall "endlich viele Lösungen, n>1) wird innerhalb von Anwendungsproblemen, bei denen eingeschränkte Grundmengen vorliegen, diskutiert.

Die schrittweise Lösung eines LGS mit den Funktionen SCALE_ELEMENT und FORCE0

Aufgabe: Die Schnittpunktberechnung zweier Geraden g1 und g2 führt nach Gleichsetzen auf das lineare Gleichungssystem

$4t+7s = -1$ **"Verstehen"**: Wie heißen die Geradengleichungen?
$5t+7s = 8$
$2t+6s = 3$

Die erweiterte Koeffizientenmatrix wird in DERIVE eingegeben mit
DECLARE MATRIX Zeilenanzahl=3 Spaltenanzahl=3

DERIVE-Protokoll

Augenblickliche Matrix	**Anwendung einer Funktion auf diese Matrix**
1: $\begin{bmatrix} 4 & 7 & -1 \\ 5 & 7 & 8 \\ 2 & 6 & 3 \end{bmatrix}$	**G1:=G1*1/4** ***scale_element(#1,1,1/4)*** Hauptdiagonalelement a(1,1) auf 1 bringen, dazu Zeile 1 mit 1/4 multiplizieren
2: $\begin{bmatrix} 1 & \frac{7}{4} & -\frac{1}{4} \\ 5 & 7 & 8 \\ 2 & 6 & 3 \end{bmatrix}$	**G2:=G2 -5*G1** *force0* ***(#2,2,1,1)*** Element an Position (2,1) auf 0 bringen, 1.Zeile ist Pivotzeile.
3: $\begin{bmatrix} 1 & \frac{7}{4} & -\frac{1}{4} \\ 0 & -\frac{7}{4} & \frac{37}{4} \\ 2 & 6 & 3 \end{bmatrix}$	**G3:=G3 -2*G1** ***force0(#3,3,1,1***) Element an Position (3,1) auf 0 bringen, Zeile 1 ist Pivotzeile
4: $\begin{bmatrix} 1 & \frac{7}{4} & -\frac{1}{4} \\ 0 & -\frac{7}{4} & \frac{37}{4} \\ 0 & \frac{5}{2} & \frac{7}{2} \end{bmatrix}$	**G2:=G2*(-4/7)** ***scale_element(#4,2,-4/7)*** Hauptdiagonalelement a(2,2) auf 1 bringen, dazu Zeile 2 mit -4/7 multiplizieren

$$5: \quad \begin{bmatrix} 1 & \frac{7}{4} & -\frac{1}{4} \\ 0 & 1 & -\frac{37}{7} \\ 0 & \frac{5}{2} & \frac{7}{2} \end{bmatrix}$$

G3:=G3 -5/2*G2

force0(#5,3,2,2)
Element an Position (3,2) auf 0 bringen,
die 2.Zeile ist Pivotzeile

$$6: \quad \begin{bmatrix} 1 & \frac{7}{4} & -\frac{1}{4} \\ 0 & 1 & -\frac{37}{7} \\ 0 & 0 & \frac{117}{7} \end{bmatrix}$$

G1:=G1-7/4*G2

force0(#6,1,2,2)
Element an Position (1,2) auf 0 bringen,
die 2.Zeile ist Pivotzeile

$$7: \quad \begin{bmatrix} 1 & 0 & 9 \\ 0 & 1 & -\frac{37}{7} \\ 0 & 0 & \frac{117}{7} \end{bmatrix}$$

G3:=G3*7/117

scale_element(#7,3,7/117)
Hauptdiagonalelement a(3,3) auf 1 bringen, dazu
Zeile 3 mit 7/117 multiplizieren.

$$8: \quad \begin{bmatrix} 1 & 0 & 9 \\ 0 & 1 & -\frac{37}{7} \\ 0 & 0 & 1 \end{bmatrix}$$

Die gewünschte Form ist erreicht,
die Lösungsmenge kann abgelesen
werden. - Das umgeformte LGS lautet:

1t+0s = 9
0t+1s = -37/7
0t+0s = 1 Gleichung 3 ist unerfüllbar. Keine Lösung! Kein Schnittpunkt vorhanden!

Die Anforderungen an den Lehrer

Erwägt der Lehrer Computereinsatz, so gibt es für ihn u.a. folgende Fragen zu klären.

Einsatz eines speziellen Unterrichtsprogramm zur Bearbeitung linearer Gleichungssysteme mit didaktisch-methodischen Komponenten (z.B. Literatur [2])

dieser Weg wird hier angesichts der Ausrichtung der Tagung nicht weiter verfolgt

Einsatz eines CAS (z.B. von DERIVE)

Für den Lehrer besteht nun zunächst die Aufgabe, zu erforschen, welche Möglichkeiten zur Lösung von LGS DERIVE zur Verfügung stellt. Er wird feststellen, daß man ein LGS

1. durch Benutzung der Funktion ROW_REDUCE sofort auf die Endform bringen kann,
2. über Matrizenoperationen (inverse Matrix - falls existent) lösen kann,
3. über diverse Zwischenschritte (unter denen auch die gängigen Elementarumformungen sind) bearbeiten kann.

Den Intentionen der meisten Lehrpläne folgend (und dem eigenen mathematischen Gewissen) wird er sich für 3. entscheiden und diesen Weg in seinen Unterrichtsgang einbauen. Das bedeutet eine Analyse der Funktionen SCALE_ELEMENT, SUBTRACT_ELEMENTS, FORCE0 und PIVOT.

Mit Bausteinen arbeiten - Parametrisierung

Das besondere methodische Problem liegt hier darin, daß all diese Funktionen parametrisiert sind. Aus Sicht der Informatik handelt es sich um Bausteine, die in einer Modulbibliothek (VECTOR.MTH) zur Verfügung stehen, sofern man dieses Modul vorher lädt (*TRANSFER - LOAD - UTILITY (oder DERIVE) - VECTOR.MTH)*. **An dieser Stelle trifft man sich mit heutigen Intentionen des Informatikunterrichts, die auch als Ziele des Mathematik-Unterrichts zu fordern sind**!

- Zerlegen eines komplexen Problems in Teilprobleme
- Benutzung fertiger Bausteine für Teilprobleme
  (oder ggf. auch Konstruieren von Bausteinen für Teilprobleme)

- Zusammensetzung zur Gesamtlösung

Solche Bausteine (Prozeduren oder Funktionen) haben auch in der Informatik in der Regel Parameter in Form von Eingangsparametern und Ausgangsparametern. Bei den benötigten DERIVE-Funktionen handelt es sich um Funktionen mit Eingangsparametern, die Funktionen liefern dann einen Wert, mit dem man weiterarbeiten kann.

Betrachten wir ein Beispiel:

SCALE_ELEMENT(#4, 2, -4/7) hat drei Eingangsparameter

#4 benennt das zu bearbeitende Objekt, in diesem Fall z.B. eine Matrix A
2 gibt die Nummer der zu bearbeitende Zeile der Matrix A bekannt
-4/7 ist der Faktor, mit dem die 2.Zeile multipliziert wird

SCALE_ELEMENT(#4, 2, -4/7) liefert die Matrix B, die nun zur weiteren Bearbeitung zur Verfügung steht.

Wenn man diesen Befehl abschickt, liefert DERIVE zunächst die Ausgabe:

$$\texttt{scale\_element}\left(\begin{bmatrix} 1 & \frac{7}{4} & -\frac{1}{4} \\ 0 & -\frac{7}{4} & \frac{37}{4} \\ 0 & \frac{5}{2} & \frac{7}{2} \end{bmatrix}, 2, -4/7\right)$$

Diese zeigt die Einsetzung des Objekts #4.

Erfahrungsgemäß macht es den Schülern kaum Schwierigkeiten, mit fertigen Bausteinen und ihren Parametern umzugehen, sofern eine eindeutige Beschreibung - am besten mit einem Beispiel - vorliegt. So arbeitet z.B. auch der Berliner Informatik-Lehrplan schon im Anfangsunterricht mit dem Bausteinansatz und der Einbetttung von Bausteinen in komplexe Problemstellungen (Literatur [ ],[ ]).

Bausteine dokumentieren

Die Beschreibungen der hier benötigten Funktionen im DERIVE-Handbuch sind allerdings für den Schulgebrauch, zumal im Grundkurs wenig geeignet. Der Lehrer wird die Funktionen besser dokumentieren müssen - auch in Abstimmung mit den

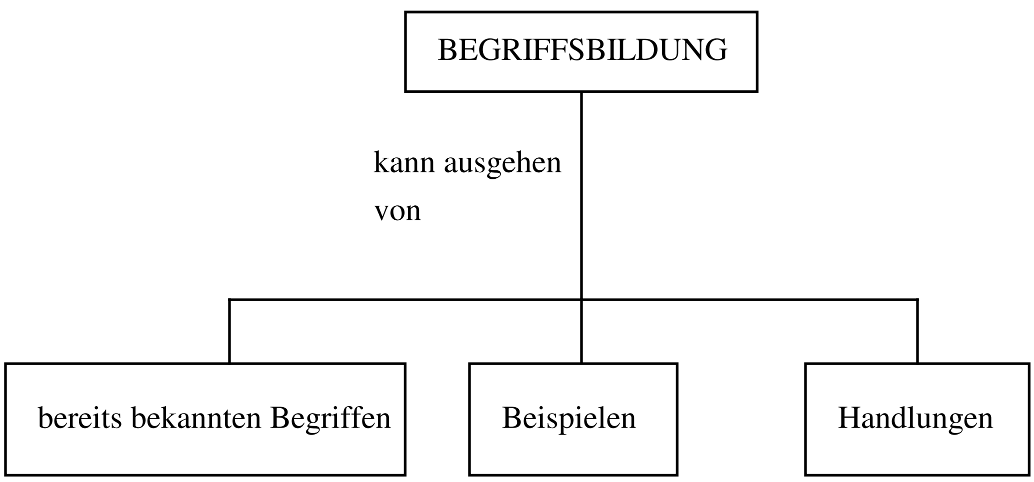
Vorkenntnissen seines Kurses. Beispiele für solche Dokumentationen, die insgesamt eine "Formelsammlung" für den Schüler bilden, finden sich oben.

Die DERIVE-Funktionen müssen vom Lehrer so dokumentiert werden, daß sie der Schüler selbständig und zielgerichtet verwenden kann! ***Eine DERIVE-FORMEL-SAMMLUNG entsteht!***

Begriffslernen

Durch die Einführung von Begriffen

- kann die Komplexität von Sachverhalten reduziert werden
- können neue Gegenstände in Zusammenhänge eingeordnet werden
- können allgemeine Aussagen leichter formuliert werden
- wird das Lehren und Lernen vereinfacht



Die Verwendung von DERIVE-Funktionen und ihrer Parameter führt zwangsläufig dazu, daß im Unterricht passende Begriffe eingeführt werden müssen. Insofern steuern CAS auch hier die Unterrichtsdidaktik und -Methodik und es kann sein, daß der Lehrer an dieser Stelle abhängig wird vom CAS.

Zum Eingeben von Matrizen werden die Optionen *DECLARE MATRIX* benötigt. Daraufhin fordert DERIVE Zahlen für *ROW* (Zeilenanzahl) und *COLUMN* (Spaltenanzahl). Anschließend werden die Positionen der einzugebenden Matrixelemente mit Doppelindizes angegeben. Bei allen Handlungen in Zusammenhang mit der Eingabe einer Matrix wird also immer wieder an die Fachbegriffe erinnert. Gleichzeitig wird deutlich, daß zu einer Matrix immer drei Angaben gehören

- Zeilenanzahl
- Spaltenanzahl
- Matrixelemente

In der Informatik entspricht das der Einführung der Datenstruktur "matrix":

```
TYPE matrix = RECORD
                  wert: ARRAY[1..maxgrad,1..maxgrad] OF REAL;
                  zeilenanzahl: INTEGER;
                  spaltenanzahl: INTEGER;
              END;
```

Die in Bausteinen enthaltenen Algorithmen analysieren, neue entwerfen

In einem stärker algorithmisch ausgerichteten Unterricht wird man zumindest in einige Bausteine "hineinsehen".

Was bedeutet das hier? Dargestellt in einem Struktogramm passiert bei *SCALE_ELEMENT(#4, 2, -4/7)* folgendes:

Eingabe der Elemente a(i,k) einer Matrix A		(Beispiel A=#4, i=1..3, k=1..3)
Eingabe des Multiplikators s		(hier s= -4/7)
Eingabe des Zeilenindexes z		(hier z=2)
Kopiere die Matrix A auf die Matrix B B:=A;		
für den Spaltenindex j von 1 bis k		(hier j von 1 bis 3)
	Berechne die neuen Elemente von Zeile z b(z, j) := s * a(z, j)	
Ausgabe der Elemente b(i,k) von Matrix B		

Abb. : Skalarmultiplikation s * Zeilenvektor
("Eingeben" bedeutet beim Funktionsaufruf "Übergeben der Werte")

Auch programiertechnische Aspekte lassen sich ansprechen, etwa indem man sich das Modul VECTOR.MTH mit seinen Bausteinen auf den Bildschirm ausgeben läßt und nun einzelne Bausteine analysiert.

```
"File VECTOR.MTH, copyright (c) 1990 by Soft Warehouse, Inc."
i_ := [1, 0, 0]
```

```
3:    j_ := [0, 1, 0]
4:    k_ := [0, 0, 1]  usw.
usw.

12:
SCALE_ELEMENT (v,i,s):=
VECTOR(IF (m_ =i,s ELEMENT (v,i),ELEMENT(v,m_)),m_,DIMENSION (v))
```

Allerdings stellt sich angesichts der Komplexit mancher hier definierten Funktionen die Frage, wie weit ein Eindringen erwünscht ist. Beim Grundkurs Mathematik sicher nicht, beim Leistungskurs Mathematik eventuell. **Für den Informatikunterricht ergeben sich hier interessante Ansätze zum funktionalen Programmieren.**

Das Üben

- **Kann man im Unterricht nun auf das (bisherige) Üben verzichten?**

Weitgehend ja!

Das Üben erhält hier einen anderen Charakter und einen anderen Stellenwert. Hier geht es um

- das Gewinnen von Sicherheit im Umgang mit den DERIVE-Funktionen
- das Üben von Dokumentationen
- die Überprüfung des Verständnisses
- die zielgerichtete Anwendung von DERIVE-Funktionen mit dem Ziel eines optimalen (kurzen, eleganten) Lösungsweges

Beispiel für die Überprüfung des Verständnisses

Arbeitsauftrag: Kommentieren Sie die Punkte 1: bis 7:! Die Anwendungsaufgabe hieß:
Bestimmen Sie den Schnittpunkt der Geraden g und h mit

$$g: \quad r = \begin{matrix} 2 \\ -8 \\ 0 \end{matrix} + t \begin{matrix} 2 \\ 1 \\ 0 \end{matrix} \qquad h: \quad r = \begin{matrix} 0 \\ -4 \\ -5 \end{matrix} + s \begin{matrix} 0 \\ -1 \\ 1 \end{matrix}$$

Kommentare

$$1: \quad \begin{bmatrix} 2 & 0 & -2 \\ 1 & 1 & 4 \\ 0 & -1 & -5 \end{bmatrix}$$

┌┌ 2   0   -2 ┐   ┐

2:	$\text{SCALE\_ELEMENT}\left(\begin{bmatrix} 1 & 1 & 4 \\ 0 & -1 & -5 \end{bmatrix}, 1, \frac{1}{2}\right)$

3: $\begin{bmatrix} 1 & 0 & -1 \\ 1 & 1 & 4 \\ 0 & -1 & -5 \end{bmatrix}$

4: FORCE0 $\left[\begin{bmatrix} 1 & 0 & -1 \\ 1 & 1 & 4 \\ 0 & -1 & -5 \end{bmatrix}, 2, 1, 1\right]$

5: $\begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & 5 \\ 0 & -1 & -5 \end{bmatrix}$

6: FORCE0 $\left[\begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & 5 \\ 0 & -1 & -5 \end{bmatrix}, 3, 2, 2\right]$

7: $\begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & 5 \\ 0 & 0 & 0 \end{bmatrix}$

Hinweis: Bei der Zusammenstellung dieses Arbeitsbogens merkt der Lehrer, daß das DERIVE-Protokoll (TRANSFER - SAVE - DERIVE (oder PASCAL)) in anderer Form gespeichert wird, als die Bildschirmausgabe erfolgte - hier nämlich in Form einer Liste.

```
[[2,0,-2],[1,1,4],[0,-1,-5]]
SCALE_ELEMENT([[2,0,-2],[1,1,4],[0,-1,-5]],1,1/2)
[[1,0,-1],[1,1,4],[0,-1,-5]]
FORCE0([[1,0,-1],[1,1,4],[0,-1,-5]],2,1,1)
[[1,0,-1],[0,1,5],[0,-1,-5]]
FORCE0([[1,0,-1],[0,1,5],[0,-1,-5]],3,2,2)
[[1,0,-1],[0,1,5],[0,0,0]]
```

Wieder eine Schwierigkeit, wenn man die Schüler hier selbständig arbeiten läßt. So ergäbe sich hier der Zwang, diese andere Schreibweise einzuführen. Allerdings ließe sich das umgehen, wenn man den PRINT-Befehl in DERIVE benutzt und die Daten nicht auf den Drucker, sondern in eine Datei leitet: TRANSFER - PRINT - FILE.

Lernziele zur Unterrichtseinheit
"Lineare Gleichungssysteme nach Gauß"

Die nun folgende Liste zeigt Lernzziele, die man in einer Unterrichtseinheit über LGS nach dem Gauß-Verfahren anstreben kann. Welche der Ziele ausgewählt werden, richtet sich besonders nach der Art des Einsatzes von DERIVE.

Die doppelt gerahmten Ziele sollen in jedem Fall erreicht werden. Sie stellen m.E. das Minimalprogramm einer Unterrichtseinheit dar.
Die einfach gerahmten Ziele können hinzukommen.
Die kursiv markierten Ziele machen den Handlungsspielraum aus, den man durch den Einsatz von DERIVE erhält, abhängig von den verwendeten Funktionen ROW_REDUCE bzw. SCALE_ELEMENT und FORCE0.

Lernziele aus mathematischer Sicht

Probleme, die auf LGS führen, kennen und ihre Modellierung durch LGS erläutern können. Ansätze rekonstruieren können.
- aus der Analysis
- aus der Analytischen Geometrie, Abbildungsgeometrie
- aus der Stochastik
- aus der Wirtschaft oder anderen Anwendungen

Den Begriff des linearen Gleichungssystems erläutern können.

Begriffsbildungen in Zusammenhang mit LGS kennen (und erläutern können): Vektor, Matrix, (erweiterte) Koeffizientenmatrix,
Lösungsvektor, ...

LGS zu vorgegebener Lösungsmenge aufstellen können.

LGS aus Anwendungssituationen aufstellen können.

Die Zielsetzung des Gauß-Algorithmus erläutern können.

Die Durchführung des Gauß-Algorithmus erläutern und
eventuellen Zeilen- und Spaltentausch begründen können.

Die Lösungsmenge eines LGS mit Hilfe von Elementarumformungen für einfache Fälle von Hand berechnen und die "Zwischen-LGS" deuten können.

Die Lösungsmenge eines LGS mit Hilfe von DERIVE (scale_element, force0) schrittweise gemäß den Elementarumformungen berechnen und die "Zwischen-LGS" deuten können.

Die Umformung eines LGS mit dem Befehl row_reduce durchführen können.

Die Arbeit mit DERIVE problembezogen strukturieren und dokumentieren können.

Die Bedeutung der Parameter der DERIVE-Funktionen scale_element, force0, row_reduce erläutern können.

Die zu den DERIVE-Funktionen scale_element, (force0, row_reduce) gehörenden Algorithmen formulieren können (z.B. Struktogramm).

Die Lösungsmenge *je einer* Gleichung eines LGS von der Lösungsmenge des LGS unterscheiden können!

Das bei den Elementarumformungen entstehende Endschema interpretieren und problemgemäß auswerten können.

Das Ablesen der Lösungsmenge aus dem Endschema begründen können.

Endschemata mit den unterschiedlichen Möglichkeiten für die Lösungsmengen notieren können.

Lösungskriterien für LGS unter Verwendung des Matrixrang-Begriffs (oder anders) formulieren können.

Schlecht konditionierte LGS entwerfen und die Lösungsproblematik aufzeigen können.

Andere Lösungsmethoden von LGS als den Gauß-Algorithmus kennen.
(Iteration)

Den Zusammenhang zwischen der Lösung von LGS und der Bestimmung inverser Matrizen erläutern können.

Die Gleichung AX = B formal nach X unter Verwendung der
Inversen A^{-1} von A auflösen können.

(GK und LK beachten!!!)

Lernziele für die Informatik

Die von DERIVE für Matrizen bzw. Vektoren verwendete Datenstruktur erläutern können.

Die DERIVE-Funktionen als Prozeduren verstehen und dazugehörige Algorithmen (in PASCAL) konstruieren können.

Die für LGS verwendeten DERIVE-Funktionen analysieren können (Daten, Parameter, Algorithmen).

Die (für LGS vorhandene) Oberfläche von DERIVE beurteilen können.

Zusammenfassung:

Die Betrachtungen zeigen am Beispiel der Behandlung linearer Gleichungssysteme in der Sekundarstufe 2, daß durch Computereinsatz und speziell den Einsatz von Computeralgebrasystemen ein weiter zusätzlicher Handlungsspielraum entsteht, der den bisherigen Unterricht über dieses Gebiet wesentlich verändern wird.

Literatur

[1] Vorläufiger Rahmenplan, Fach Informatik, Senatsverwaltung für Schule, Berufsbildung und Sport, Berlin, gültig ab Schuljahr 1993/94

[2] Lehmann, E.: Projekte im Informatik-Unterricht, Dümmler-Verlag 1995