



VEKTORRECHNUNG

MIT EINEM CAS-RECHNER

Jürgen Geißelbrecht

Ein Unterrichtsbehelf zum Einsatz moderner Technologien
im Mathematikunterricht

Die Verwendung eines CAS kann das traditionelle Spektrum in der Vektorrechnung in zweierlei Hinsicht erweitern:

| INHALTLICH | FORMAL |
|--|---|
| <ul style="list-style-type: none"> ▪ Verzicht auf „schöne“ Ergebnisse, dadurch realitätsnäher. ▪ Komplexere Aufgabenstellungen, ermöglicht durch Automatisierung von immer wiederkehrenden Algorithmen. ▪ Größerer ästhetischer Reiz durch verstärkte Hinwendung zu räumlichen Problemen. | <ul style="list-style-type: none"> ▪ Stärkere Betonung des Problemlösens durch Konzentration auf den Lösungsweg. ▪ Notwendigkeit planmäßigen und strukturierten Arbeitens. ▪ Spezialisieren und Verallgemeinern durch Verwenden von Skripts. ▪ Zeitliche Möglichkeit des Probierens und der Überprüfung von Lösungen. ▪ Kennenlernen von verschiedenen Exaktheitsniveaus bei Begründungen. |

Eine entscheidende Rolle spielen dabei Skripts (Text-Dateien). Diese dienen dazu Rechenabläufe zu speichern, sind aber nicht so komplex wie Programme und benötigen auch keine Programmierbefehle. Skripts können editiert und abgearbeitet werden.

Anhand von zwei im Unterricht ausprobierten Themen wird dies nun näher ausgeführt.

THEMA 1: BEGRÜNDEN UND BEWEISEN AUF UNTERSCHIEDLICHEN NIVEAUS

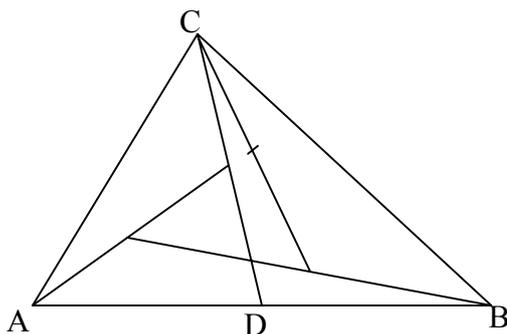
Die Grundidee ist hier folgende: Anstatt dem bezuglosen Berechnen von Schnittpunkten etc. geht es hier um die Frage, wie eine bestimmte Eigenschaft (Orthogonalität, Kollinearität, Äquidistanz usw.) nachgeprüft werden kann.

1.1 EINE WENIG BEKANNTE EIGENSCHAFT VON DREIECKEN (5. KLASSE)

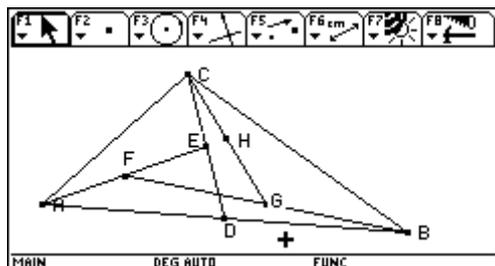
Seien A, B, C die Eckpunkte eines Dreiecks. Weitere Punkte werden wie folgt konstruiert: D ist der Mittelpunkt der Seite AB , E der Mittelpunkt der Strecke DC , F der Mittelpunkt der Strecke EA , G der Mittelpunkt der Strecke FB , H der Mittelpunkt der Strecke GC . Dann liegen A, E und H auf einer Geraden und es gilt: $AE : AH = 8 : 9$.

Für diese Aufgabe genügen bereits elementare Kenntnisse der Vektorrechnung. Der übliche Weg, den ich mit den Schülern einschlage, lässt sich in 4 Stufen einteilen.

Stufe 1: Geometrische Veranschaulichung (Zeichnung).

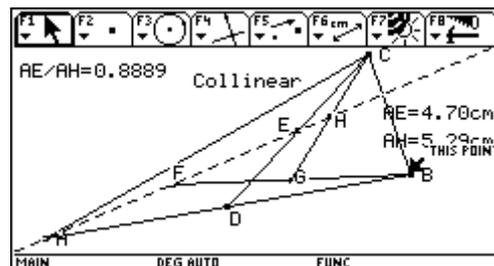
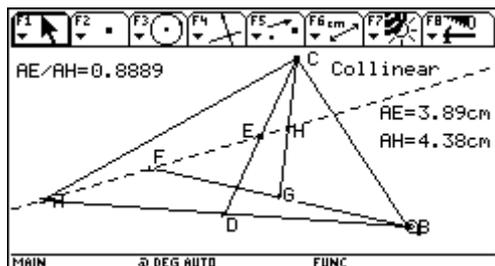


Diese Veranschaulichung lässt sich mit Hilfe von Dynamischer Geometrie (Cabri) auf dem TI-Rechner noch weiter führen.



Im Zugmodus lässt sich das Dreieck verändern. Cabri überprüft (allerdings nur probabilistisch) die Kollinearität der Punkte.

Das Streckenverhältnis bleibt erhalten.



Stufe 2: Gemeinsames Nachrechnen mit konkreten Zahlenwerten.

Siehe dazu Skript 1.

Anmerkungen:

- Die Schüler bevorzugen meistens die Formel $A + 1/2(B-A)$ anstatt $1/2(A+B)$.
- Die letzte Zeile liefert einen Vektor, in dem komponentenweise durchdividiert wurde. Sind beide Komponenten gleich, so beweist das die Kollinearität von A, E und H. Außerdem erhält man auf diese Weise das gesuchte Verhältnis.

Skript 1

$[\emptyset; \emptyset] \rightarrow pa$
 $[1; \emptyset] \rightarrow pb$
 $[\emptyset.3; \emptyset.7] \rightarrow pc$
 $pa + 1/2 * (pb - pa) \rightarrow pd$
 $pd + 1/2 * (pc - pd) \rightarrow pe$
 $pe + 1/2 * (pa - pe) \rightarrow pf$
 $pf + 1/2 * (pb - pf) \rightarrow pg$
 $pg + 1/2 * (pc - pg) \rightarrow ph$
 $(pe - pa) ./ (ph - pa)$

Stufe 3: Individuelles Ausprobieren mit anderen Zahlenwerten, z.B. Zufallszahlen. Jeder Schüler probiert es mit eigenen Zahlenwerten aus oder verwendet die rand()-Funktion um besonders „hässliche“ Zahlen zu erzeugen.

Stufe 4: Verallgemeinern auf allgemeine Zahlen, falls möglich. (Mehr dazu siehe später).

Damit ist der Satz allgemein bewiesen. Mathematiker würden nun so vorgehen, dass sie nach weitergehenden Verallgemeinerungen suchen (Schüler kommen von sich aus eher selten auf diese Idee).

Als mögliche Verallgemeinerung bietet sich an: Gilt der Satz in ähnlicher Form auch für ein anderes Teilverhältnis? Dies führt uns zu den Stufen 5 bis 7.

Zuerst lässt man das die Schüler wieder mit dem Cabri-Modell überprüfen, bzw. Vermutungen aufstellen.

Stufe 5:

Anstatt der Zahl $1/2$ wird eine andere Zahl eingesetzt, z.B. $1/3$. Auch hier ergibt sich eine Kollinearität der Punkte A, E und H. Das Streckenverhältnis ist $27 : 35$.

Stufe 6: Verallgemeinerung auf einen allgemeinen Teilungsfaktor x . Siehe dazu Skript 2. Wir haben Glück. Der TI-89 liefert uns in annehmbarer Zeit ein Ergebnis:

$$\frac{\overline{AE}}{\overline{AH}} = \frac{1}{-x^3 + 3x^2 - 3x + 2}$$

Anmerkungen:

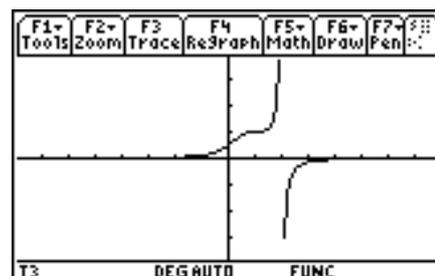
- Hat der TR Schwierigkeiten, könnte man eine Exaktheitsstufe heruntergehen und z.B. vorläufig wieder konkrete Zahlenwerte für A, B und C einsetzen.
- Die nahe liegende Schreibweise $[a1;a2]$ usw. ist leider nicht möglich, da $c1, c2$ usw. für den Data/Matrix-Editor reservierte Variablen sind.

Skript 2

```
[ax; ay] → pa
[bx; by] → pb
[cx; cy] → pc
pa+x*(pb-pa) → pd
pd+x*(pc-pd) → pe
pe+x*(pa-pe) → pf
pf+x*(pb-pf) → pg
pg+x*(pc-pg) → ph
(pe-pa) ./ (ph-pa)
```

Stufe 7: Funktionale Betrachtung des Teilverhältnisses.

Hier wird eine Querverbindung zur Funktionenlehre geschaffen. Wir fragen uns, für welches x das Problem überhaupt definiert bzw. lösbar ist. Der Graph der Funktion $y(x) = 1/(-x^3+3x^2-3x+2)$ hat deutlich sichtbar eine Polstelle bei $x = 2$. Man sollte sich nun anhand einer Skizze klarmachen, warum für $x = 2$ kein Verhältnis existiert (Antwort: Weil die Punkte A und H zusammenfallen).



Ebenso kläre man die Frage, wie eigentlich der Fall $x = 0$ aussieht (hier fallen A, E und H zusammen. Für $x = 0$ ist das Verhältnis nicht definiert, obwohl die Funktion dies nicht indiziert, allerdings existiert der Grenzwert für $x \rightarrow 0$; er ist $1/2$). Die funktionale Betrachtung lässt auch die Fälle $x < 0$ als möglich und sinnvoll erscheinen (was tatsächlich der Fall ist).

Auch weitere Verallgemeinerungen sind denkbar: z.B. ob nach einer weiteren „Umdrehung“ im gleichen Stil wieder ein zu AE kollinärer Punkt entsteht (das tut es), ob dies nach jeder „Umdrehung“ der Fall ist (anscheinend ja), ob sich der Satz auf Vierecke übertragen lässt (das tut er).

Nachbemerkung:

Nicht immer schafft der TI eine Berechnung ausschließlich mit Variablen. Hier können folgende Überlegungen getroffen werden.

- Verwendet man Zufallszahlen anstatt Variablen, so ist dies für den TI nicht schwieriger als mit „schönen“ Zahlen. Stimmt das zu Beweisende auch nun, ist dies zwar im strengen mathematischen Sinn noch nicht ausreichend, aber immerhin ein starkes Indiz für die Richtigkeit des Satzes. Es gilt hier nur aufzupassen, dass durch Rundungsfehler minimal unterschiedliche Ergebnisse auftreten können, die vom TI als nicht gleich interpretiert werden.

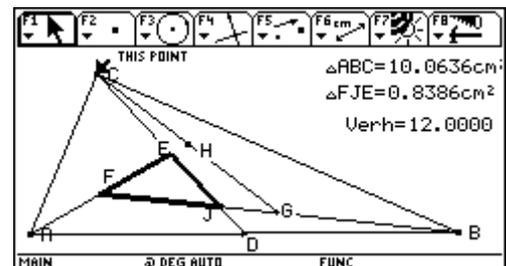
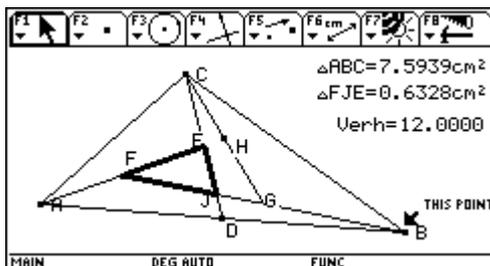
- Eine Vereinfachung ohne Beschränkung der Allgemeinheit ist in vielen Fällen möglich, indem man einen Punkt als (0 | 0) und einen zweiten Punkt als (1 | 0) setzt. Dies entspricht der freien Wahl des Nullpunktes und der Längeneinheit sowie der Invarianz gegenüber Drehung des Koordinatensystems.

Weitere Anregungen, die für die 5. Klasse geeignet sind:

- Übertrage den oben behandelten Satz auf 4-, 5-, usw. Ecke.
- Beweise, dass der Schwerpunkt die Schwerlinien im Verhältnis 1:2 teilt.
- Außer den bekannten merkwürdigen Punkten H, S, U und I sind hunderte weitere Dreieckszentren bekannt. Beispiele und Konstruktionsanleitungen finden sich unter <http://www2.evansville.edu/ck6/tcenters/class/>.
- Können die Schüler weitere Invarianzen finden? Mit Hilfe der dynamischen Geometrie ist das Experimentieren sehr einfach. So könnte man z.B. ein, von den Teilungslinien gebildetes Dreieck herausgreifen und fragen, ob das Verhältnis der Flächeninhalte

großes Dreieck : Teildreieck auch konstant bleibt?

Nach der, durch die Konstruktion gestützten Vermutung folgt der Beweis und die Ableitung einer allgemeinen Formel (Begründungen durch die Teilverhältnistreue der Dreieckstransformationen werden sich dabei sicher ergeben!).



- Gilt dieses konstante Verhältnis für alle möglichen Teildreiecke?
- Wie sieht das Dreieck nächster Stufe aus, wenn die Konstruktion fortgesetzt wird? Gibt es ein letztes Dreieck?

1.2 SPIEGELUNGSEIGENSCHAFTEN VON KEGELSCHNITTEN

(7. KLASSE)

Wer Karl May („Winnetous Erben“) gelesen hat, kennt sicherlich folgende Eigenschaft einer Ellipse:

Ein Strahl, der von einem Brennpunkt einer Ellipse ausgeht, wird von der Ellipse so reflektiert, dass er nach der Reflexion durch den anderen Brennpunkt verläuft.

Nach einer Zeichnung (Stufe 1) stellt sich die Frage, wie die Eigenschaft, auch im Falle konkreter Zahlen, nachgewiesen werden kann. Ein Algorithmus könnte so lauten:

- Setze a und e fest. Die kleine Halbachse b ergibt sich dann von selbst.
- Setze einen Punkt T fest mit $-a < x_T < a$; berechne das dazugehörige $y_T > 0$.
- Berechne die Ableitung k der Ellipsenfunktion bei $x = x_T$.
- Bilde den Tangentialvektor t auf die Kurve in T .
- Weise nach, dass $\angle(F_1T, t) = \angle(TF_2, t)$

Der Algorithmus für Stufe 2 und Stufe 3 ist in Skript 3 abgebildet.

Skript 3

```

delvar a,e,tx
5→a
3→e
3.5→tx
√(a^2-e^2)→b
b/a*√(a^2-x^2)→e11
[-e;0]→pf1
[e;0]→pf2
e11|x=tx→ty
[tx;ty]→pt
d(e11,x)|x=tx→k
[1;k]→vt
cos⁻¹(dotp(unitv(vt),unitv(pt-pf1)))→α
cos⁻¹(dotp(unitv(vt),unitv(pf2-pt)))→β
α-β

```

Anmerkungen:

- Der Befehl `delvar` soll uns später ermöglichen mit Variablen zu rechnen, da damit jede Zahlenbelegung gelöscht wird.
- Es würde genügen, die $\cos^{-1}()$ -Funktion wegzulassen, da bei Gleichheit der Argumente auch das Ergebnis gleich wäre.
- Wenn das Ergebnis der letzten Zeile 0 ist, ist die Behauptung nachgewiesen.

In Stufe 4 bekommen wir Probleme. Der TI ist so genau, dass er jede Menge `conj()`-Funktionen in die Lösung einbaut, die eine weitere Vereinfachung des Terms unterbindet. Hier hilft händisches Löschen des Wortes „conj“ oder die kleine selbstgeschriebene Funktion `xreal()`; siehe Anhang 1.

Ehrlicher ist es vielleicht keine Tricks zu verwenden, sondern zu akzeptieren, dass wir auf einem bestimmten Exaktheitsniveau stehen geblieben sind. Wenn das bewusst gemacht werden kann, haben die Schüler viel über das Beweisen dazugelernt.

Erweiterung:

Ähnliche Sätze gelten für die Hyperbel und die Parabel:

Ein Strahl, der von einem Brennpunkt einer Hyperbel ausgeht, wird von der Hyperbel so reflektiert, dass der zweite Brennpunkt auf der Verlängerung des reflektierten Strahls liegt.

Ein Strahl, der vom Brennpunkt einer Parabel ausgeht, wird von der Parabel so reflektiert, dass er nach der Reflexion parallel zur Achse der Parabel verläuft. (Parabolspiegel!)

Die Schüler können dazu angehalten werden, zeichnerisch (oder mit Hilfe von CABRI) auf diese Analogschlüsse zu kommen. Durch kleine Änderungen im Skript können auch diese Eigenschaften nachgewiesen werden.

Weitere Anregungen:

- Eine Ellipse und eine Hyperbel mit gleichen Brennpunkten schneiden einander immer im rechten Winkel.

Die Tangenten im Schnittpunkt S schneiden die y -Achse in den Punkten P und Q . Dann liegen die Punkte P, Q, S, F_1, F_2 auf einem Kreis.

- Von einem beliebigen Punkt P einer Hyperbel wird eine Gerade parallel zur näher gelegenen Asymptote as_1 sowie die Tangente t gelegt. Die Schnittpunkte dieser beiden Geraden mit der anderen Asymptote as_2 seien H und T . Dann halbiert H die Strecke OT . Sei außerdem U der Schnittpunkt von t und as_1 . Dann halbiert P die Strecke TU .
- Eine Parabel wird mit einer beliebigen Geraden durch den Brennpunkt geschnitten. Die Tangenten an die Schnittpunkte stehen dann normal aufeinander.
- Durch eine Parabel wird eine Sehne S_1S_2 gelegt. Parallel zu dieser Sehne wird eine Tangente t gelegt, die die Parabel im Punkt P berührt. Die Tangenten t_1 (in S_1) und t_2 (in S_2) schneiden einander im Punkt T .
Dann gilt: Der Flächeninhalt S_1S_2P ist doppelt so groß wie der Flächeninhalt t_1t_2t .
- Satz von *PASCAL*: Sechs Punkte eines Kegelschnitts mögen ein konvexes Sechseck bilden. Die Schnittpunkte der Verlängerungen jeweils zweier gegenüberliegender Seiten liegen auf einer Geraden, der *Pascalschen Geraden*. Welche Ausnahmen gibt es?
- Der dazu „duale“ Satz von *BRIANCHON*: Sechs Tangenten eines Kegelschnitts mögen ein konvexes Sechseck bilden. Die Verbindungsgeraden zweier gegenüberliegender Ecken treffen sich in einem Punkt, dem *Brianchon'schen Punkt*. Welche Ausnahmen gibt es?

Problem 4: Wie lässt sich feststellen, ob eine Begrenzungsfläche bestrahlt wird oder im Schatten liegt?

Man bildet den nach außen gerichteten Normalvektor n (3-Finger-Regel) und berechnet den Winkel zwischen n und dem Lichtvektor ℓ . Liegt der Winkel zwischen -90° und 90° , so liegt die Fläche im Schatten, andernfalls im Licht.

Es sei noch erwähnt, dass sich nach meiner Erfahrung alle Schnittprobleme, die sich aus diesen Fragestellungen ergeben, auf den Schnitt zwischen Ebene und Gerade zurückführen lassen. Ein Kernstück besteht daher aus folgender Sequenz (Skript 4):

Gerade und Ebene werden definiert.

Gerade und Ebene werden gleichgesetzt. Es entsteht ein Vektor aus 3 Gleichungen; im Unterschied zu *DERIVE* muss man mit dem *TI* die drei Gleichungen einzeln angeben. `solve(g1sys, {t, u, v})` würde nicht funktionieren.

Skript 4

```
pp+t*v1→g1
pa+u*(pb-pa)+v*(pc-pa)→e1
g1=e1→g1sys
solve(g1sys[1,1] and g1sys[2,1]
      and g1sys[3,1], {t, u, v})→ls
g1||ls→ps
```

Anmerkung: Zur Vereinfachung habe ich eine Funktion `vsolve()` programmiert, welche die Zeilen 3 und 4 automatisiert. Ein Listing der Funktion findet sich im Anhang 2.

Anhand des folgenden, kommentierten Skripts 5 lässt sich der Lösungsweg für die eingangs gestellte Fragestellung nachvollziehen.

Skript 5

```
[0.;0.;0.]→pp:[0.;0.;5.]→pq
[2.;2.;-1.]→v1
[3.;4.;0.]→pa:[6.;4.;0.]→pb
[3.;4.;2.]→pc:[3.;7.;0.]→pd
[6.;7.;0.]→pe:[3.;7.;2.]→pf

pq+t*v1→g1
[u;v;0]→e1
vsolve(g1=e1, {t, u, v})→lq1
g1||lq1→pq1

pa+t*(pb-pa)→g2
pb+t*(pc-pb)→g3
pe+t*(pf-pe)→g4
pp+u*(pq-pp)+v*v1→e2

vsolve(g2=e2, {t, u, v})→lg
g2||lg→pg

vsolve(g3=e2, {t, u, v})→lh
g3||lh→ph

vsolve(g4=e2, {t, u, v})→li
g4||li→pi

pb+t*(pe-pb)→g5
vsolve(g5=e1, {t, u, v})→lj
g5||lj→pj
```

Die Punkte werden definiert. Dies ermöglicht eine unkomplizierte nachträgliche Änderung. Zur Vermeidung komplizierter Brüche wird ein Komma eingegeben.

Das Bild des oberen Stab-Endes auf die x-y-Ebene (E_1) wird berechnet. (Noch wissen wir nicht, ob dieser Punkt möglicherweise im Schatten des Prismas liegt.) $Q' = (10|10|0)$

Zur Analyse, wie der Schatten über die Kanten verläuft, werden die 3 Kanten AB, BC und EF aufgestellt, ebenso der Schattenraum (E_2)

Der Schatten verläuft über die Kante AB im Punkt $G(4|4|0)$.

Der Schatten verläuft über die Kante BC im Punkt $H(4|4|1.33)$

Der Wert $t = -0,33$ zeigt an, dass der Schatten *nicht* über die Kante EF verläuft. Punkt I hat also keine Relevanz.

Also läuft der Schatten wahrscheinlich über BE. Tatsächlich, und zwar im Punkt $J(6|6|0)$.

Anzustreben ist m.E. eine möglichst leicht verallgemeinerungsfähige Vorgangsweise, in der zu Beginn die Zahlenwerte definiert werden und danach nur mehr algorithmisch gearbeitet wird.

2.1 DURCHDRINGUNG EINER PYRAMIDE UND EINES QUADERS

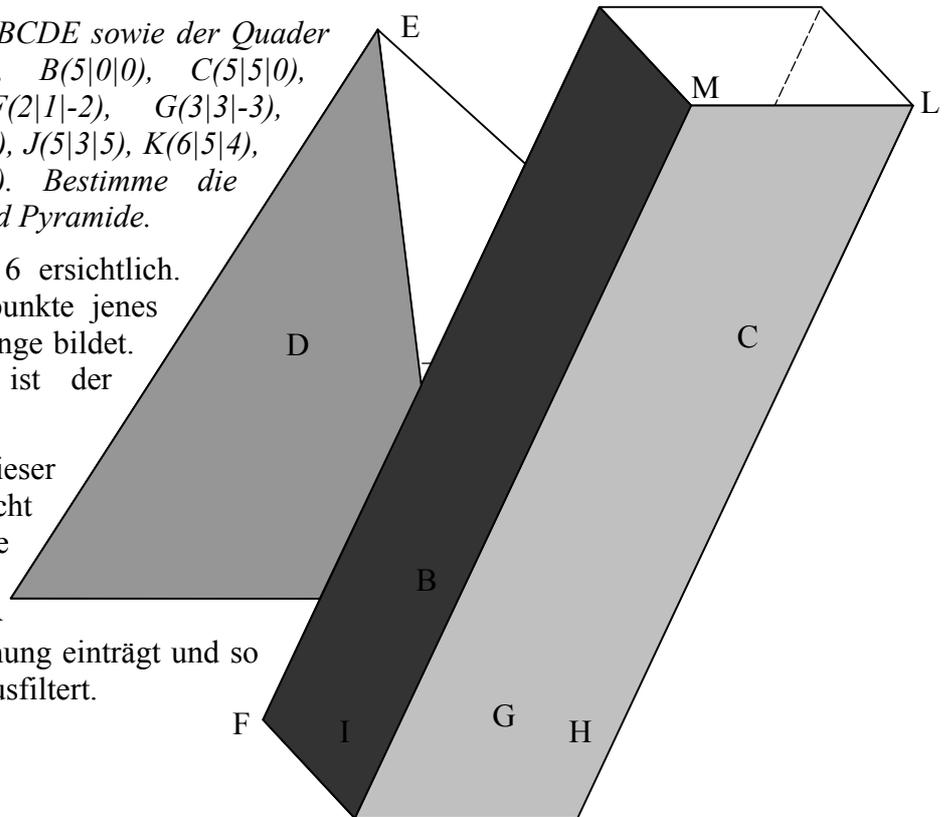
K

(6. KLASSE)

Gegeben ist die Pyramide $ABCDE$ sowie der Quader $FGHIJKLM$ mit $A(0|0|0)$, $B(5|0|0)$, $C(5|5|0)$, $D(0|5|0)$, $E(2,5|2,5|5)$; $F(2|1|-2)$, $G(3|3|-3)$, $H(5|1,75|-3,5)$, $I(4|-0,25|-2,5)$, $J(5|3|5)$, $K(6|5|4)$, $L(8|3,75|3,5)$, $M(7|1,75|4,5)$. Bestimme die Schnittmenge von Quader und Pyramide.

Die Lösung ist aus Skript 6 ersichtlich. Berechnet werden die Eckpunkte jenes Polyeders, der die Schnittmenge bildet. Auf der nächsten Seite ist der Schnittkörper visualisiert.

Das Schwierige bei dieser Aufgabe ist, dass man nicht einfach blindlings jede Fläche mit jeder Kante schneidet, sondern jedes Zwischen-A Ergebnis gleich in die Zeichnung einträgt und so mögliche Schnittpunkte herausfiltert.

Skript 6

```

[0;0;0]→pa:[5;0;0]→pb:[5;5;0]→pc
[0;5;0]→pd:[2.5;2.5;5]→pe
[2;1;-2]→pf:[3;3;-3]→pg
[5;1.75;-3.5]→ph:[4;-.25;-2.5]→pi
[5;3;5]→pj:[6;5;4]→pk
[8;3.75;3.5]→pl:[1;1.75;4.5]→pm

pg+t*(pk-pg)→g1
pa+u*(pb-pa)+v*(pd-pa)→e1
vsolve(g1=e1,{t,u,v})→lln:g1|lln→pn

pb+u*(pc-pb)+v*(pe-pb)→e2
vsolve(g1=e2,{t,u,v})→lo:g1|lo→po

pb+t*(pc-pb)→g2
pg+u*(ph-pg)+v*(pk-pg)→e3
vsolve(g2=e3,{t,u,v})→lp:g2|lp→pp

pf+t*(pj-pf)→g3
vsolve(g3=e1,{t,u,v})→lq:g3|lq→pq

vsolve(g3=e2,{t,u,v})→lr:g3|lr→pr

pa+t*(pb-pa)→g4
pf+u*(pj-pf)+v*(pi-pf)→e4
vsolve(g4=e4,{t,u,v})→ls:g4|ls→ps

vsolve(g2=e4,{t,u,v})→ls:g2|ls→ps

```

Definition der Eckpunkte

Schnitt der Kante GK mit der Pyramidengrundfläche ABCD; führt auf Punkt $N(4,3|3,9|0)$. Die Variable ln kann nicht verwendet werden (reserviertes Wort!).

Schnitt der Kante GK mit der Seitenfläche BCE; führt auf Punkt $O(4,6|4,1|0,8)$.

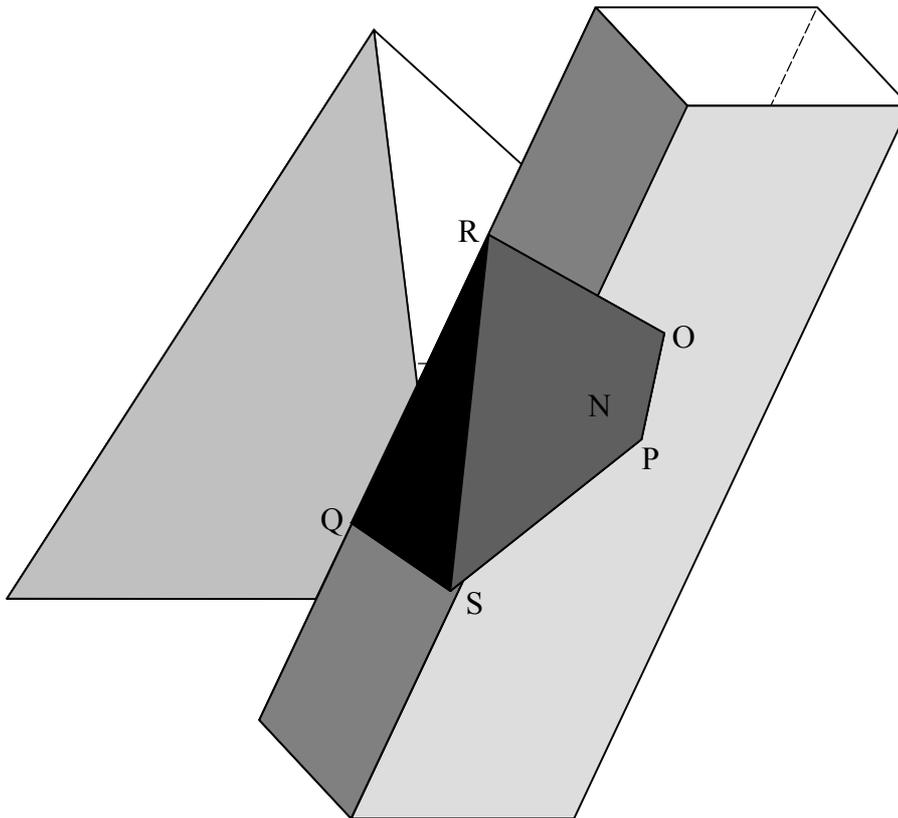
Schnitt der Kante BC mit der Quaderfläche GHLK; führt auf Punkt $P(5|3,5|0)$.

Schnitt der Kante FJ mit der Pyramidengrundfläche ABCD; führt auf Punkt $Q(2,9|1,6|0)$.

Schnitt der Kante FJ mit der Seitenfläche BCE; führt auf Punkt $R(3,8|2,2|2,3)$.

Schnitt der Kante AB mit der Quaderfläche FIMJ; kein Schnittpunkt ($t > 1$). Lösung wird verworfen.

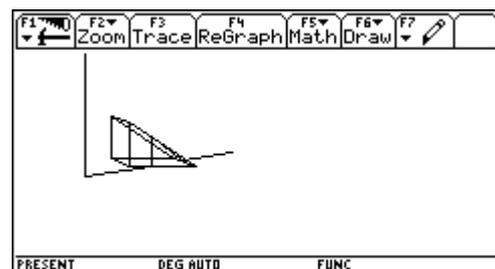
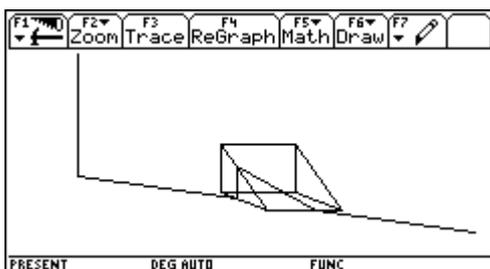
Stattdessen wird die Kante BC mit FIMJ geschnitten; führt auf Punkt $S(5|0,5|0)$.



Ergänzung:

Mit Hilfe eines Programmpakets, das den Schülern zur Verfügung gestellt wird, lassen sich auch auf dem TI schöne und doch eindrucksvolle graphische Darstellungen in Grund-, Auf- oder Schrägriss, in axonometrischer oder perspektivischer Darstellung erzielen.

Sehen Sie hier zwei Versuche für das Prisma + Stab + Schatten.



Das zugehörige Papier kann von der ACDCA/T³ –Homepage heruntergeladen werden.

<http://www.acdca.ac.at/material/t3/t33dobj.htm>

ANHANG 1: Die Funktion xreal()

```

xreal(ex)
Func
Local s,s1,pos,i,n
string(ex)→s:{ }→pos:∅→i
Loop
  inString(s,"conj",i+1)→i
  If i=∅ Then:Exit
  Else:augment(pos,{i})→pos:EndIf
EndLoop
dim(pos)→n
If n=∅ Then:s→s1
Else
  left(s,pos[1]-1)→s1
  For i,1,n1
    s1&mid(s,pos[i]+4,pos[i+1]-
pos[i]-4)→s1
  EndFor
  s1&mid(s,pos[n]+4)→s1
EndIf
Return expr(s1)
EndFunc

```

ANHANG 2: Die Funktion vsolve()

```

vsolve(glsys,vlist)
Func
Local d1,d2,n,i,zkette,modus,ex
rowDim(glsys)→d1
colDim(glsys)→d2
If d1>1 and d2=1 Then: 1→modus: d1→n
  ElseIf d1=1 and d2>1 Then: 2→modus:
d2→n
  Else: Return "Dimension error."
EndIf
"solve("&string(glsys[1,1])→zkette
For i,2,n
  zkette&" and "→zkette
  If modus=1 Then:
zkette&string(glsys[i,1])→zkette
  Else:
zkette&string(glsys[1,i])→zkette
  EndIf
EndFor
zkette&" "&string(vlist)&")"→zkette
expr(zkette)→ex
Return ex
EndFunc

```