THE DERIVE - NEWSLETTER #119

ISSN 1990-7079

THE BULLETIN OF THE DERIVE USER GROUP

+ CAS-TI

Contents:

1	Letter of the Editor
2	Editorial - Preview
3	Wolfgang Alvermann Tumbling Tour – Taumeltour im Vergnügungspark (E/D)
10	Sebastian Rauh Moving the Cube
12	A Bug with taylor?
13	Herbert Nieder Affine Abbildungen – Affine Mappings (D/E)
32	Central End Examination at a German Vocational School
37	Matthew Myers TI-89 Keystroke Functions (1)

September 2020

Some Recommended Links

Prof. Manfred Kronfellner's website provides a couple of interesting papers to Didactics of Mathematics and History of Mathematics (in German):

http://www.algebra.tuwien.ac.at/kronfellner/

http://www.algebra.tuwien.ac.at/kronfellner/Ausg.Kap.d.Fachdidaktik/index.htm

http://www.algebra.tuwien.ac.at/kronfellner/Geschichte%20der%20Mathematik-Uni-SS18/Geschichte%20der%20Mathematik-Uni-SS18.zip

http://www.algebra.tuwien.ac.at/kronfellner/Geschichte%20der%20Mathematik-Uni-SS18/index.htm

You can download three complete books as pdf-files:

https://www.whitman.edu/mathematics/multivariable/multivariable.pdf
(Single and Multivariable Calculus, 515 pages

https://www.whitman.edu/mathematics/higher_math_online/higher_math.pdf
(An Introduction to Higher Mathematics, 136 pages)

https://www.whitman.edu/mathematics/cgt_online/cgt.pdf
(An Introduction to Combinatorics and Graph Theory, 153 pages)

Link provided by Sebastian Rauh:

Hey, all together,

Great Video, have a look at this. Funny integration method!

https://www.youtube.com/watch?v=xiIsPEqyTqU&feature=youtu.be

Regards

Sebastian

What's your Opinion?

Wolfgang Alvermann saw a Quiz show in TV where the following question was posed:

Is a secant a straight line which intersects a curve in exact
a) one point,
b) two points or
c) three points?

Is it correct to pose this question and to expect a unique answer?

Dear DUG Members,

I welcome you to the 119th issue of our newsletter and hope that you all are healthy. This DNL has become rather extensive with 50 pages.

Please notice the links given on the information page. They connect you with the homepage of Prof. Manfred Kronfellner (TU Vienna) and focus on didactics and history of mathematics. Furthermore, there is another link (provided by Sebastian Rauh) to a video presenting an "interesting" integration method. Then we have a question on a quiz question posed in a TV-show (Thanks to Wolfgang Alvermann).

Herbert Nieder and Wolfgang Alvermann are the authors of two comprehensive contributions, which have been extended with additions for treatments with TI-Nspire and DERIVE. Many thanks for the intense and fruitful communication with you.

In our last DNL Wolfgang Alvermann presented his final exam from 1968. For comparison we have now a final exam of the same type of school from one of the last years. It's worth the comparison. The remaining part and solutions will be given in the next DNL.

After a short but worth seeing animation of a cube (Sebastian Rauh) follows a – at least for me – very exciting contribution from a – yes, you read right – US prison. Please read at first the handwritten letter, which was sent to me by Dr. Robert Haas, mentor of the author. Behind this contribution is a human fate. I am especially grateful to Dr. Haas, who cares so responsibly for MM. And, after long, I could like with Mathtools in the last DNL, be busy with my old love, the V200 and the TI-89.

I close with the – now very common – wish: "Stay Healthy!" and am looking forward to our next contacts.

Best regards as ever,

Josef

Liebe DUG-Mitglieder,

ich begrüße Euch alle herzlich zur 119. Ausgabe unseres Newsletters und hoffe, dass Ihr alle gesund seid. Dieser DNL ist mit 50 Seiten wieder recht umfangreich geworden.

Beachtet bitte die Links auf der Infoseite. Sie führen zur Homepage von Prof. Manfred Kronfellner (TU-Wien) mit den Schwerpunkten Didaktik und Geschichte der Mathematik. Außerdem gibt es eine Frage zu einer im TV gestellten Quizfrage (Dank an Wolfgang Alvermann für den Hinweis). Ein weiterer Link (Dank an Sebastian Rauh) führt zu einer „interessanten" Integrationsmethode.

Herbert Nieder und Wolfgang Alvermann sind die Autoren zweier umfangreicher Beiträge, die mit Anhängen zur Bearbeitung mit TI-Nspire, bzw. DERIVE erweitert wurden. Vielen Dank auch hier für die intensive und fruchtbare Korrespondenz mit Euch.

Im vorigen DNL hat uns Wolfgang Alvermann sein Abitur aus dem Jahr 1968 vorgestellt. Zum Vergleich gibt es hier Abitur aus dem gleichen Schultyp aus einem der letzten Jahre. Den restlichen Teil + Lösungen findet Ihr dann im nächsten DNL. Der Vergleich lohnt sich.

Nach einer kurzen, aber sehenswerten Animation eines Würfels von Sebastian Rauh folgt ein für mich sehr aufregender Beitrag aus einem – Ihr lest richtig – US-Strafgefängnis. Bitte lest zuerst den handschriftlichen Brief, der mir über Dr. Robert Haas, dem Betreuer des Autors, gesandt wurde. Hinter dem Beitrag steckt ein Menschenschicksal. Mein Dank gilt hier im ganz besonderen Dr. Haas, der sich so verantwortungsvoll seinem Schützling widmet. Und ich habe mich nach langer Zeit wieder – wie bei den Mathtools im letzten DNL – mit meiner alten Liebe, dem V200 und TI-89 beschäftigen können.

Ich schließe mit dem jetzt üblichen Wunsch: „Bleibt gesund!" und freue mich auf unsere nächsten Kontakte.

Beste Grüße,

Josef

The *DERIVE-NEWSLETTER* is the Bulletin of the *DERIVE* & CAS-*TI User Group*. It is published at least four times a year with a content of 40 pages minimum. The goals of the *DNL* are to enable the exchange of experiences made with *DERIVE, TI*-CAS and other CAS as well to create a group to discuss the possibilities of new methodical and didactical manners in teaching mathematics.

Editor: Mag. Josef Böhm
D´Lust 1, A-3042 Würmla, Austria
Phone: ++43-(0)660 31 36 365
e-mail: nojo.boehm@pgv.at

Contributions:
Please send all contributions to the Editor. Non-English speakers are encouraged to write their contributions in English to reinforce the international touch of the *DNL*. It must be said, though, that non-English articles will be warmly welcomed nonetheless. Your contributions will be edited but not assessed. By submitting articles, the author gives his consent for reprinting it in the *DNL*. The more contributions you will send, the more lively and richer in contents the *DERIVE* & CAS-*TI Newsletter* will be.

Next issue: December 2020

Preview: Contributions waiting to be published

Some simulations of Random Experiments, J. Böhm, AUT, Lorenz Kopp, GER
Wonderful World of Pedal Curves, J. Böhm, AUT
Simulating a Graphing Calculator in *DERIVE*, J. Böhm, AUT
Cubics, Quartics – Interesting features, T. Koller & J. Böhm, AUT
Logos of Companies as an Inspiration for Math Teaching
Exciting Surfaces in the FAZ
BooleanPlots.mth, P. Schofield, UK
Old traditional examples for a CAS – What´s new? J. Böhm, AUT
Mandelbrot and Newton with *DERIVE*, Roman Hašek, CZ
Tutorials for the NSpireCAS, G. Herweyers, BEL
Dirac Algebra, Clifford Algebra, Vector-Matrix-Extension, D. R. Lunsford, USA
Another Approach to Taylor Series, D. Oertel, GER
Statistics of Shuffling Cards, Charge in a Magnetic Field, H. Ludwig, GER
More Applications of TI-Innovator™ Hub and TI-Innovator™ Rover
Surfaces and their Duals, Cayley Symmetroid, J. Böhm, AUT
Investigations of Lottery Game Outcomes, W Pröpper, GER
A Collection of Special Problems, W. Alvermann, GER
DERIVE Bugs? D. Welz, GER
Tweening & Morphing with TI-NspireCX-II-T, J. Böhm. AUT
The Gap between Poor and Rich, J. Böhm, AUT
With a Turtle to Fractals, J. Wagner, GER
Final Exam, part 2
More functions from Matthew and from Bhuvanesh's Mathtools-library
QR-Code light, 153 is another Special Number, and others

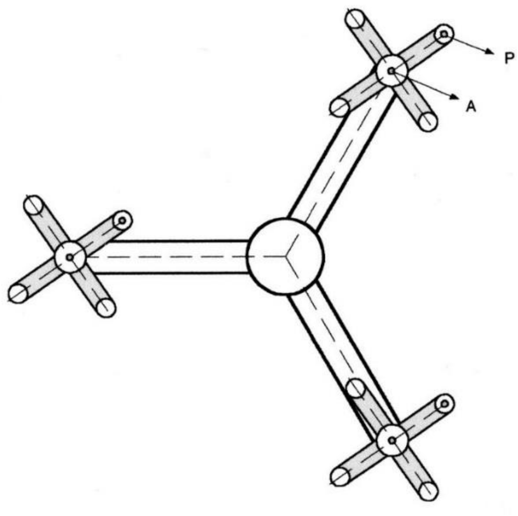
Impressum:
Medieninhaber: *DERIVE* User Group, A-3042 Würmla, D´Lust 1, AUSTRIA
Richtung: Fachzeitschrift
Herausgeber: Mag. Josef Böhm

Tumbling Tour in the Amusement Park

Wolfgang Alvermann, Germany

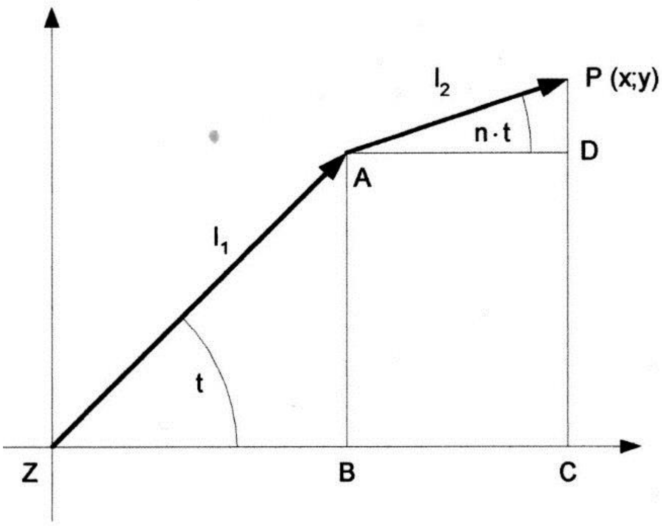
On fairs rides on a carousel belong to the popular entertainments offered.
The sketch shows the principle of the BREAK DANCER, consisting of four arms which move on a slanting disk. This motion shall be described in the following.

- To make it easy we will restrict the problem on the *x*-*y*-plane.
- It is sufficient to investigate only one arm of the ride – point *P*.
- Initially we will do without the concepts of speed *v* and angular velocity ω.



Assisted by the following graph the trajectory of point *P* shall be described for some cases:

1) Uniform motion of *A* and *P*
   a) Rotations in same direction
   b) Rotations in opposite direction
2) Counterclockwise motion of *A* and nonuniform motion of *P* around *A* – in same direction and opposite as well
3) Some considerations on speed and angular velocity



Designations:

Z: Position of the central vertical axis (hub)

A: Position of the vertical axis of the turnstile

P: Position of the riding person

t: Rotation angle

n: Can be interpreted as the ratio of the rotation velocities

Description of point *P*

The following relations allow representing the position of a riding person:

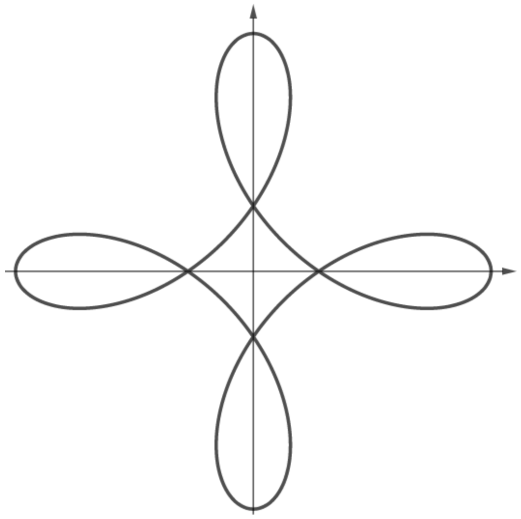
$$\left.\begin{array}{l}\overline{ZA}=\overline{ZB}+\overline{BA}=l_1\cdot\cos(t)+l_1\cdot\sin(t)\\ \overline{AP}=\overline{AD}+\overline{DP}=l_2\cdot\cos(n\cdot t)+l_2\cdot\sin(n\cdot t)\end{array}\right\}\rightarrow\qquad\begin{array}{l}x(t)=l_1\cdot\cos(t)+l_2\cdot\cos(n\cdot t)\\ y(t)=l_1\cdot\sin(t)+l_2\cdot\sin(n\cdot t)\end{array}$$

Hence, the motion of point *P* can be described as a curve given in parameter form.

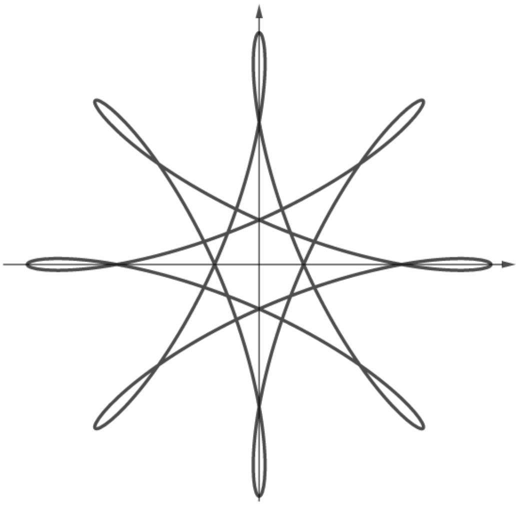
$$P(t)=\begin{cases}l_1\cdot\cos(t)+l_2\cdot\cos(n\cdot t)\\ l_1\cdot\sin(t)+l_2\cdot\sin(n\cdot t)\end{cases}\quad t=m\cdot\pi$$

Parameters for the graphs below are:

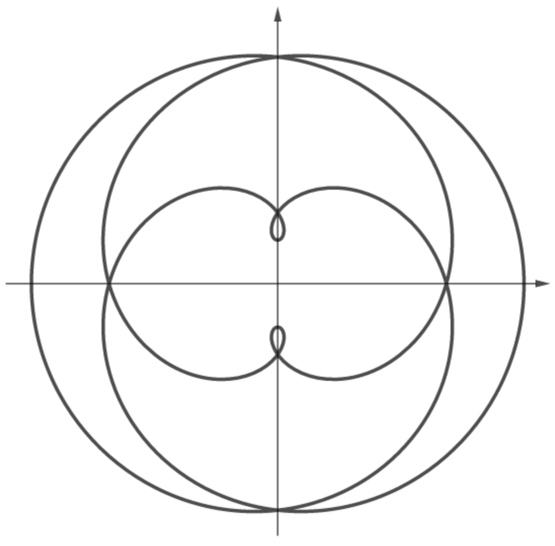
$l_1 = 5$ $l_2 = 3.5$ $n = \{-3, -5/3, 5/3, 4\}$



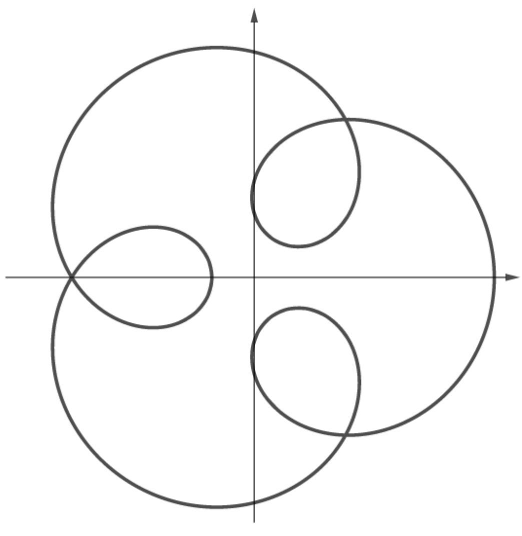
n = -3



n = -5/3



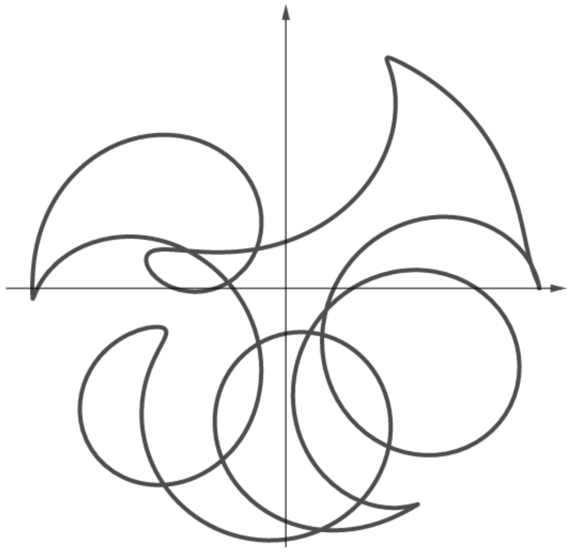
n = 5/3



n = 4

Nonuniform (irregular) motions result substituting a function $n(t)$ for n. We define $n(t) = -\cos(3t)$ as an example.

$$P(t) = \begin{cases} l_1 \cdot \cos(t) + l_2 \cdot \cos(-\cos(3t) \cdot t) \\ l_1 \cdot \sin(t) + l_2 \cdot \sin(-\cos(3t) \cdot t) \end{cases} \quad t = m \cdot \pi \qquad \text{here with } m = 2$$



This ride can really make you feel sick!

Numerical calculations for uniform motions

Realistic rotation numbers (slightly adapted):

$$n_{Hub} = 12 \frac{1}{\min} = \frac{1}{5} \frac{1}{s} \qquad n_{Cross} = 30 \frac{1}{\min} = \frac{1}{2} \frac{1}{s}$$

Angular velocity ω indicates, how fast the rotation angle t changes with a fixed body. This results for uniform motions in:

$$\omega_h = 2 \cdot \pi \cdot n_{Hub} \approx 75.4 \frac{1}{\min} \approx 1.25 \frac{1}{s} \qquad \text{which is approx } 72°$$

$$\omega_C = 2 \cdot \pi \cdot n_{Cross} \approx 188.5 \frac{1}{\min} \approx 3.14159 \frac{1}{s} \qquad \text{which is } 180°$$

This leads to the speeds of A and P:

$$v_A = \omega_h \cdot l_1 = 6.25 \frac{m}{s} = 22.5 \frac{km}{h}$$

Relating to a fixed point A one would receive for P:

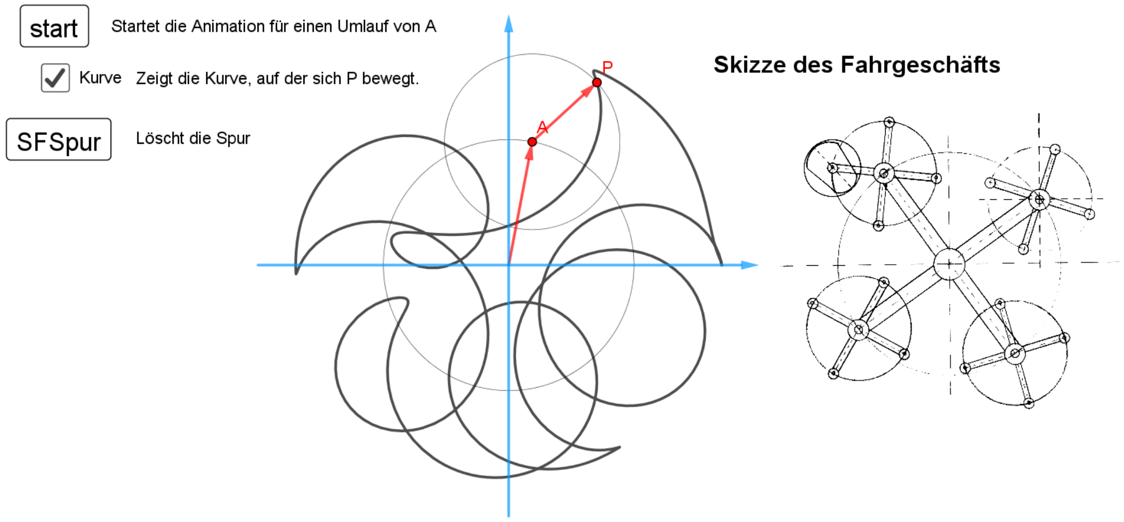
$$v_P = \omega_{Cross} \cdot l_2 \approx 11.7 \frac{m}{s} \approx 39.6 \frac{km}{h}$$

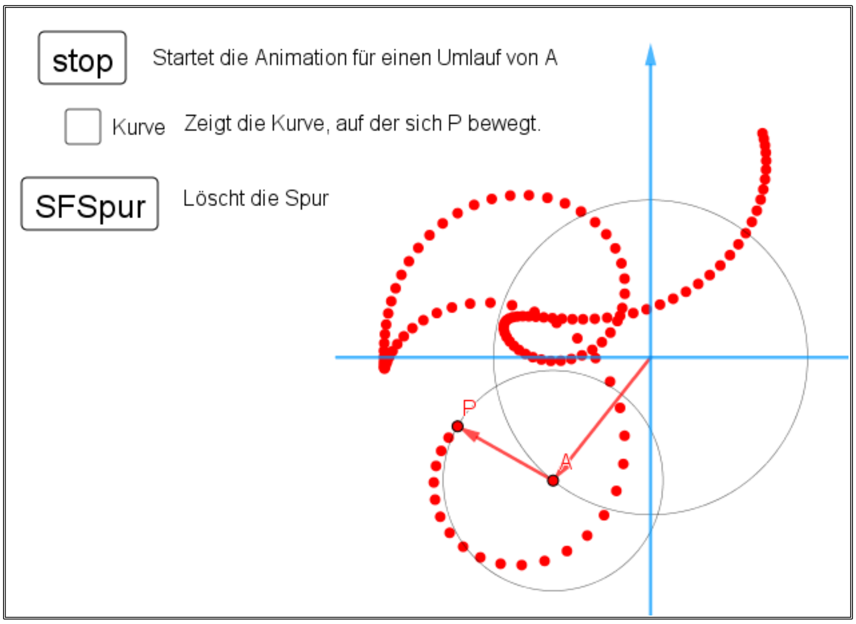
These velocities superimpose each other like the orbit of earth around sun with $v_E \approx 30\, {}^{km}\!/_{s}$ and the simultaneous rotation of earth around its axis with $v_R \approx 0.464\, {}^{km}\!/_{s}$. But we don't notice anything of this.

Wolfgangs deutsche Originalfassung ist unter den Dateien in MTH119.zip zu finden, ebenso wie die GeoGebra-Datei, die er zur besseren Illustration erzeugt hat.

Taumeltour im Vergnügungspark

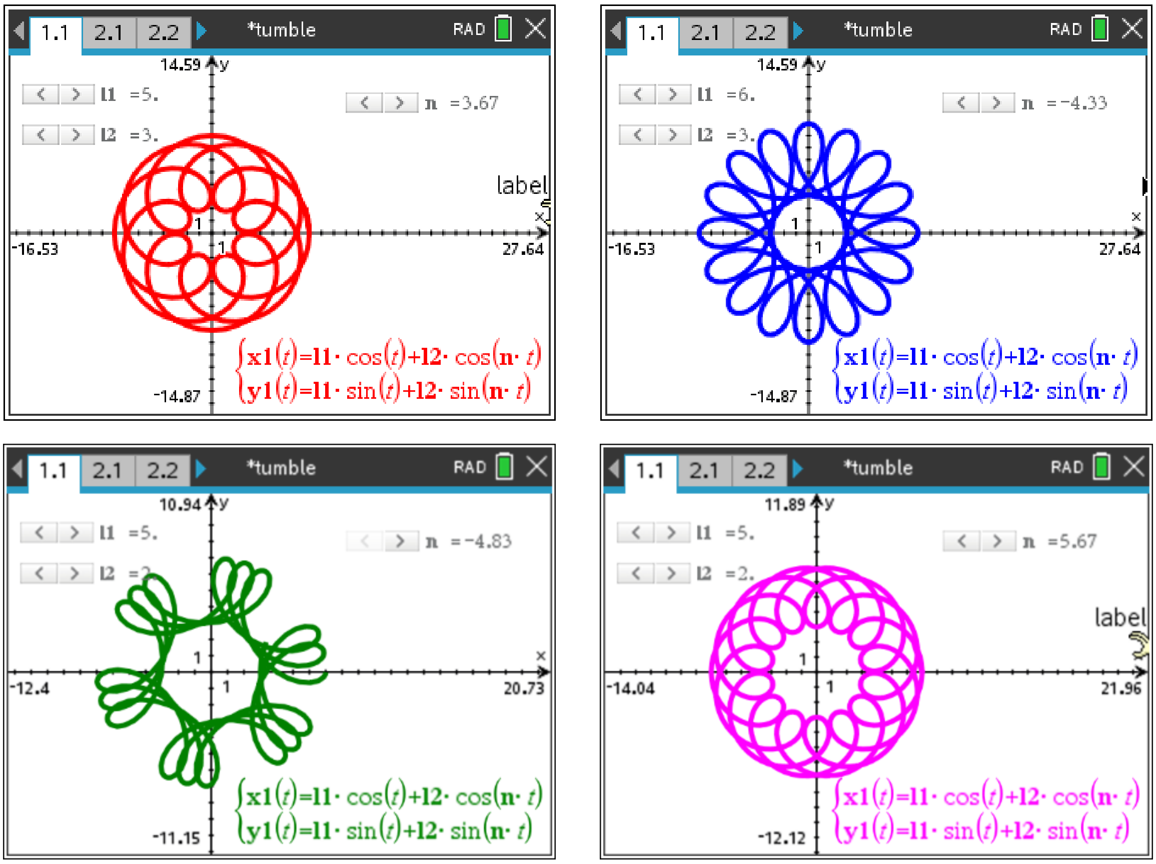
Eine vereinfachte Animation des Fahrgeschäfts “BREAK DANCE”





Wolfgang‘s original contribution in German is among the files contained in MTH119.zip together with his GeoGebra file. Using the buttons one can start and stop the animation or hide and show the trace. The middle tick mark presents or hides the curve.

In agreement with Wolfgang I add the TI-Nspire treatment, beginning with some variations of the pretty parameter curves from page 4. I installed sliders for the radii of the circles and for the coefficient *n*.



The form of the *BREAK DANCE* depends on the function *n*(*t*). For plotting I address them as f1(x), f2(x), …

$$f1(x):=k\cdot\sum_{i=0}^{4}\left(\frac{\cos((2\cdot i+1)\cdot k\cdot x)}{(2\cdot i+1)^2}\right):f2(x):=k\cdot\sum_{i=0}^{4}\left(\frac{\sin((2\cdot i-1)\cdot k\cdot x)}{2\cdot i-1}\right) \quad \textit{Done}$$

$$x1(t):=l1\cdot\cos(t)+l2\cdot\cos(f1(t)\cdot t):y1(t):=l1\cdot\sin(t)+l2\cdot\sin(f1(t)\cdot t) \quad \textit{Done}$$

$$x2(t):=l1\cdot\cos(t)+l2\cdot\cos(f2(t)\cdot t):y2(t):=l1\cdot\sin(t)+l2\cdot\sin(f2(t)\cdot t) \quad \textit{Done}$$

$$f3(x):=2\cdot\left|\sin\left(\frac{k\cdot x}{\pi}\right)\right| \quad \textit{Done}$$

$$x3(t):=l1\cdot\cos(t)+l2\cdot\cos(f3(t)\cdot t):y3(t):=l1\cdot\sin(t)+l2\cdot\sin(f3(t)\cdot t) \quad \textit{Done}$$

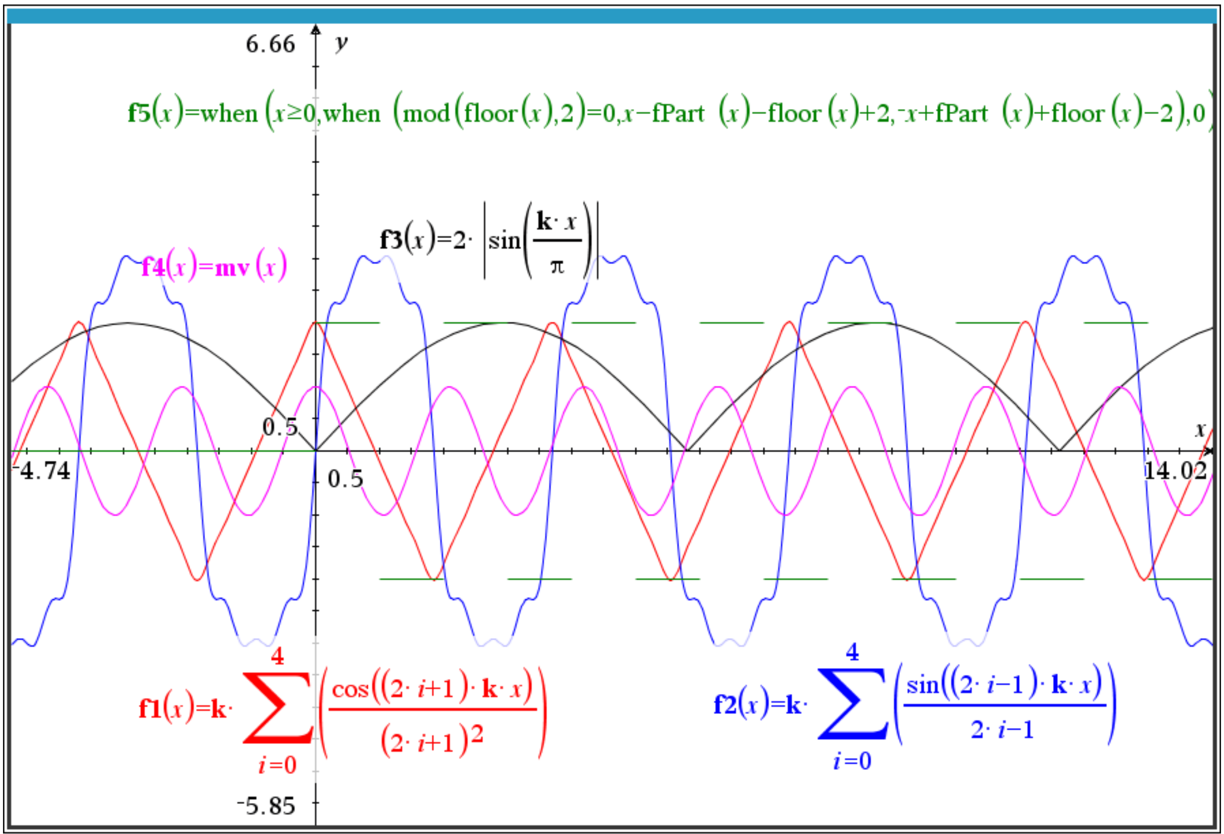
$$mv(x):=\cos(3\cdot x) \quad \textit{Done}$$

$$x4(t):=l1\cdot\cos(t)+l2\cdot\cos(mv(t)\cdot t):y4(t):=l1\cdot\sin(t)+l2\cdot\sin(mv(t)\cdot t) \quad \textit{Done}$$

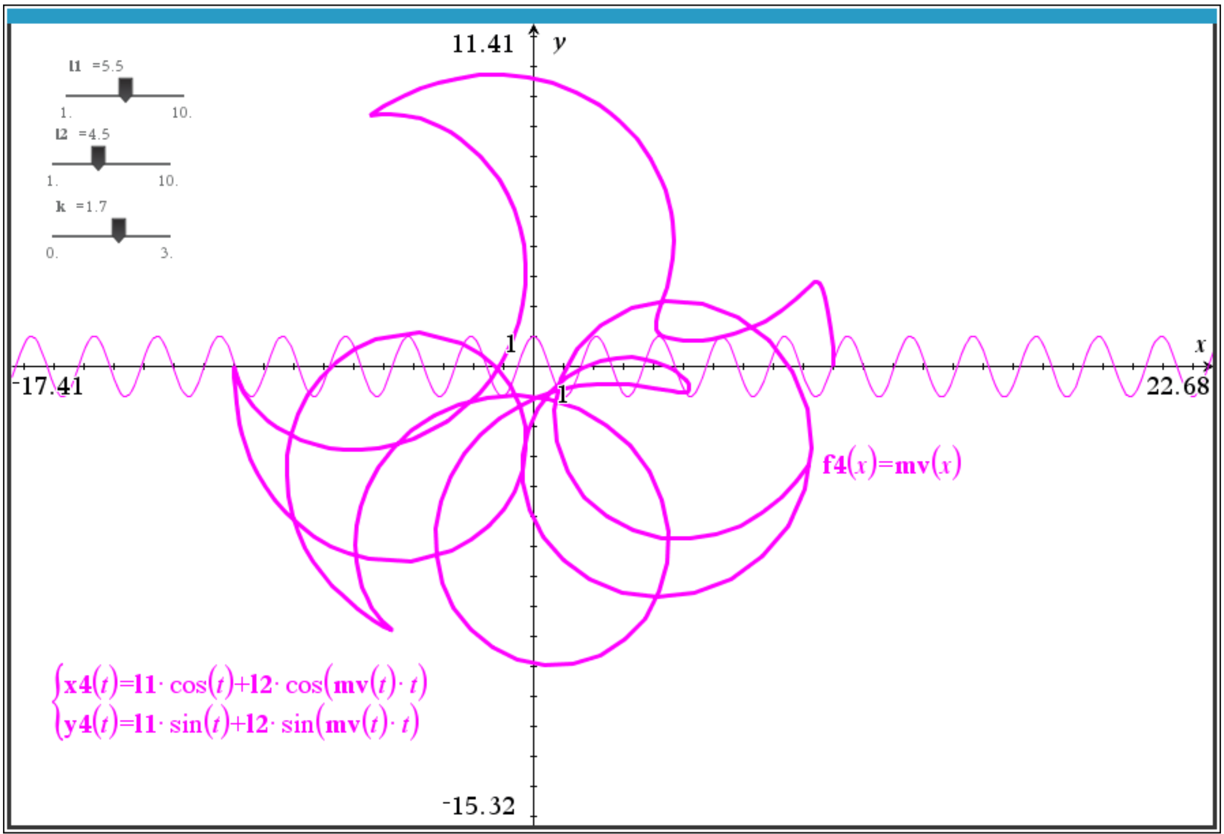
$$f5(x):=\text{when}(x\geq 0,\text{when}(\text{mod}(\text{floor}(x),2)=0,x-\text{fPart}(x)-\text{floor}(x)+2,{}^-x+\text{fPart}(x)+\text{floor}(x)-2),\blacktriangleright \quad \textit{Done}$$

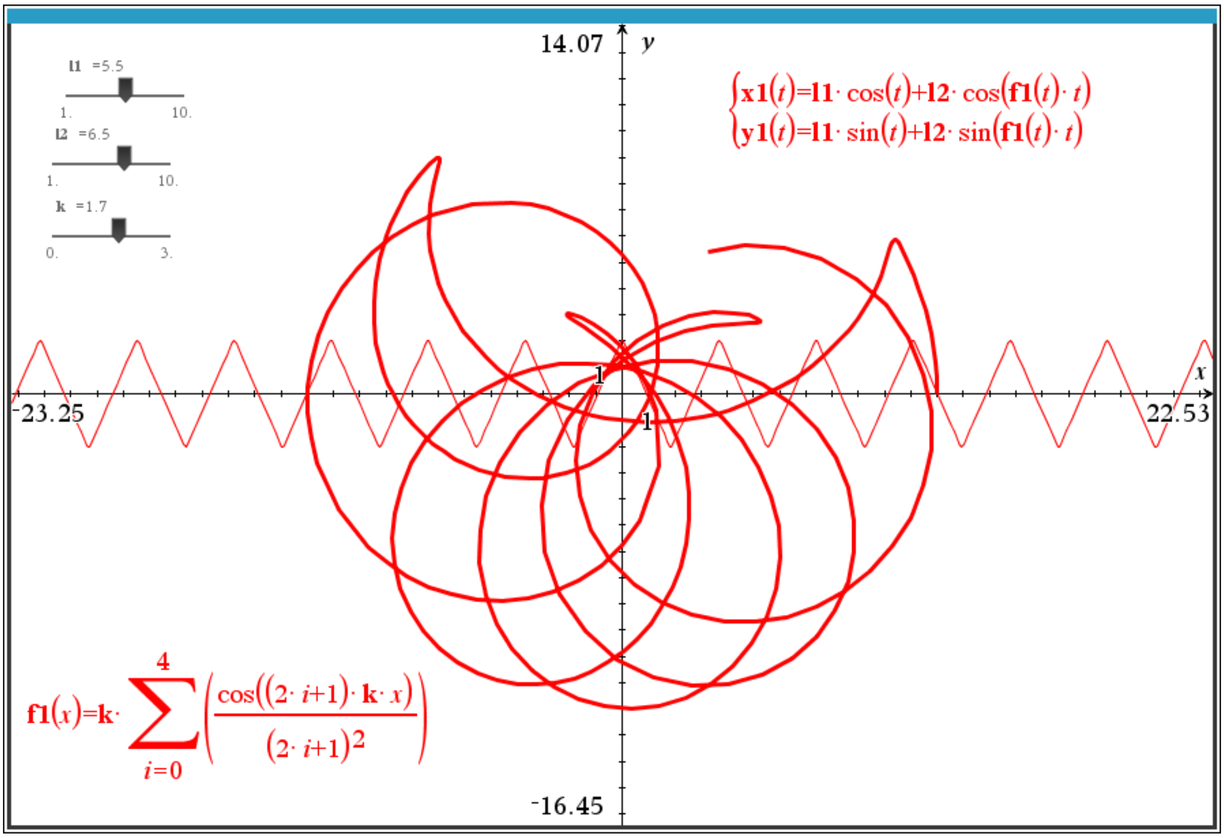
$$x5(t):=l1\cdot\cos(t)+l2\cdot\cos(f5(t)\cdot t):y5(t):=l1\cdot\sin(t)+l2\cdot\sin(f5(t)\cdot t) \quad \textit{Done}$$

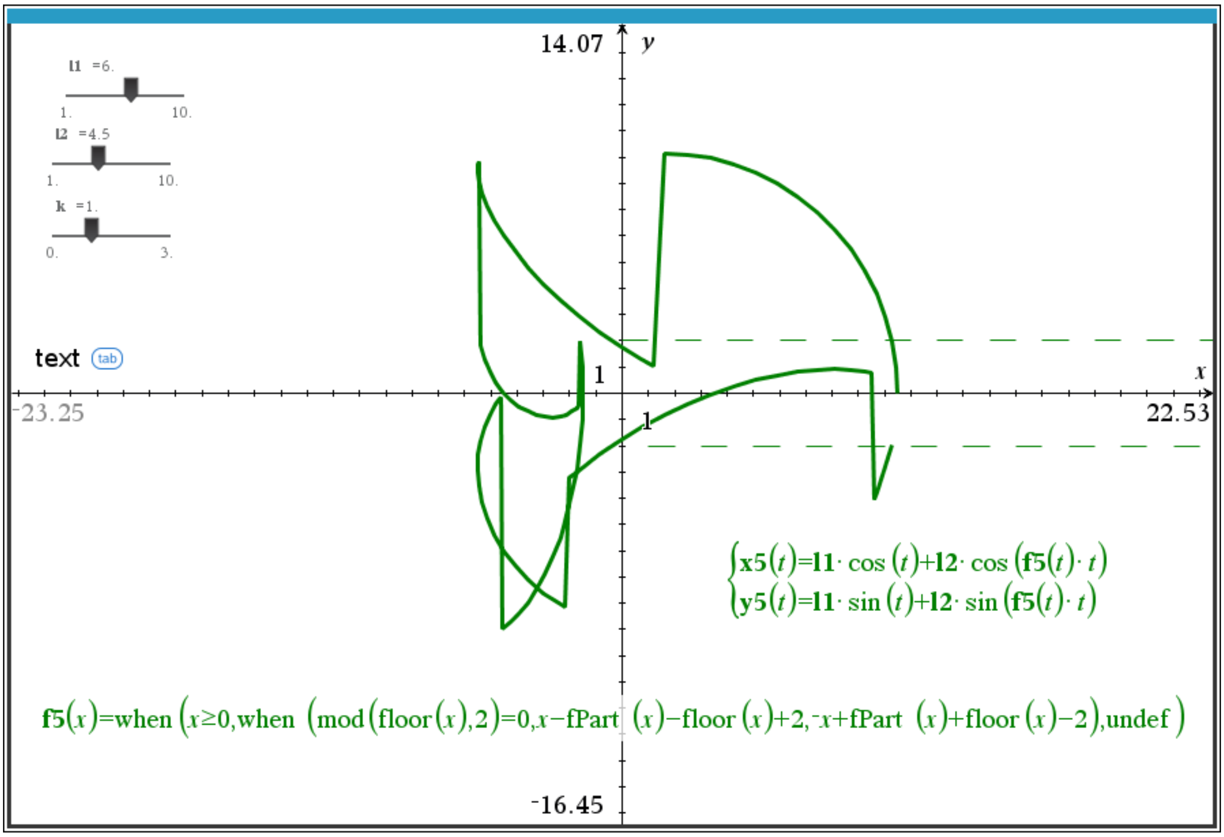
This screen shows the various functions used as $n(t)$ in Wolgang's tumble tour.



And here are some possible rides on the carousel – again with sliders, of course:





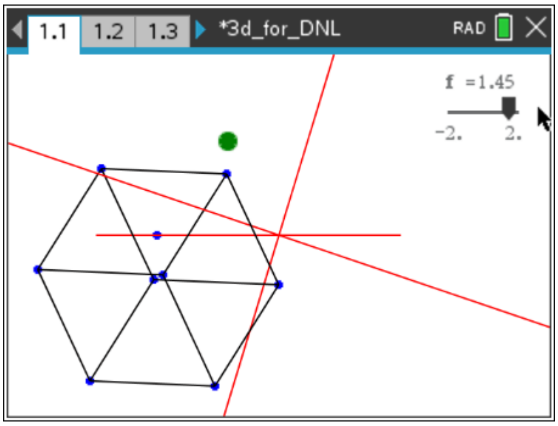


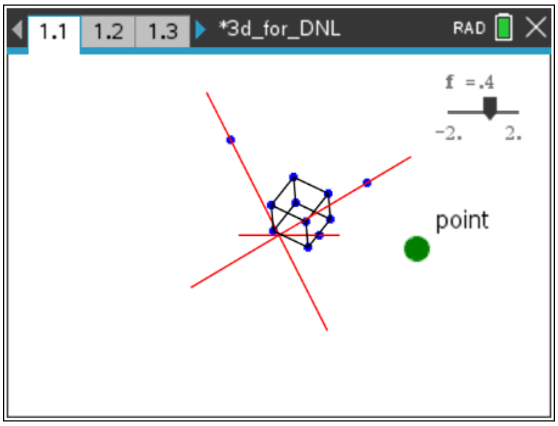
I believe that it should be no problem to undertake the rides on the DERIVE screen, too.

Hello Josef,

attached a file showing a 3d-cube in 2d-Graphs App. Have a look. It took me long time considering. The thick green point can be moved in order to change the perspective. Slider for *f* allows zooming.

Regards, Sebastian Rauh






```
Define f2x(nr,von,zu)=
Func
:nr:=string(nr)
:von:=string(von)::zu:=string(zu)
: Return "x"&nr&"(t):=xx["&von&"]+t*(xx["&zu&"]-xx["&von&"])"
:EndFunc

Define f2y(nr,von,zu)=
Func
:nr:=string(nr)
:von:=string(von)::zu:=string(zu)
:Return "y"&nr&"(t):=yy["&von&"]+t*(yy["&zu&"]-yy["&von&"])"
:EndFunc

Define project(p)=
Prgm
:Local i
:i:=5
:For i,4,6
:expr(f2x(i,i,i+1)):expr(f2y(i,i,i+1))
:EndFor
:expr(f2x(7,4,7)):expr(f2y(7,4,7))
:For i,8,11
:expr(f2x(i,i,i+1)):expr(f2y(i,i,i+1))
:EndFor
:expr(f2x(12,8,11)):expr(f2y(12,8,11))
:For i,13,16
:expr(f2x(i,i-9,i-5)):expr(f2y(i,i-9,i-5))
:EndFor
:EndPrgm
```

Projektionsmatrizen Projection matrices

Rotationsmatrizen Rotation matrices

Planare Projektion Planar projection

$$\mathbf{rx}(\theta):=\begin{bmatrix}1&0&0&0\\0&\cos(\theta)&-\sin(\theta)&0\\0&\sin(\theta)&\cos(\theta)&0\\0&0&0&1\end{bmatrix} \blacktriangleright \textit{Done}$$

Projektion auf z=0: $\mathbf{pp}:=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&\frac{1}{d}&1\end{bmatrix} \blacktriangleright \begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&\frac{1}{d}&1\end{bmatrix}$

$$\mathbf{ry}(\theta):=\begin{bmatrix}\cos(\theta)&0&\sin(\theta)&0\\0&1&0&0\\-\sin(\theta)&0&\cos(\theta)&0\\0&0&0&1\end{bmatrix} \blacktriangleright \textit{Done}$$

Perspektivische Projektion Perspective projection

$$\mathbf{rz}(\theta):=\begin{bmatrix}\cos(\theta)&-\sin(\theta)&0&0\\\sin(\theta)&\cos(\theta)&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix} \blacktriangleright \textit{Done}$$

$$\mathbf{pp2}:=\begin{bmatrix}\mathbf{f}&0&0&0\\0&\mathbf{f}&0&0\\0&0&\mathbf{f}&0\\0&0&1&0\end{bmatrix} \blacktriangleright \begin{bmatrix}0.4&0&0&0\\0&0.4&0&0\\0&0&0.4&0\\0&0&1&0\end{bmatrix}$$

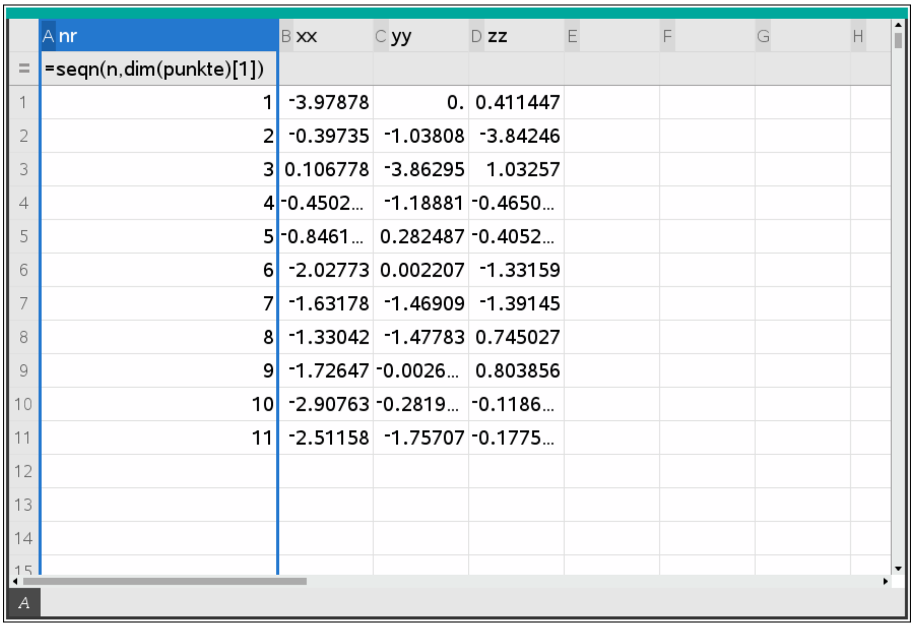
$$\mathbf{punkte}:=\begin{bmatrix}10&0&0&1\\0&10&0&1\\0&0&10&1\\1.&2.&2.54&1\\2.&1.&-1.&1\\4.7&3.7&-1.&1\\3.7&4.7&2.54&1\\3.5&-0.5&3.96&1\\4.5&-1.5&0.41&1\\7.2&1.19&0.41&1\\6.2&2.19&3.96&1\end{bmatrix} \blacktriangleright \begin{bmatrix}10&0&0&1\\0&10&0&1\\0&0&10&1\\1.&2.&2.54&1\\2.&1.&-1.&1\\4.7&3.7&-1.&1\\3.7&4.7&2.54&1\\3.5&-0.5&3.96&1\\4.5&-1.5&0.41&1\\7.2&1.19&0.41&1\\6.2&2.19&3.96&1\end{bmatrix}$$

$(\mathbf{pp2}\cdot(\mathbf{cube}[2])^{T})[2,1]$

$\mathbf{projek}(koor,xyz):=(\mathbf{pp2}\cdot\mathbf{ry}(-\mathbf{w1x})\cdot\mathbf{rx}(\mathbf{w2y})\cdot(\mathbf{punkte}[koor])^{T})[xyz\quad 1] \blacktriangleright \textit{Done}$

$\dim(\mathbf{punkte})[1]$

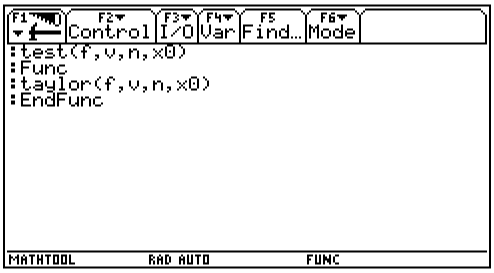
$\mathbf{project}(3) \blacktriangleright \textit{Done}$

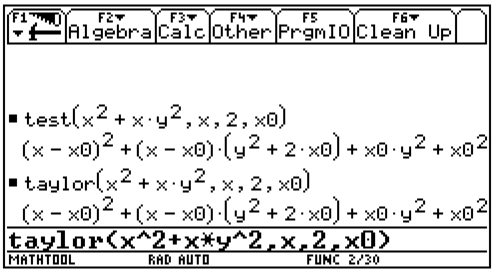


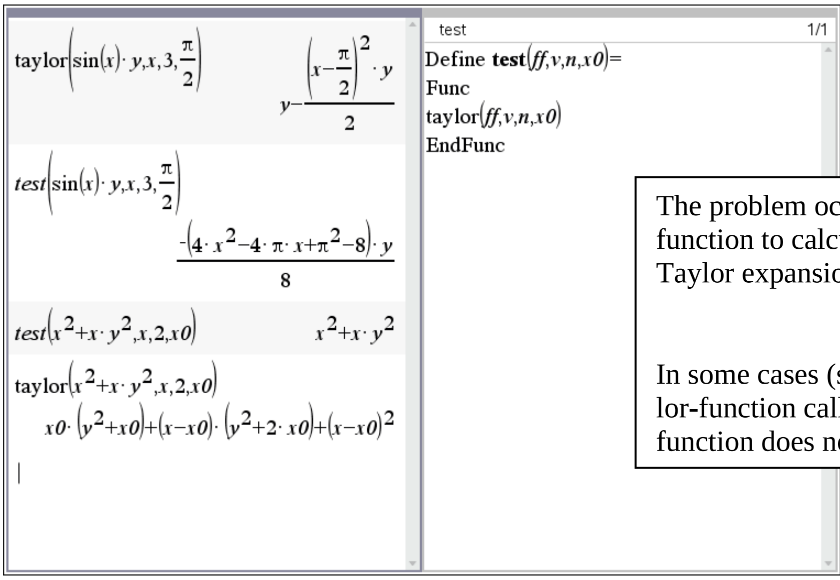
	A nr	B xx	C yy	D zz	E	F	G	H
=	=seqn(n,dim(punkte)[1])							
1	1	-3.97878	0.	0.411447				
2	2	-0.39735	-1.03808	-3.84246				
3	3	0.106778	-3.86295	1.03257				
4	4	-0.4502...	-1.18881	-0.4650...				
5	5	-0.8461...	0.282487	-0.4052...				
6	6	-2.02773	0.002207	-1.33159				
7	7	-1.63178	-1.46909	-1.39145				
8	8	-1.33042	-1.47783	0.745027				
9	9	-1.72647	-0.0026...	0.803856				
10	10	-2.90763	-0.2819...	-0.1186...				
11	11	-2.51158	-1.75707	-0.1775...				
12								
13								
14								

A Bug with Tayor?

Working with Bhuvanesh Bhatt's mathtool-library a came across a strange behavior of Nspire. Compare the V200 results with the Nspire-results:







The problem occurs within a larger function to calculate multivariate Taylor expansions.

In some cases (see 2nd example) taylor-function called with another function does not work!

Last message: It's not a bug. Shame on me, it's my fault. Just do the expansion by hand – or let TI-Nspire expand the last result!! Poor Josef …

Affine Abbildungen – Affine Mappings

Herbert Nieder, Germany

Ich beginne mit dem Abdruck einer Auswahl aus einem intensiven E-Mail-Verkehr zwischen Herbert und mir. Dieser war sehr wichtig für die endgültige Form dieses Beitrags.

I start reprinting a selection of an intense e-mail-communication between Herbert and me. This was very important for the final form of this contribution.

Sehr geehrter Herr Mag. Josef Böhm,

bei dem Versuch, affine Abbildungen im R^2 systematisch zu erfassen, bin ich auf das Problem gestoßen, dass so gut wie alle Artikel irgendwo stecken bleiben, auch Wikipedia oder zum Beispiel https://www.cg.tuwien.ac.at/courses/CG1/textblaetter/02%20Geometrische%20Transformationen.pdf

Ich wollte eine konkrete Darstellung in Form von Matrizen und keinen Baukasten. Homogene Koordinaten werden fast überall verwendet, aber die letzte Konkretisierung fehlt, z.B. habe ich nirgendwo eine Matrix gefunden, die eine Scherung an der Geraden y=mx+b darstellt, ebenso wenig eine Spiegelung an derselben, auch keine Scherungen oder Spiegelungen an Parallelen zur y-Achse.

Der Schlüssel zum Verständnis der Ergebnisse der Matrixmultiplikationen der TU Wien im Vergleich mit der Darstellung der Geraden y=m x + b als Fixgerade der Abbildungen war für mich die Gleichsetzung m = tan(alpha) und der Gleichung $1/\cos^2=\tan^2 + 1$. Als ich das zusammengebracht hatte, war der Rest "nur" noch eine Fleißarbeit.

Falls Sie mit dem Ergebnis etwas anfangen können, freut es mich, falls nicht, wäre ich für eine kurze Mitteilung dankbar.

Falls Sie die Ergebnisse brauchbar finden, aber eine andere Form wünschen, könnte ich diese liefern, seit 1.2.2018 bin ich Pensionär und kann mich kümmern, aber sonst hätte ich mich wohl auch nicht damit beschäftigt.

Beste Grüße

Herbert Nieder

Dear Mr. Böhm,
trying to capture affine mappings in R^2 systematically I came across the problem that almost all articles get stuck somewhere, even Wikipedia or for example see:

https://www.cg.tuwien.ac.at/courses/CG1/textblaetter/02%20Geometrische%20Transformationen.pdf

I wanted a concrete presentation in matrix form and not a toolbox. Homogeneous coordinates are used almost everywhere, but I miss the last step. So, I didn't find a matrix presenting a shearing at the line y = m·x + b or the reflection on the line. There are no shearings or reflections at horizontal lines.

The key for understanding the matrix multiplications given in the tuwien-article for me was taking slope m = tan(alpha) and the identity $1/\cos^2 = \tan^2 + 1$. Then it was easily done.

I would be glad if you would find the paper useful for the DNL. If not, I would be grateful for a short message. If you would prefer another form or format, then I could deliver it. I am retired since February 2018, so I can care of it …
Best regards
Herbert Nieder

Sehr geehrter Herr Nieder,

vorerst herzlichen Glückwunsch zum Erreichen des Ruhestands in - wie man ja eindrucksvoll sieht - in voller geistiger Frische und voller Schaffensdrang.

Ich finde Ihren Beitrag sehr gut und sehr nützlich. Die homogenen Koordinaten gehen ja leider immer wieder unter.

Ich möchte den Beitrag sicher veröffentlichen.

Er würde noch mehr ins Auge stechen, wenn die Transformationen durch konkrete Beispiele illustriert werden. Man braucht immer solche "Eyecatcher".

Könnten Sie eventuell Ihre dfw-Datei um ein paar Beispiele erweitern.

Ich würde dann für einen zusätzlichen Transfer auf die TI-Nspire-Plattform sorgen.

Mit lieben Grüßen und vielem Dank für den Beitrag, der sicherlich keine „Orchideenanwendung“ des CAS ist.

Ihr Josef

Dear Mr. Nieder,

first of all, my hearty congratulations for reaching retirement in – how it can be seen very impressively – in full mental freshness and full creative urge.

I find your contribution very fine and very useful as well. Homogeneous coordinates often don’t get the attention which they deserve. I’d like to publish your article certainly.

It would be great to illustrate the transformations with concrete examples. One always needs such „eyecatchers”.

I could care for an additional transfer on the TI-Nspire platform.

Best regards and many thanks for your paper, which is really not an ”Orchids-Application”.

Yours Josef

… Einige mails später / Some mails later:

Lieber Josef,

ich habe die meisten Deiner Vorschläge eingearbeitet, allerdings nicht die Kombination der Drehung und der Streckung, die habe ich in Form eines Beispiels einer Matrizenmultiplikation an das Ende gestellt, die verschiedene Zentren erlaubt, sozusagen zum Weiterprobieren mit verschiedenen Kombinationen.

Der Artikel kommt mir mittlerweile ziemlich groß vor, nicht nur wegen der Anzahl der Bytes (zu groß?). Andererseits ist eine Entwicklung zu sehen - von der ursprünglichen 1:1-Umsetzung der homogenen Koordinaten bis zu den Funktionen, die direkt mit den Punkten arbeiten - auf die ich ungern verzichten würde.

Sehr interessant finde ich übrigens das Verhalten des Kreuzes mit Abkürzung bezüglich der Scherung. Das hatte ich nicht erwartet.

Beste Grüße

Herbert

Dear Josef,

I followed most of your proposals except the combination of rotation and expansion which is included at the end in form of an example for the matrix multiplication. It allows various centres and encourages for experimenting with various combinations.

I have the impression that the article has become very large (too large?). At the other hand one can follow the development from the initial 1:1 application of the homogeneous coordinates up to functions which work directly with figures. I would not like to do without it.

I find the behaviour of the sheared Cross very interesting. I didn’t expect this.

Best regards

Herbert

Lieber Herbert,

das ist Dir sehr gut gelungen. Die "Ecke" im Kreuz ist super, da erkennt man die Transformationen viel besser.

Der Artikel ist nicht zu groß und ich finde gerade die Darstellung der Entwicklung sehr wichtig.

Wenn Du nichts dagegen hast, werde ich die gewöhnliche Drehstreckung doch noch mit hineinnehmen (sie kommt ja bei der Multiplikation von komplexen Zahlen vor).

Deine Zusammensetzung der beiden Transformationen am Schluss ist wichtig und lädt zu weiteren Experimenten ein.

Vielleicht schaffe ich noch eine Kurzform für den TI-Nspire.

Wenn der Artikel fertig ist - wird wohl etwas dauern - schicke ich Dir diesen gerne zu.

Es wäre schön, wenn uns bei einer anderen Gelegenheit nochmals eine so fruchtbare Zusammenarbeit gelänge.

Ich habe z.B. noch sehr wenig mit homogenen Koordinaten gearbeitet.

Liebe Grüße

Josef

Dear Herbert,

you have done very well. The “short cut“ in the Cross is great. One can recognize the transformations much better.

The article is definitely not too large und and like you, I find the presentation how it develops very important.

If you don’t mind, I’d like to add the ordinary twist stretch (which applies when multiplying complex numbers).

Your final composition of both transformations is important and invites for further experimenting.

Maybe, I will produce a short version for TI-NspireCAS. I will send the contribution when it will be ready – but I fear that it will last some time.

It would be great if we could cooperate – at any other occasion – once more.

I have not worked much with homogeneous coordinates; it was very fruitful for me.

Warm regards

Josef

Lieber Josef,

ich habe nichts dagegen.

Danke für die Blumen 😊

Beste Grüße

Herbert Nieder

Dear Josef,

I have no objections.

Thanks for the flowers (compliments) 😊

Beste Grüße

Herbert Nieder

Affine Abbildungen im R^2 – Affine Mappings in R^2

(in kartesischen Koordinaten – in Cartesian coordinates)

Verschiebung von $\begin{pmatrix} x \\ y \end{pmatrix}$ um $\begin{pmatrix} u \\ v \end{pmatrix}$: Translation of $\begin{pmatrix} x \\ y \end{pmatrix}$ by $\begin{pmatrix} u \\ v \end{pmatrix}$:	$\begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} u \\ v \end{pmatrix}$
Streckung/Stauchung mit Zentrum Ursprung: Extension/Dilatation with center in origin:	$\begin{pmatrix} s & 0 \\ 0 & s \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix}$
Drehung um den Ursprung (um Winkel α): Rotation around origin (by angle α):	$\begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix}$
Spiegelung an der Ursprungsgeraden $y = m\,x$: Reflection on $y = m\,x$:	$\begin{pmatrix} -\frac{m^2-1}{m^2+1} & \frac{2m}{m^2+1} \\ \frac{2m}{m^2+1} & \frac{m^2-1}{m^2+1} \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix}$
Sonderfall: Spiegelung an der y-Achse: Special case: Reflection on y-axis:	$\begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix}$
Scherung an $y = m\,x + b$ mit Faktor a: Shearing on $y = m\,x + b$ with scaling factor a:	$\begin{pmatrix} 1+a & -\frac{a}{m} \\ a \cdot m & 1-a \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} -\frac{a \cdot b}{m} \\ a \cdot b \end{pmatrix}, \quad a, m \neq 0$
Sonderfälle: Scherung an x-Achse, y-Achse: Special cases: Shearing on x-axis, y-axis:	$\begin{pmatrix} 1 & a \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ a & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix}$
Drehstreckung mit Zentrum $\begin{pmatrix} u \\ v \end{pmatrix}$ um Winkel α und Streckfaktor r: Rotation with center $\begin{pmatrix} u \\ v \end{pmatrix}$ by angle α and scaling factor r:	$\begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \cdot \begin{pmatrix} r & 0 \\ 0 & r \end{pmatrix} \cdot \begin{pmatrix} x-u \\ y-v \end{pmatrix} + \begin{pmatrix} u \\ v \end{pmatrix}$
Allgemeine affine Abbildung: General affine mapping:	$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}, \quad a_{11} \cdot a_{22} - a_{12} \cdot a_{21} \neq 0$

In my opinion it is desirable to perform the transformations by one single matrix. This allows to execute affine mappings one after the other by matrix multiplications. This can be done by working with homogeneous coordinates, which is calculating in "level 1", i.e. vector $\begin{pmatrix} x \\ y \end{pmatrix}$ changes to $\begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$.

Affine Abbildungen im R^2 als Matrizen im R^3

Affine Mappings in R^2 as Matrices in R^3

Verschiebung von $\begin{pmatrix} x \\ y \end{pmatrix}$ um $\begin{pmatrix} u \\ v \end{pmatrix}$: Translation of $\begin{pmatrix} x \\ y \end{pmatrix}$ by $\begin{pmatrix} u \\ v \end{pmatrix}$:	$\begin{pmatrix} 1 & 0 & u \\ 0 & 1 & v \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$
Streckung/Stauchung mit Zentrum(u;v) und den Faktoren r_x und r_y: Extension/Dilatation with center in (u;v) and scaling factors r_x and r_y:	$\begin{pmatrix} r_x & 0 & u\cdot(1-r_x) \\ 0 & r_y & v\cdot(1-r_y) \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$
Drehung um (u;v) um Winkel α: Rotation with center (u;v) by angle α:	$\begin{pmatrix} \cos\alpha & -\sin\alpha & -u\cdot\cos\alpha+v\cdot\sin\alpha+u \\ \sin\alpha & \cos\alpha & -v\cdot\cos\alpha-u\cdot\sin\alpha+v \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$
Spiegelung an der Geraden $y = m\,x + b$: Reflection on $y = m\,x + b$:	$\begin{pmatrix} -\frac{m^2-1}{m^2+1} & \frac{2\cdot m}{m^2+1} & -\frac{2\cdot b\cdot m}{m^2+1} \\ \frac{2\cdot m}{m^2+1} & \frac{m^2-1}{m^2+1} & \frac{2\cdot b}{m^2+1} \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$
Sonderfälle: Spiegelung an x-Achse, y-Achse: Special cases: Reflection on x-axis, y-axis:	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$
Sonderfall: Spiegelung an $x = u$: Special case: Reflection on $x = u$:	$\begin{pmatrix} -1 & 0 & -2u \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$
Scherung an $y = m\,x + b$ mit Faktor a: Shearing on $y = m\,x + b$, scaling factor a:	$\begin{pmatrix} \frac{m^2-a\cdot m+1}{m^2+1} & \frac{a}{m^2+1} & -\frac{a\cdot b}{m^2+1} \\ -\frac{a\cdot m^2}{m^2+1} & \frac{m^2+a\cdot m+1}{m^2+1} & -\frac{a\cdot b\cdot m}{m^2+1} \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$
Sonderfälle: Scherung an x-Achse, y-Achse: Special cases: Shearing on x-axis, y-axis:	$\begin{pmatrix} 1 & a & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 \\ a & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ a & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix}$
Sonderfall: Scherung an $x = u$ (Faktor a): Special case: Shearing on $x = u$ (scaling a):	$\begin{pmatrix} 1 & 0 & 0 \\ a & 1 & -a\cdot u \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$

Drehstreckung mit Zentrum $\begin{pmatrix} u \\ v \end{pmatrix}$ um Winkel α und Streckfaktor r: Rotation with center $\begin{pmatrix} u \\ v \end{pmatrix}$ by angle α and scaling factor r:	$\begin{pmatrix} r\cos\alpha & -r\sin\alpha & -u\cdot r\cos\alpha+v\cdot r\sin\alpha+u \\ r\sin\alpha & r\cos\alpha & -v\cdot r\cos\alpha-u\cdot r\sin\alpha+v \\ 0 & 0 & 1 \end{pmatrix}\cdot\begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$
Allgemeine affine Abbildung: General affine mapping:	$\begin{pmatrix} a_{11} & a_{12} & a_1 \\ a_{21} & a_{22} & a_2 \\ 0 & 0 & 1 \end{pmatrix}\cdot\begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$

Comment:

Vectors of form $\begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$ don't form a vector space. But this is not important for our special problem.

Als konkretes Beispiel wird das Dreieck mit A(1;0), B(5;0) und C(5;3) abgebildet. Die Wahl der Koordinaten ist nicht zufällig, so wollte ich, dass die Seitenlängen des Dreiecks ganzzahlig sind und ein Winkel 90° beträgt.

Die Spiegel- bzw. Scherungsgerade ist parallel zur Hypotenuse des Dreiecks ABC, um die Zeichnung insbesondere der Scherung übersichtlich zu halten.

Eine Überprüfung der Korrektheit der Matrizen ist damit aber trotz dieser "Vereinfachung" möglich.

As an example triangle A(1;0), B(5;0) und C(5;3) is mapped. I wanted to have integer coordinates with one angle 90°. Reflection axis and Shearing axis are both parallel to the hypothenuse of the triangle in order to keep the plot clear. Despite this simplification one can double check the correctness of the matrices.

Mit der folgenden Einstellung kann man Punkt A und Faktor a unterscheiden, ebenso Punkt B und y-Achsenabschnitt b.

The following setting allows distinguishing between point A and scaling a and between point B and y-intercept b as well.

#10: CaseMode := Sensitive

#11: $\left[A := \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}, B := \begin{bmatrix} 5 \\ 0 \\ 1 \end{bmatrix}, C := \begin{bmatrix} 5 \\ 3 \\ 1 \end{bmatrix}\right]$

Die folgende Vereinfachung ist ein Vorschlag von Josef und ermöglicht, das Ergebnis direkt zu zeichnen:
Following procedure (proposed by Josef) enables immediate plotting:

#12: $\begin{bmatrix} 1 & 5 & 5 & 1 \\ 0 & 0 & 3 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$ ROW [1, 2]' $= \begin{bmatrix} 1 & 0 \\ 5 & 0 \\ 5 & 3 \\ 1 & 0 \end{bmatrix}$

Für die Zeichnung des Dreiecks ABC mithilfe der Methode der verbundenen Punkte mit DERIVE ist es erforderlich, zum Schluss C mit A zu verbinden, also insgesamt A–B–C–A, daher sind es 4 Spalten in #14 bzw 4 Zeilen in der erzeugten Matrix.

For plotting triangle ABC as connected points it is necessary to finally connect C with A, which is A–B–C–A. So, we have 4 columns and then 4 rows in the produced matrix.

Spiegelung an der Geraden y = 3x/4 + 1 (#5)
Das erste Beispiel soll schrittweise behandelt werden.

Reflection on straight line y = 3x/4 + 1
The first example will be treated stepwise.

$$\#13:\quad \begin{bmatrix} -\dfrac{m^2-1}{m^2+1} & \dfrac{2\cdot m}{m^2+1} & -\dfrac{2\cdot b\cdot m}{m^2+1} \\ \dfrac{2\cdot m}{m^2+1} & \dfrac{m^2-1}{m^2+1} & \dfrac{2\cdot b}{m^2+1} \\ 0 & 0 & 1 \end{bmatrix}$$

Die Werte für m und b werden mit der Substitutionsfunktion von Derive eingesetzt und die Matrix berechnet ...

Values for m and b are substituted using DERIVE's SUBST-tool and the matrix is evaluated ...

$$\#14:\quad \begin{bmatrix} -\dfrac{\left(\frac{3}{4}\right)^2-1}{\left(\frac{3}{4}\right)^2+1} & \dfrac{2\cdot\frac{3}{4}}{\left(\frac{3}{4}\right)^2+1} & -\dfrac{2\cdot 1\cdot\frac{3}{4}}{\left(\frac{3}{4}\right)^2+1} \\ \dfrac{2\cdot\frac{3}{4}}{\left(\frac{3}{4}\right)^2+1} & \dfrac{\left(\frac{3}{4}\right)^2-1}{\left(\frac{3}{4}\right)^2+1} & \dfrac{2\cdot 1}{\left(\frac{3}{4}\right)^2+1} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \dfrac{7}{25} & \dfrac{24}{25} & -\dfrac{24}{25} \\ \dfrac{24}{25} & -\dfrac{7}{25} & \dfrac{32}{25} \\ 0 & 0 & 1 \end{bmatrix}$$

... und nacheinander mit A, B und C multipliziert.

... and one after the other multiplied by A, B and C.

$$\#15:\quad \begin{bmatrix} \dfrac{7}{25} & \dfrac{24}{25} & -\dfrac{24}{25} \\ \dfrac{24}{25} & -\dfrac{7}{25} & \dfrac{32}{25} \\ 0 & 0 & 1 \end{bmatrix}\cdot A = \begin{bmatrix} -\dfrac{17}{25} \\ \dfrac{56}{25} \\ 1 \end{bmatrix}$$

$$\#16: \quad \begin{bmatrix} \frac{7}{25} & \frac{24}{25} & -\frac{24}{25} \\ \frac{24}{25} & -\frac{7}{25} & \frac{32}{25} \\ 0 & 0 & 1 \end{bmatrix} \cdot B = \begin{bmatrix} \frac{11}{25} \\ \frac{152}{25} \\ 1 \end{bmatrix}$$

$$\#17: \quad \begin{bmatrix} \frac{7}{25} & \frac{24}{25} & -\frac{24}{25} \\ \frac{24}{25} & -\frac{7}{25} & \frac{32}{25} \\ 0 & 0 & 1 \end{bmatrix} \cdot C = \begin{bmatrix} \frac{83}{25} \\ \frac{131}{25} \\ 1 \end{bmatrix}$$

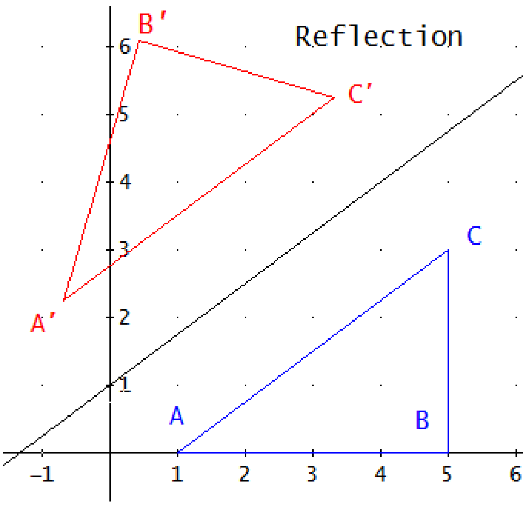
Mit der Vereinfachung (#12) können die Ergebnisse zeichenfertig dargestellt werden.

Applying procedure #12 the results can be made ready for plotting in one step.

$$\#18: \quad \left(\begin{bmatrix} \frac{7}{25} & \frac{24}{25} & -\frac{24}{25} \\ \frac{24}{25} & -\frac{7}{25} & \frac{32}{25} \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 5 & 5 & 1 \\ 0 & 0 & 3 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \right) \text{ROW } [1, 2]' = \begin{bmatrix} -\frac{17}{25} & \frac{56}{25} \\ \frac{11}{25} & \frac{152}{25} \\ \frac{83}{25} & \frac{131}{25} \\ -\frac{17}{25} & \frac{56}{25} \end{bmatrix}$$

Die verbundenen Punkte aus #12 (blau) und #20 (rot) können gezeichnet werden.

Points from #12 (blue) and #18 (red) can be plotted.



Scherung an der Geraden y = 3/4x + 1 mit dem Faktor a = 2

Shearing on the line y = 3/4x + 1 with scale factor a = 2

#19: $$\begin{bmatrix} \frac{m^2 - a\cdot m + 1}{m^2 + 1} & \frac{a}{m^2 + 1} & -\frac{a\cdot b}{m^2 + 1} \\ -\frac{a\cdot m^2}{m^2 + 1} & \frac{m^2 + a\cdot m + 1}{m^2 + 1} & -\frac{a\cdot b\cdot m}{m^2 + 1} \\ 0 & 0 & 1 \end{bmatrix}$$

eingesetzt und direkt berechnet:

substituted and evaluated:

#20: $$\begin{bmatrix} \frac{1}{25} & \frac{32}{25} & -\frac{32}{25} \\ -\frac{18}{25} & \frac{49}{25} & -\frac{24}{25} \\ 0 & 0 & 1 \end{bmatrix}$$

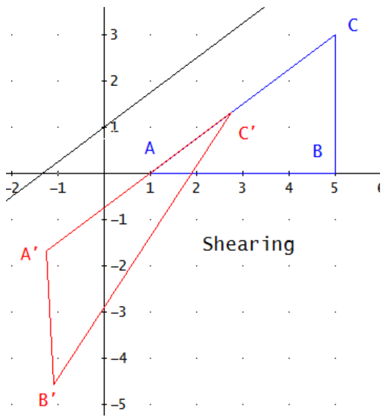
und multipliziert in einem Schritt wie oben: (auf die Berechnung für die einzelnen Punkte wird verzichtet).

and multiplied in one step as above: (without calculation of the single points).

#21: $$\left(\begin{bmatrix} \frac{1}{25} & \frac{32}{25} & -\frac{32}{25} \\ -\frac{18}{25} & \frac{49}{25} & -\frac{24}{25} \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 5 & 5 & 1 \\ 0 & 0 & 3 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}\right) \text{ROW } [1, 2]'$$

Die verbundenen Punkte aus #12 (blau) und #21 (rot) können nun gezeichnet werden. #21 muss nicht berechnet werden, sondern kann direkt gezeichnet werden.

Connected points from #12 (blue) and #21 (red) are plotted. It's not necessary to simplify #21, the expression can immediately be plotted.



Durch die zusammenfassende Darstellung ist es möglich, auch komplexere Figuren mit wenigen Schritten abzubilden.

Using the summarizing representation, it is possible to represent complex figures in a few steps.

Josef schlug folgendes Verfahren vor, bei dem nur noch die Koordinaten der Punkte eingegeben werden, aus denen die homogenen Komponenten durch Derive erzeugt werden.

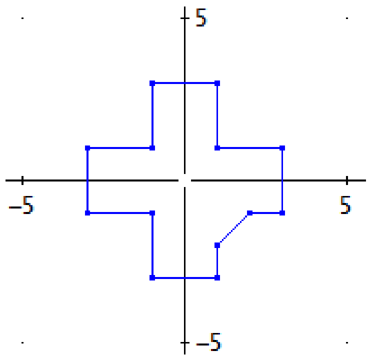
Josef proposed the following procedure for creating the homogeneous coordinates of the figures from their Cartesian coordinates.

```
#22:   make_hom(obj) := APPEND_COLUMNS(obj, [VECTOR(1, k, 1, DIM(obj))]')'
```

Als Beispiel ein Kreuz mit einer "Abkürzung", um Symmetrien zu vermindern, geschert an der Beispielgeraden y=3/4x+1. Kreuz ist als transponierte Matrix definiert, um Platz zu sparen.

Our example is a cross with a "shortcut" to reduce symmetries sheared on line y = 3/4x + 1. Kreuz is defined as transposed matrix in order to save space.

#23: $\text{Kreuz} := \begin{bmatrix} 1 & 1 & -1 & -1 & -3 & -3 & -1 & -1 & 1 & 1 & 2 & 3 & 3 & 1 \\ 1 & 3 & 3 & 1 & 1 & -1 & -1 & -3 & -3 & -2 & -1 & -1 & 1 & 1 \end{bmatrix}'$



Die folgende Anweisung dient nur dazu, das Ergebnis von make_hom zu zeigen, sie wird aber nie einzeln benötigt, sondern immer nur in Verbindung mit den Matrizen bzw. den folgenden Definitionen.

The following command serves only to demonstrate the effect of make_hom. It will never be used separately, but always in connection with the matrices in the definitions following.

```
#24:   make_hom(Kreuz)
```

hat folgendes Ergebnis, das direkt mit den Matrizen verarbeitet werden kann

gives the "Kreuz" in homogeneous coordinates for further calculations

#25: $\begin{bmatrix} 1 & 1 & -1 & -1 & -3 & -3 & -1 & -1 & 1 & 1 & 2 & 3 & 3 & 1 \\ 1 & 3 & 3 & 1 & 1 & -1 & -1 & -3 & -3 & -2 & -1 & -1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$

Definition der Funktion einer **Scherung** eines punktweise definierten Objekts obj (Polygon) an der Geraden y = m·x + b mit dem Faktor a.

Definition of function **shearing**, which transforms object obj (polygon), on line y = m·x + b by factor a.

$$\#26:\quad \text{shearing(obj, m, b, a)} := \left(\begin{bmatrix} \frac{m^2 - a\cdot m + 1}{m^2 + 1} & \frac{a}{m^2 + 1} & -\frac{a\cdot b}{m^2 + 1} \\ -\frac{a\cdot m^2}{m^2 + 1} & \frac{m^2 + a\cdot m + 1}{m^2 + 1} & -\frac{a\cdot b\cdot m}{m^2 + 1} \\ 0 & 0 & 1 \end{bmatrix} \cdot \text{make\_hom(obj)} \right) \text{ ROW [1, 2]'}$$

Scherung von Kreuz an der Geraden y = 1/2x + 3 mit den Faktoren 2 und –2

Shearing of Kreuz on line y = 1/2x + 3 with scale factors 2 and –2

#27: Kreuz

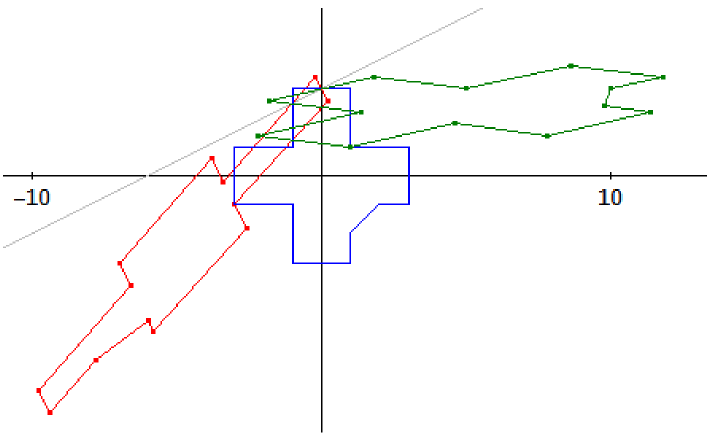
$$\#28:\quad \frac{1}{2}\cdot x + 3$$

$$\#29:\quad \text{shearing}\left(\text{Kreuz}, \frac{1}{2}, 3, 2\right)$$

$$\#30:\quad \text{shearing}\left(\text{Kreuz}, \frac{1}{2}, 3, -2\right)$$

Die Ausdrücke #29 und #30 können ohne weitere Berechnung geplottet werden.

Expressions #29 and #30 can be plotted without any further simplification.



Definition der Funktion einer **Drehung um (u;v) mit dem Winkel α.**

Definition of a **rotation with rotation center (u;v) and angle α.**

$$\#31:\quad \text{rot(obj, u, v, }\alpha) := \left(\begin{bmatrix} \text{COS}(\alpha) & -\text{SIN}(\alpha) & -u\cdot\text{COS}(\alpha) + v\cdot\text{SIN}(\alpha) + u \\ \text{SIN}(\alpha) & \text{COS}(\alpha) & -v\cdot\text{COS}(\alpha) - u\cdot\text{SIN}(\alpha) + v \\ 0 & 0 & 1 \end{bmatrix} \cdot \text{make\_hom(obj)} \right) \text{ ROW [1, 2]'}$$

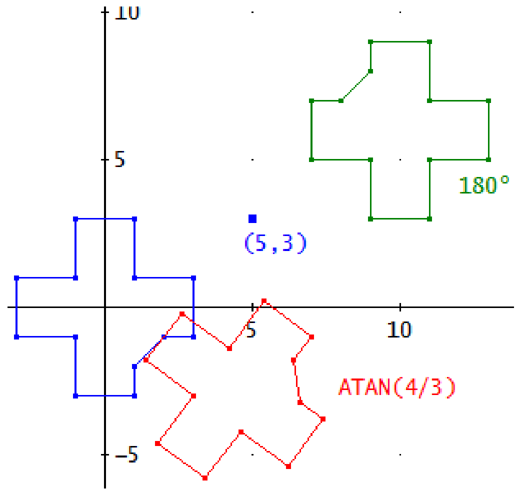
Das Beispiel von weiter oben, Drehung an (5;3) mit dem Winkel ATAN(4/3)~53,13°. Die Ausdrücke #33 und #34 (Punktspiegelung) können sofort, ohne Berechnung geplottet werden.

Example from above: center (5;3) and angle ATAN(4/3) ~ 53.13°. Expressions #33 and #34 (reflection on point) can be immediately plotted.

#32: [5, 3]

#33: $\text{rot}\left(\text{Kreuz}, 5, 3, \text{ATAN}\left(\frac{4}{3}\right)\right)$

#34: rot(Kreuz, 5, 3, 180°)

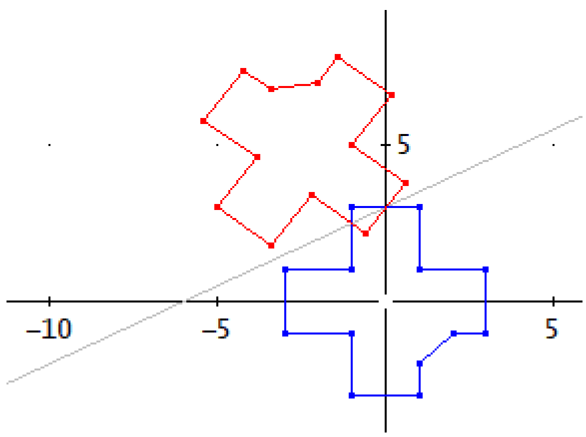


Definition der Funktion einer **Geradenspiegelung an der Geraden** y = m·x + b.
Spiegelung von Kreuz an y = 1/2x + 3.

Reflection across line y = m·x +b.
Reflection of the Cross across y = 1/2x + 3

$$\#35:\quad \text{refl\_line}(\text{obj}, m, b) := \left(\begin{bmatrix} -\frac{m^2-1}{m^2+1} & \frac{2\cdot m}{m^2+1} & -\frac{2\cdot b\cdot m}{m^2+1} \\ \frac{2\cdot m}{m^2+1} & \frac{m^2-1}{m^2+1} & \frac{2\cdot b}{m^2+1} \\ 0 & 0 & 1 \end{bmatrix}\cdot \text{make\_hom}(\text{obj})\right)\ \text{ROW}\ [1, 2]'$$

$$\#36:\quad \text{refl\_line}\left(\text{Kreuz}, \frac{1}{2}, 3\right)$$

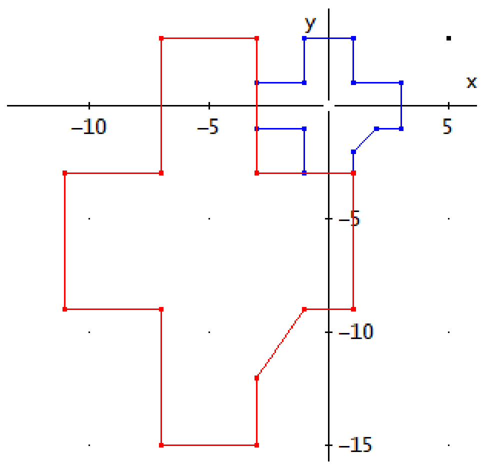


Definition der Funktion einer **Streckung mit dem Zentrum (u;v)** und dem Faktor r in x-Richtung und dem Faktor s in y-Richtung
Beispiel: Zentrum (5;3) mit den Streckfaktoren 2 und 3

Definition of a dilatation with center (u;v) and factor r in x-direction and factor s in y-direction
Example: Center (5;3) with scale factors 2 and 3

#37: $\text{stretch(obj, u, v, r, s)} := \left(\begin{bmatrix} r & 0 & u\cdot(1-r) \\ 0 & s & v\cdot(1-s) \\ 0 & 0 & 1 \end{bmatrix}\cdot\text{make\_hom(obj)}\right)\ \text{ROW}\ [1, 2]'$

#38: stretch(Kreuz, 5, 3, 2, 3)

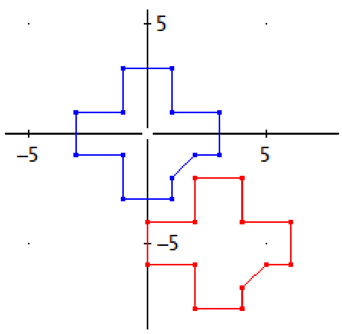


Definition einer **Verschiebung um u in x-Richtung und v in y-Richtung**
Beispiel: Horizontale Schiebung um 3, vertikale Schiebung um -5

Definition of a translation by u in x-direction and by v in y-direction
Example: Horizontal shift by 3 and vertical shift by -5

#39: $\text{shift(obj, u, v)} := \left(\begin{bmatrix} 1 & 0 & u \\ 0 & 1 & v \\ 0 & 0 & 1 \end{bmatrix}\cdot\text{make\_hom(obj)}\right)\ \text{ROW}\ [1, 2]'$

#40: shift(Kreuz, 3, -5)



Definition der Funktion einer **Drehstreckung** mit dem Zentrum (u;v), dem Drehwinkel α und dem Streckfaktor r
Beispiele: Drehstreckung Quadrat um (0;0) und 60°, bzw. um (-3,-3) und 90°
Als demo ist das Ergebnis der Drehung (1. Schritt) extra eingezeichnet.

Definition of a **twist stretch (rotation enlargement)** with its center in (u;v), rotation angle α and scale factor r
Examples: Mapping of a square with center (0;0) by 60° and with center (-3,-3) by 90°. For demonstration purpose the result of the rotation (1st step) is plotted additionally.

$$\#41:\quad \mathrm{rot\_stretch}(obj,\ u,\ v,\ r,\ \alpha) := \left(\begin{bmatrix} r\cdot\mathrm{COS}(\alpha) & -r\cdot\mathrm{SIN}(\alpha) & -r\cdot u\cdot\mathrm{COS}(\alpha) + r\cdot v\cdot\mathrm{SIN}(\alpha) + u \\ r\cdot\mathrm{SIN}(\alpha) & r\cdot\mathrm{COS}(\alpha) & -r\cdot v\cdot\mathrm{COS}(\alpha) - r\cdot u\cdot\mathrm{SIN}(\alpha) + v \\ 0 & 0 & 1 \end{bmatrix}\cdot\mathrm{make\_hom}(obj)\right)\ \mathrm{ROW}\ [1,\ 2]'$$

$$\#42:\quad sq := \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}'$$

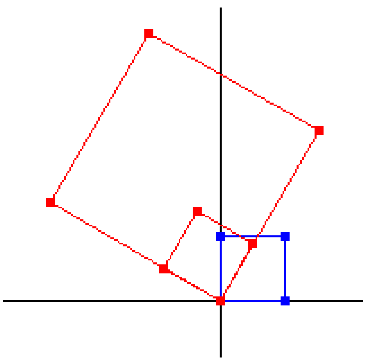
$$\#43:\quad \mathrm{rot\_stretch}\left(sq,\ 0,\ 0,\ 3,\ \frac{\pi}{3}\right)$$

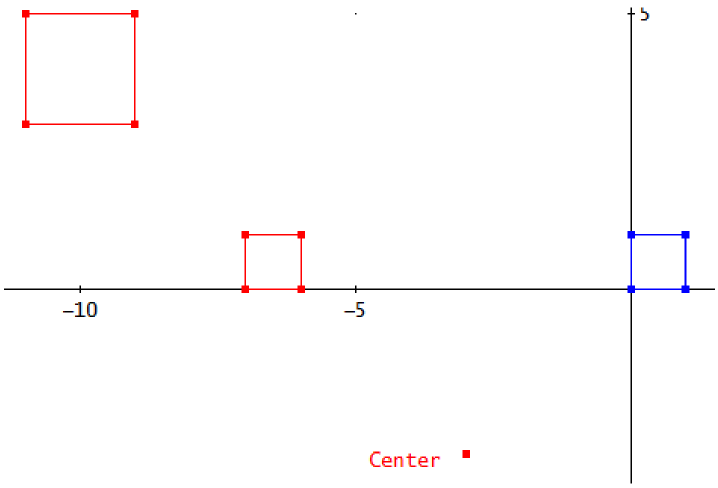
$$\#44:\quad \mathrm{rot}\left(sq,\ 0,\ 0,\ \frac{\pi}{3}\right)$$

$$\#45:\quad \mathrm{rot\_stretch}(sq,\ -3,\ -3,\ 2,\ 90°)$$

$$\#46:\quad \mathrm{rot}(sq,\ -3,\ -3,\ 90°)$$

$$\#47:\quad [-3\ \ -3]$$

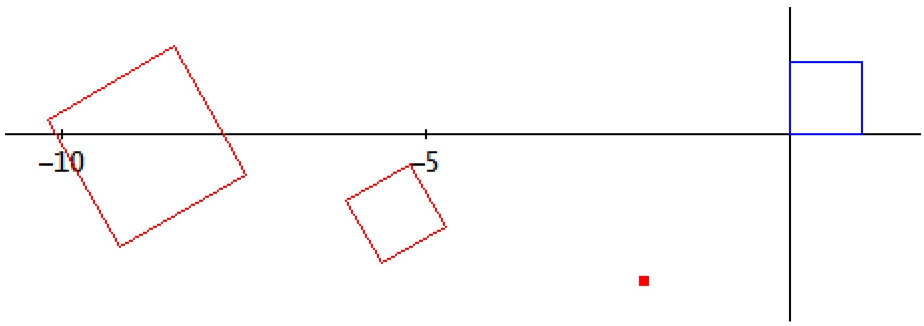




$$\#49:\quad \mathrm{rot\_stretch}(sq,\ -2,\ -2,\ 2,\ 120°)$$

$$\#50:\quad \mathrm{rot}(sq,\ -2,\ -2,\ 120°)$$

$$\#51:\quad [-2,\ -2]$$



So funktioniert die Drehstreckung:

Das Drehzentrum wird (mit der Figur) in den Ursprung verschoben, dann wird gedreht und gestreckt tmp); anschließend muss die Verschiebung wieder rückgängig gemacht werden.

How this mapping works:
The rotation center is shifted (together with the figure) into the origin, then rotation and expansion are performed; finally, the shift must be reversed.

$$\#53:\quad \text{VECTOR}(v\_ - [u,\ v],\ v\_,\ sq)' = \begin{bmatrix} -u & 1-u & 1-u & -u & -u \\ -v & -v & 1-v & 1-v & -v \end{bmatrix}$$

$$\#54:\quad tmp := \left(\begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix} \cdot \begin{bmatrix} r & 0 \\ 0 & r \end{bmatrix} \cdot \begin{bmatrix} -u & 1-u & 1-u & -u & -u \\ -v & -v & 1-v & 1-v & -v \end{bmatrix}\right)'$$

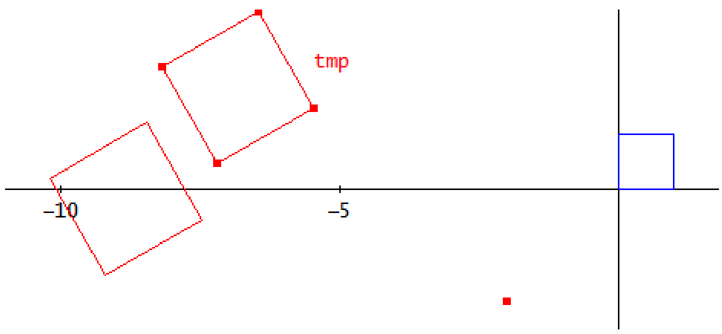
$$\#55:\quad \text{VECTOR}(v\_ + [u,\ v],\ v\_,\ tmp)$$

$$\#56:\quad \begin{bmatrix} -r\cdot u\cdot \cos(\alpha) + r\cdot v\cdot \sin(\alpha) + u & -r\cdot v\cdot \cos(\alpha) - r\cdot u\cdot \sin(\alpha) + v \\ r\cdot(1-u)\cdot \cos(\alpha) + r\cdot v\cdot \sin(\alpha) + u & -r\cdot v\cdot \cos(\alpha) + r\cdot(1-u)\cdot \sin(\alpha) + v \\ r\cdot(1-u)\cdot \cos(\alpha) + r\cdot(v-1)\cdot \sin(\alpha) + u & r\cdot(1-v)\cdot \cos(\alpha) + r\cdot(1-u)\cdot \sin(\alpha) + v \\ -r\cdot u\cdot \cos(\alpha) + r\cdot(v-1)\cdot \sin(\alpha) + u & r\cdot(1-v)\cdot \cos(\alpha) - r\cdot u\cdot \sin(\alpha) + v \\ -r\cdot u\cdot \cos(\alpha) + r\cdot v\cdot \sin(\alpha) + u & -r\cdot v\cdot \cos(\alpha) - r\cdot u\cdot \sin(\alpha) + v \end{bmatrix}$$

Die Werte werden eingesetzt und die Matrix wird berechnet.

Values are substituted and the matrix is calculated.

$$\#57:\quad \begin{bmatrix} -2\cdot\sqrt{3} - 4 & 2\cdot\sqrt{3} - 4 \\ -2\cdot\sqrt{3} - 5 & 3\cdot\sqrt{3} - 4 \\ -3\cdot\sqrt{3} - 5 & 3\cdot\sqrt{3} - 5 \\ -3\cdot\sqrt{3} - 4 & 2\cdot\sqrt{3} - 5 \\ -2\cdot\sqrt{3} - 4 & 2\cdot\sqrt{3} - 4 \end{bmatrix}$$



Nun noch ein Beispiel dafür, wie einfach es ist, affine Abbildungen mit homogenen Koordinaten zu kombinieren; hier die Kombination aus Drehung um (5;3) um 90° mit anschließender Streckung mit dem Zentrum (10;6) und den Faktoren 2 und 3. Die Durchführung erfolgt schrittweise, und ist damit vorerst nachvollziehbar.

We will finish with one example which will demonstrate how easy it is to combine affine mappings using homogeneous coordinates; see a combination of a rotation by 90° with center (5;3) followed by a dilatation with center (10;6) with factors 2 and 3. Execution will be performed stepwise which makes it comprehensible.

Ausdruck #3 (hier nicht abgedruckt, aber im dfw-file enthalten) wird angewendet:

Using expression #3 – not printed here, but contained in the dfw-file:

Substituieren mit u=5, v=3, α=90° in #3 und berechnen ergibt:

Substitute 5, 3, and 90° for u, v and α in #3 and simplify:

#58: $\begin{bmatrix} \text{COS}(\alpha) & -\text{SIN}(\alpha) & -u\cdot\text{COS}(\alpha)+v\cdot\text{SIN}(\alpha)+u \\ \text{SIN}(\alpha) & \text{COS}(\alpha) & -v\cdot\text{COS}(\alpha)-u\cdot\text{SIN}(\alpha)+v \\ 0 & 0 & 1 \end{bmatrix}$

#59: $\begin{bmatrix} 0 & -1 & 8 \\ 1 & 0 & -2 \\ 0 & 0 & 1 \end{bmatrix}$

Substituiere 10, 6, 2, und 3 für u, v, r, und s in #2 (Streckung) und berechne:

Substitute 10, 6, 2, and 3 for u, v, r and s in #2 (expansion) and simplify:

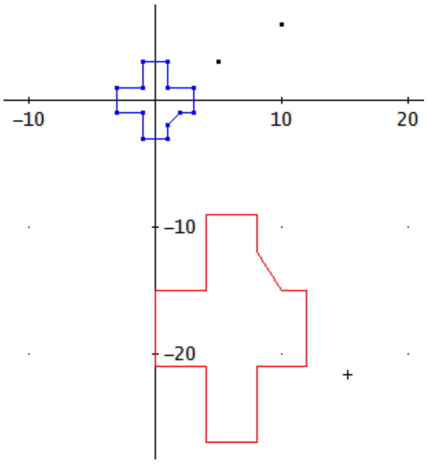
#60: $\begin{bmatrix} 2 & 0 & -10 \\ 0 & 3 & -12 \\ 0 & 0 & 1 \end{bmatrix}$

Eingeben von (#60*#59*#24)ROW[1,2]' in der Eingabezeile erzeugt #61 – und kann sofort geplottet werden (gemeinsam mit dem Original Kreuz). Die beiden Zentren werden auch gezeichnet.

Editing (#60*#59*#24)ROW[1,2]' in the Entry line gives #61 – and can immediately be plotted (together with the original Kreuz). Both centers are plotted, too.

#61: $\left(\begin{bmatrix} 2 & 0 & -10 \\ 0 & 3 & -12 \\ 0 & 0 & 1 \end{bmatrix}\cdot\begin{bmatrix} 0 & -1 & 8 \\ 1 & 0 & -2 \\ 0 & 0 & 1 \end{bmatrix}\cdot\text{make\_hom(Kreuz)}\right)\text{ROW }[1, 2]'$

#62: $\begin{bmatrix} [10, 6] \\ [5, 3] \end{bmatrix}$



Oder kürzer unter Verwendung der Funktionen als "Black Box". Das führt zum gleichen Graph.

Or shorter using the functions as "Black Box". Results in the same graph.

#63: stretch(rot(Kreuz, 5, 3, 90°), 10, 6, 2, 3)

The Nspire users are invited to investigate the various mappings applying the procedure proposed below. DERIVE makes plotting more comfortable.

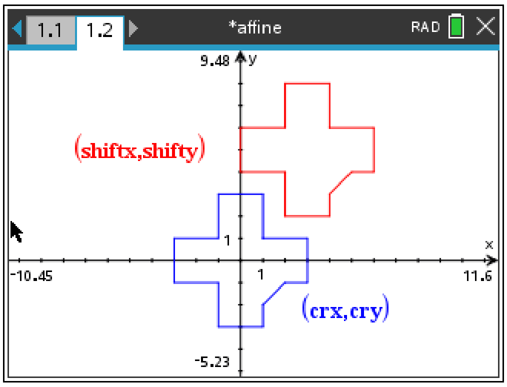
```
cr:=[1 1 -1 -1 -3 -3 -1 -1 1 1 2 3 ▸
     1 3 3 1 1 -1 -1 -3 -3 -2 -1 -1]
  [1 1 -1 -1 -3 -3 -1 -1 1 1 2 3 3 ▸
   1 3 3 1 1 -1 -1 -3 -3 -2 -1 -1 1]
plot_gr(cr)
                         coord. in lists listx & listy
                                                  Done
crx:=listx:cry:=listy
              {1,3,3,1,1,-1,-1,-3,-3,-2,-1,-1,1,1}
make_hom(cr)
```

```
plot_gr                                            3/3
Define plot_gr(obj)=
Prgm
listx:=mat▸list(obj[1])
listy:=mat▸list(obj[2])
Disp "coord. in lists listx & listy"
EndPrgm
```

```
transl                                             2/3
Define transl(obj,u,v)=
Func
Local tmp
       [1 0 u]
tmp:=  [0 1 v] · make_hom(obj)
       [0 0 1]
subMat(tmp,1,1,2,dim(obj)[2])
EndFunc
```

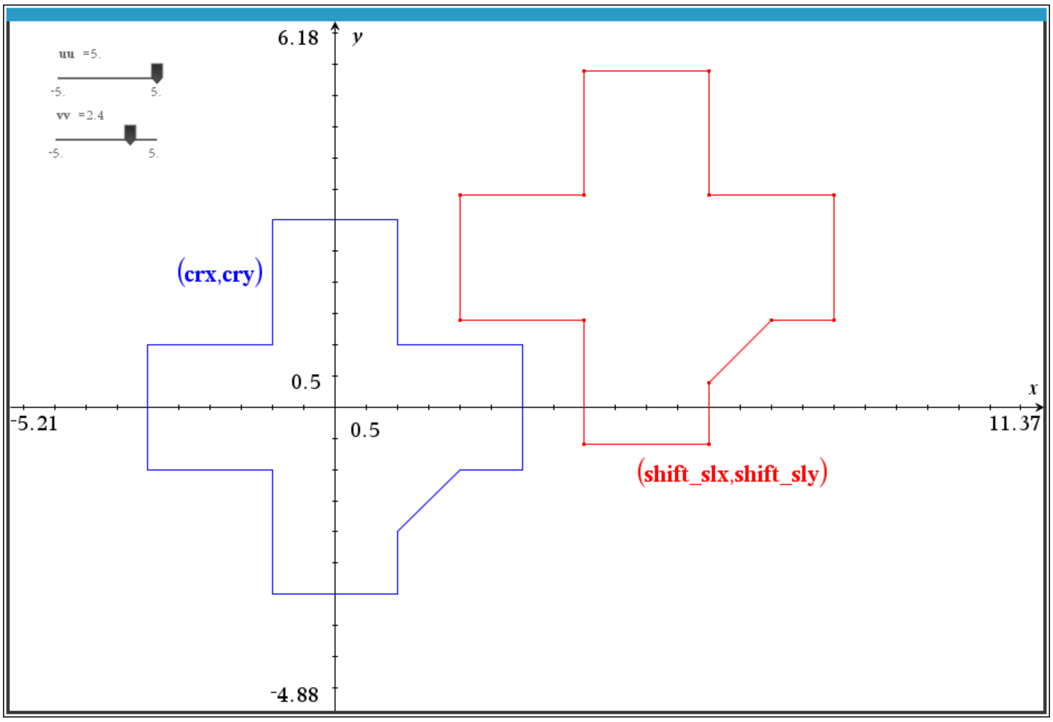
```
make_hom                                           2/2
Define make_hom(obj)=
Func
Local tmp
(augment(objᵀ,(list▸mat(seq(1,i,1,dim(obj)[2])))ᵀ))ᵀ
EndFunc
```

```
1.1  1.2          *affine                  DEG
transl(cr,3,5)
  [4 4 2 2 0 0 2 2 4 4 5 6 6 4]
  [6 8 8 6 6 4 4 2 2 3 4 4 6 6]
plot_gr(transl(cr,3,5))
                   coord. in lists listx & listy
                                            Done
shiftx:=listx:shifty:=listy
             {6,8,8,6,6,4,4,2,2,3,4,4,6,6}
```

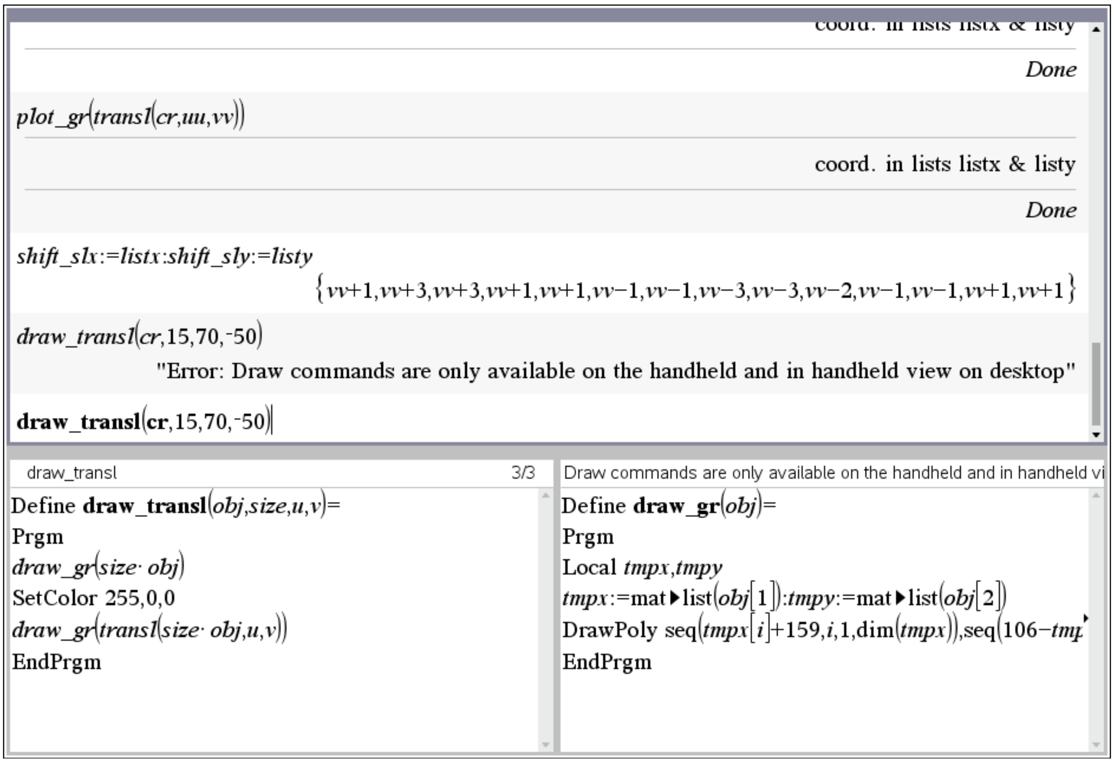


All mappings can be animated installing sliders (what can be done in DERIVE as well!!). See here the translation of the Cross as an example:

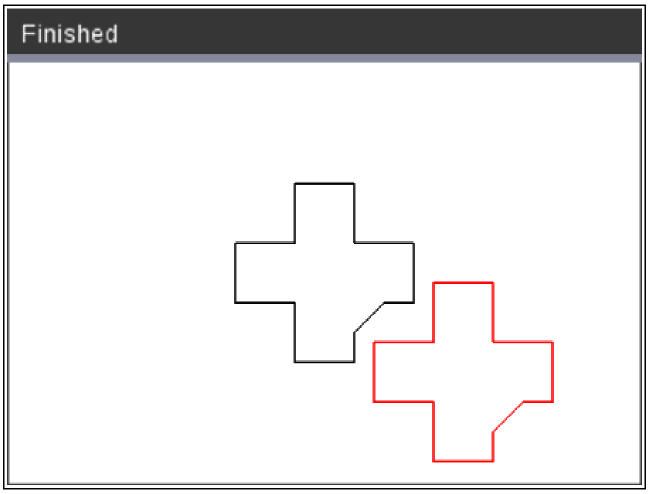
```
plot_gr(transl(cr,uu,vv))
                                 coord. in lists listx & listy
                                                          Done
plot_gr(transl(cr,uu,vv))
                                 coord. in lists listx & listy
                                                          Done
shift_slx:=listx:shift_sly:=listy
    {vv+1,vv+3,vv+3,vv+1,vv+1,vv−1,vv−1,vv−3,vv−3,vv−2,vv−1,vv−1,vv+1,vv+1}
```



The latest TI-Nspire version provides the draw functions which do plotting the polygons without storing the point lists in the Graphs App's entry. At the other hand for "drawing" more than one object it needs writing a – small – program in all cases and – what is important – one has to consider another system of coordinates (318 x 212 pixels) with (0,0) in the right above corner of the screen. You can see one example performed not using all possibilities of the Draw commands.



After the Draw screen showing the Cross together with a translation the contribution is finished with a composition of two shearings.



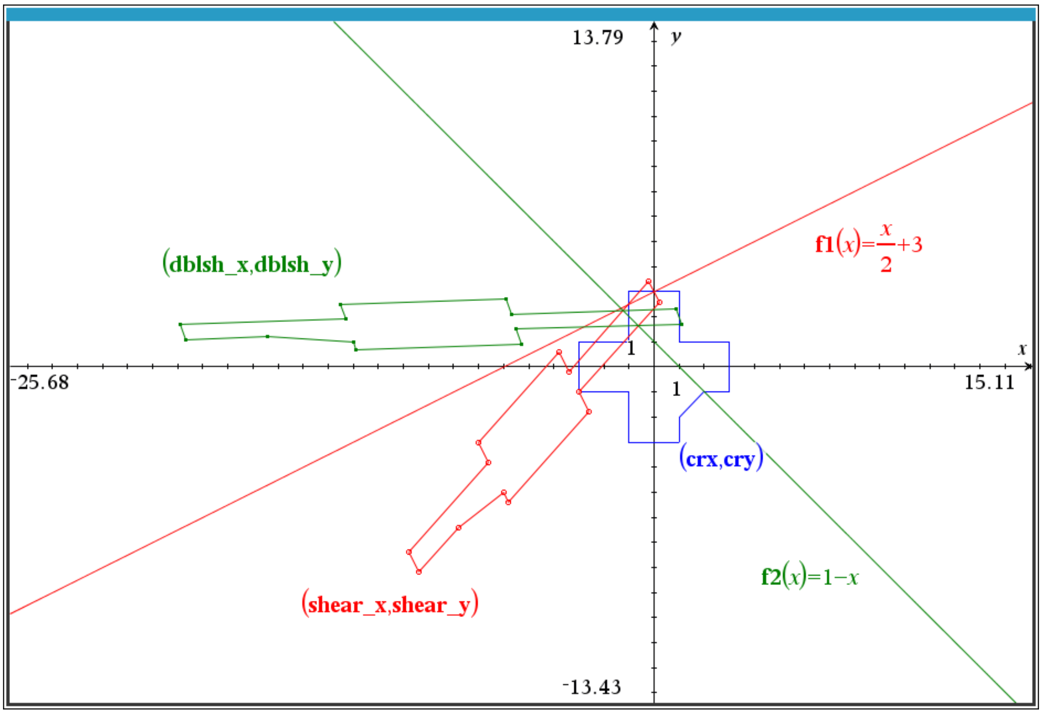
$$sh\_m(m,b,a):=\begin{bmatrix} \frac{m^2-a\cdot m+1}{m^2+1} & \frac{a}{m^2+1} & \frac{^-a\cdot b}{m^2+1} \\ \frac{^-a\cdot m^2}{m^2+1} & \frac{m^2+a\cdot m+1}{m^2+1} & \frac{^-a\cdot b\cdot m}{m^2+1} \\ 0 & 0 & 1 \end{bmatrix}$$ *Done*

$plot\_gr\left(\text{subMat}\left(sh\_m(^-1,1,1)\cdot sh\_m\left(\frac{1}{2},3,2\right)\cdot make\_hom(cr),1,1,2,\dim(cr)[2]\right)\right)$

coord. in lists listx & listy

Done

$dblsh\_x:=listx:dblsh\_y:=listy$ $\left\{\frac{3}{2},\frac{17}{10},\frac{23}{10},\frac{21}{10},\frac{27}{10},\frac{5}{2},\frac{19}{10},\frac{17}{10},\frac{11}{10},\frac{6}{5},1,\frac{7}{10},\frac{9}{10},\frac{3}{2}\right\}$



Wolfgang Alvermann showed in our last DNL his mathematics final exam from 1968. Now we will compare this with a nowadays final exam. It is a central exam for secondary vocational school from Germany. Second part (Blocks Stochastics and Analytic Geometry + Solutions will be given in next DNL, Josef)

End examination consists of two parts:

A compulsory part with 60 minutes working time, no CAS permitted, one can reach 26 points.

An electoral part consisting of three blocks (Analysis, Stochastics and Analytic Geometry; 46 pts, 24 pts, 24 pts) with two tasks each. One task from each block must be chosen.

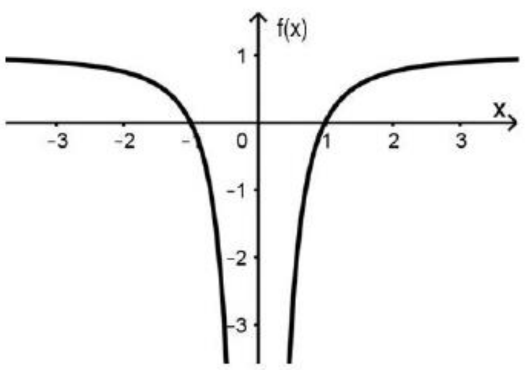
CAS and formulary (provided by school) are permitted.

Students have 30 minutes time for choosing the tasks and the 240 minutes time for working.

Compulsory Part (26 points)

Task P1

Given is function *f* with $f(x) = 1 - \frac{1}{x^2}, x \in \mathbb{R}, x \neq 0$ with zeros at $x_1 = -1$ and $x_2 = 1$. The figure shows the graph of *f*, which is symmetrical wrt the *y*-axis. Additionally given is line *g* with $g(x) = -3$.



a) Show that the x-coordinate of one of the intersection points of *f* and *g* equals $\frac{1}{2}$.

b) Calculate the area formed by the graph of *f*, the *x*-axis and line *g*.

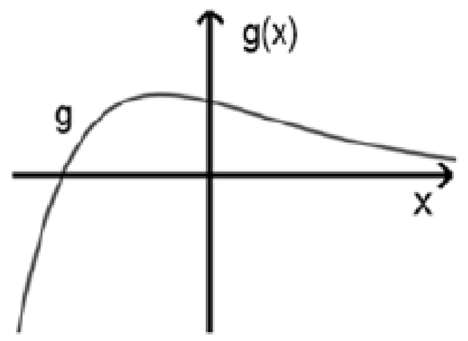
Task P2

The figure shows the graph of a differentiable function *g* defined in $\mathbb{R}$.

Considered is a function *f*, defined in $\mathbb{R}$ with its first derivative $f'(x) = e^{g(x)}$.

a) Investigate if the graph of *f* has a turning point.

b) Investigate if the graph of *f* has an inflection point.



Task P3

A wheel of fortune shows five equally sized sectors. One of them is labeled with 0, one with 1 and one with 2. The remaining two sectors are labeled with 9.

a) The wheel is turned four times. What is the probability, that numbers 2, 0, 1 and 9 appear in this given order?

b) The wheel is turned two times. What is the probability that the sum of the numbers gained is at least 11?

Task P4

In a Cartesian system of coordinates two planes *E* and *F* are given:

$$E\colon \vec{x} = \begin{pmatrix} 1 \\ -1 \\ 5 \end{pmatrix} + \lambda \cdot \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix} + \mu \cdot \begin{pmatrix} 2 \\ 3 \\ -4 \end{pmatrix}, \lambda, \mu \in IR \quad \text{and } F\colon -x + 2y + z = 1.$$

a) Give the coordinates of one point lying in *F*.

b) Show that *F* and *E* are parallel.

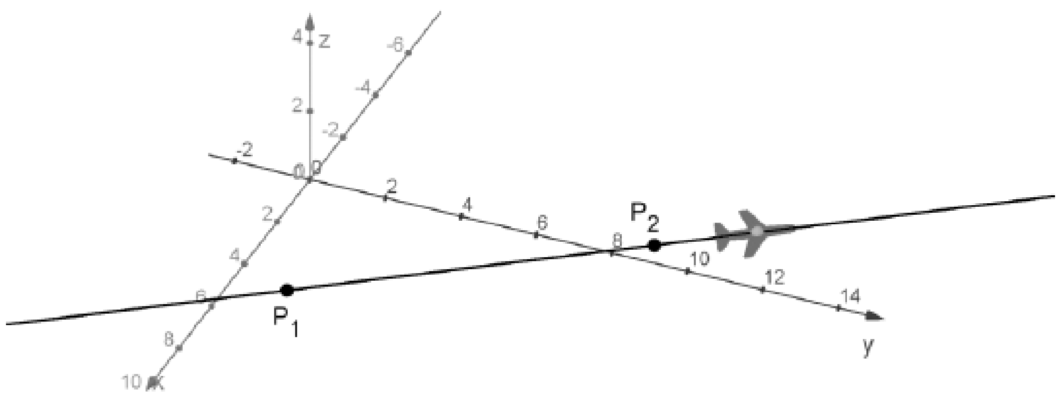
c) Give the equation of a plane which is vertical to *F* and passes the origin.

Task P5

An airplane moves with constant velocity on a straight line track. At 15:30 it passes point $P_1(6|2|1)$. One minute later it is at point P_2 $(2|10|2)$. All data are km.

a) Explain why the plane is ascending and calculate its rate of climb.

b) The cruising altitude shall be 10 km. Calculate at which position the plane will reach this altitude.



Electoral part Block Analysis – one task must be chosen, 46 points can be achieved

Task 1A

City administration plans building a new water tower (figure 1) with modern design for its water system with conditions as follow:

1. Outside diameter on the floor: d_u = 20 m
2. Height: h = 35 m (without roof)
3. Ratio v of outside diameter in height 35 m to diameter of constriction v = 3.
4. Diameter of constriction d_E = 10 m which must be in a height of at least 10 m.

Figure 1

The contracted engineering office wants to produce a model with a 3D-printer and needs the equations of functions to describe the contour of the lying tower.

First designs provide describing the outside contour by the family of functions kA_b with $kA_b(x) = 17 \cdot e^{-bx} + 0.617x - 7$.
(x and $kA_b(x)$ in meter and b in $\mathbb{R} > 0$).

a) Show that the family of functions kA_b fulfills the 1st condition and justify that this does not depend on parameter b.

   In order to adapt the form of the water tower if necessary, the influence of parameter b on the form of the graph of kA_b shall be investigated.

   Describe how the form changes when parameter b is varied.

   Calculate the value of b such that the 3rd condition is also fulfilled. Investigate if b = 0.1068 satisfies the fourth condition.

For all following parts of the task take $kA_b(x) = 17 \cdot e^{-0.107x} + 0.617x - 7$.

b) The water tower shall contain a volume of 5 000 m³ between inner wall and intermediate floor. The intermediate floor is at a height of 20 m.

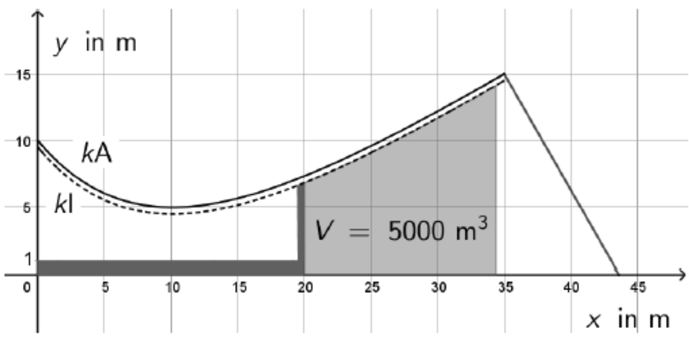


Figure 2

   The inside contour of the water tower is described by $kI(x) = 17 \cdot e^{-0.107x} + 0.617x - 7.5$.
   Calculate the height where a sensor must be installed to display the filling with 5 000 m³.

The outer shell of the tower – without roof – shall be provided with a protective coating. Calculate the amount of color if one needs one liter for 5 square meters.

(Outer shell: $M = 2\pi \cdot \int_a^b f(x) \cdot \sqrt{1+(f'(x))^2}dx$)

c) In order to support the roof of the water tower some struts are to be attached in the heigth of 35 m, which should be perpendicular to the outer wall. One of these struts is given in figure 3 as an example. Strut *s* runs from point *W* on the outer wall to point *D* on the *x*-axis.

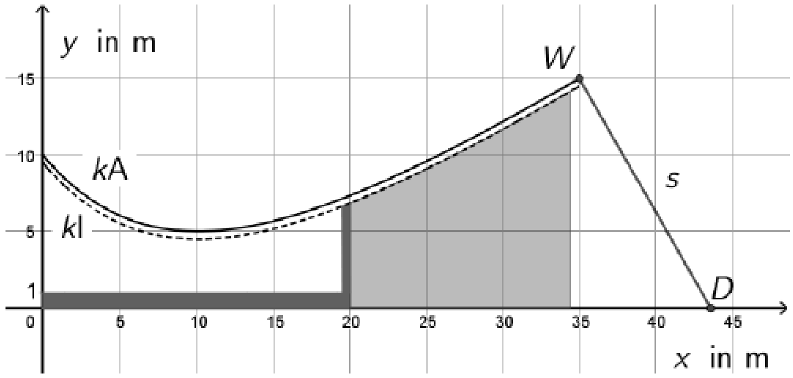


Figure 3

Calculate the height of the tower including the roof and the length of the struts.

Task 1B

A production facility for filling and labeling juice bottles is planned (figure 1). After filling in the filling station the bottles are transported upwards by a spiral conveyor into the *xy*-plane. From there they shall be carried further in the *xy*-plane by a to be developed conveyor *t* to the labeling machine *h*.

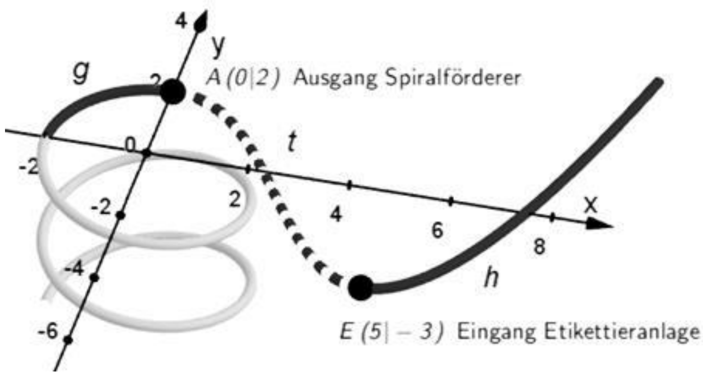


Figure 1

The course of the production can in its single parts be described by functions *x*, *g*(*x*), *t*(*x*) and *h*(*x*) in meter.

a) It must be guaranteed that transition in points *A* and *E* is smooth (i.e free of mismatch and kink and without a change in curvature) that the bottles will not roll over on the transport course (figure 1).

Find the polynomial function *t* which connects function *g* in point *A* with function *h* in point *E* according to the requirements.

For the transport course a conveyor belt must be ordered. Its length must be 2.1 times the length of the course because of the redirection. Calculate the length of the belt.

In the following the transport course is described by the function

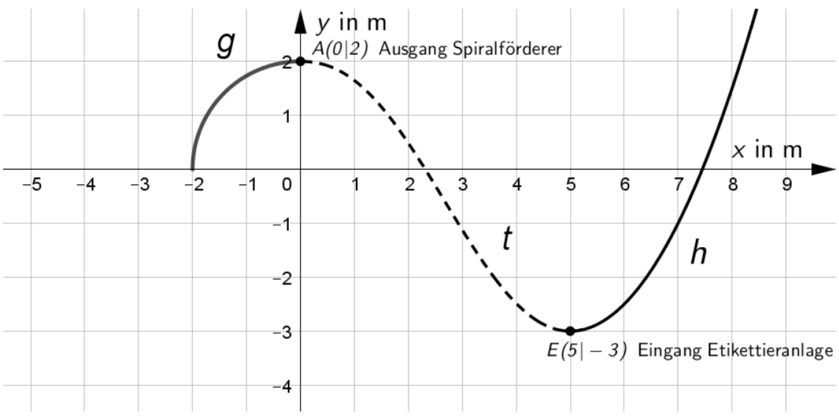
$t(x) = -0.0036x^5 + 0.05x^4 - 0.15x^3 - 0.25x^2 + 2$ for $0 < x < 5$.

b) A camera shall be attached on the wall that it has minimum distance to the transport course to control the juice bottles. The wall can be described by function *w* as follows.

$$w(x)=\begin{cases}-2.5 & \text{for } -5\le x<3.5\\ -x+1 & \text{for } 3.5\le x\le 5\\ -4 & \text{for } 5<x\le 9\end{cases}$$

Add a sketch of the wall in the coordinate system given below.

b) Planungsskizze Transportbahn, Wand und Kameraposition

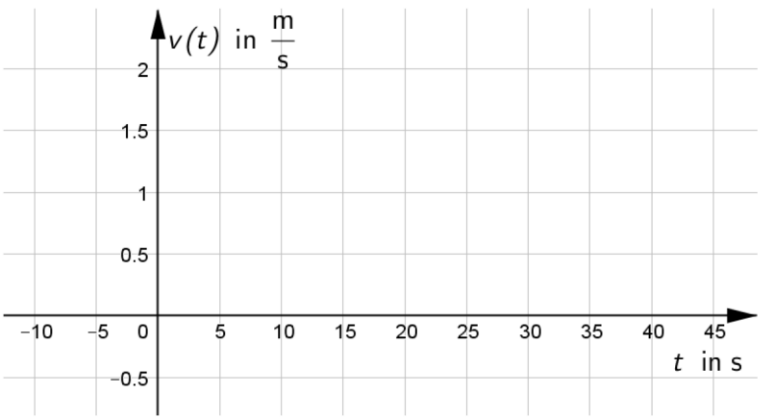


The production manager proposes point $P(3.755|-2.755)$ to fix the camera and explains that he has measured this with a right angle along wall *w*. What is the distance of this point to the transport course?

c) The conveyor must be stopped when the products are changed or when production must be interrupted. That there don't remain unlabeled bottles on the labeling machine the belt must run 50 m behind. The velocity of the belt is constant v_B = 2 m/s during production. Velocity *v* during transition from production to halt can be described approximately by function v_N with $v_N(t) = 6.25\cdot10^{-5}\,t^3 - 0.00375\,t^2 + 2$.

Change of velocity must not exceed 0.1m/s² to avoid rolling over of the bottles. Give a sketch of *v*(*t*) in the coordinate system:

c) Geschwindigkeit *v* beim Übergang vom Betrieb zum Stillstand

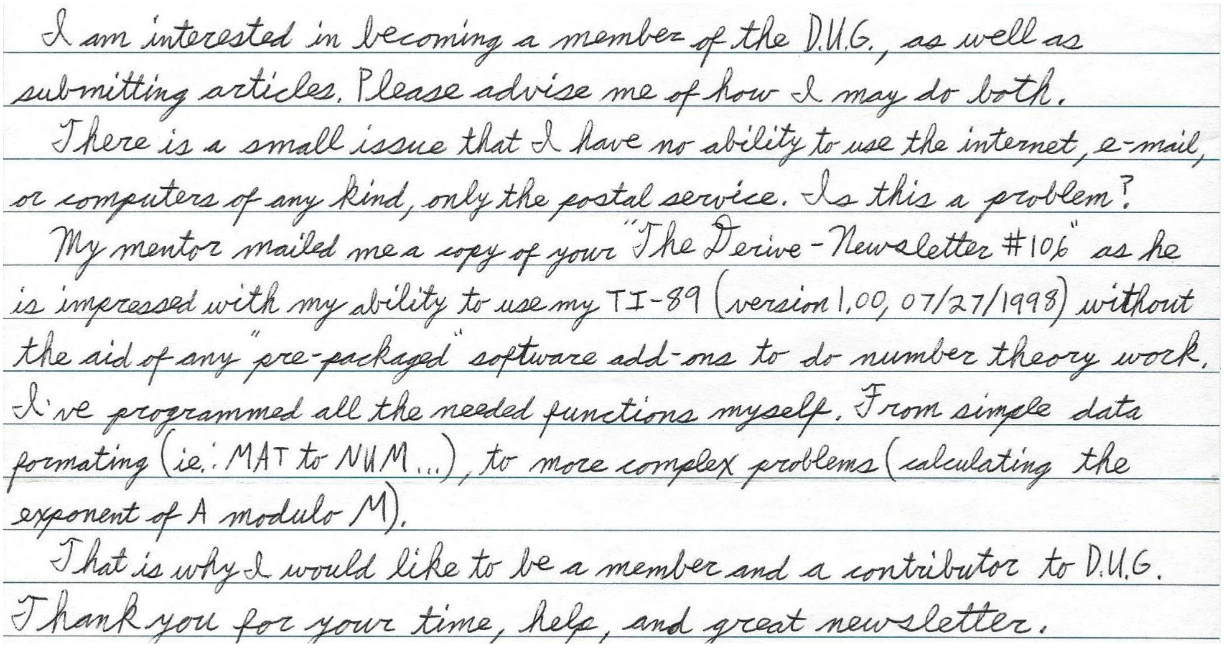


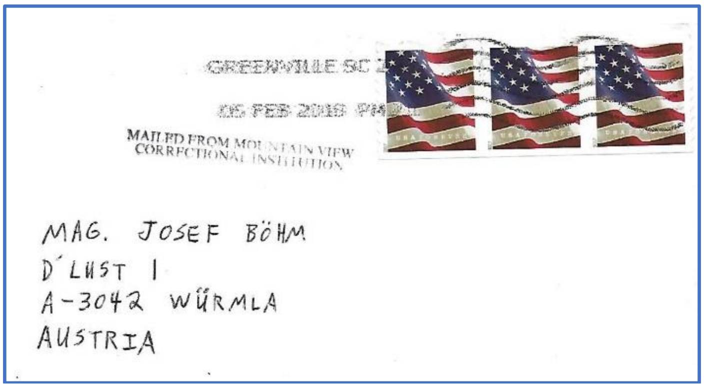
Check if the requirements with regard to the change of velocity is fulfilled.

Specify the time necessary for the belt to run after stopping the production with constant velocity that the belt will run totally 50 m behind.

TI-89 Keystroke Functions

It was in February 2018 when I received a hand written letter from the United States:


I am interested in becoming a member of the D.U.G., as well as submitting articles. Please advise me of how I may do both.
There is a small issue that I have no ability to use the internet, e-mail, or computers of any kind, only the postal service. Is this a problem?
My mentor mailed me a copy of your "The Derive-Newsletter #106" as he is impressed with my ability to use my TI-89 (version 1.00, 07/27/1998) without the aid of any "pre-packaged" software add-ons to do number theory work. I've programmed all the needed functions myself. From simple data formating (ie: MAT to NUM...), to more complex problems (calculating the exponent of A modulo M).
That is why I would like to be a member and a contributor to D.U.G.
Thank you for your time, help, and great newsletter.



This was a little bit strange. Then I inspected the envelop and found out that the postmark read: *Mailed from Mountain View Correctional Institute*. Now it became clear for me. This letter was sent from a prison. I wrote back and some time later the DUG Application form arrived.

Last August Matthew's mentor, Dr. Robert Haas, sent a mail with Matthew's contribution attached. The paper was typed – including the program codes, because Matthew has no computer to store the programs using TI-Connect or GraphLink – by Dr. Haas.

I wanted to publish Matthew's article because of its history and its background and asked back if I would be allowed to report the background of the contribution. Matthew wrote another hand written letter which was scanned by Dr. Haas and sent to me. You can find it on the following pages.

I must admit that I was – and still am – very moved. So I typed in the codes in my V200, because of the larger screen – all programs work thanks Dr. Haas' careful transcribing the code (see a part of Matthew's original paper on page 42). He does a wonderful job as Matthew's mentor, many thanks.

Dear Dr. Böhm,

Thank you for accepting my paper! I understand some of you listed concerns (20.8.20 email), but am not sure if I can fix them. I do not have access to the internet, or even a computer, so I can not send you the TI-89 files. Is it possible to open a "skeleton" TI-89 file and then copy in the code from my paper, in say Notepad? I am glad to see you were able to use my f-2n(VA) function on the TI-Nspire CAS-platform with only minor syntax changes. Was the conversion manual, or automatic (by the platform)? I understand how boring it is to retype the programs as I have had to do so several times after crashing my TI-89. The last time was about a month ago... I sometimes ask it to do too much.

Feel free to make changes to my examples, either by using screenshots (of any platform), or any other changes. Do not worry about violating my privacy. The prisons and newspapers do not worry about what they say, or even if it is true or not. I believe you would only deal with truth, and that is fine with me. If I do not like a truth about myself, I try to change myself to someone I do like. Dr. Haas knows much about me, and anything he does not know, I will try to answer. Do you have a list of questions you would like to ask about me, or just about the paper?

P.2

The story behind the paper is both short and long. The short version is I love math and programming, Dr. Haas encouraged me to share the functions I wrote to do my research, and the end result is the manuscript you received.

A more complete, and longer, version stretches back to the 1990s. Before my father (an electrical engineer) died, he got me hooked on trying to trisect an angle using only a straight edge and compass. This resulted in me learning more about angles than my school taught. But about 2 years after he died, I was arrested for murder in 1997. Shortly after I finally was transfered to prison in 1999, I enrolled in an Information Systems program there that was offered by W.P.C.C. Once again I went down the rabbit-hole of self-directed extra-curricular learning after one of the assignments was to write a program that would output all primes $\leq$ a user input value. The program was simple enough to write, but I did not like how inefficient even the best program we had was. I had found patterns that allowed me to only test a small portion of numbers in the needed range, but at the same time I kept reading how primes were distributed randomly. This disconnect between data and my (very limited) research materials led me to learn and question. One of my teachers finally told me to put my pattern into an equation... So I did, but a variable in it was giving me trouble. I

P. 3

could not predict its value, only its range. Its value had to be found by trial and error.

Time passed and I came across an article by Dr. Haas that contained his address. He was kind enough to respond to my letter (many people throw away letters unopened when they see the "mailed from ___ correctional facility"). He helped me standardize my notations and gave me good news/bad news. The good news was that there was nothing wrong with my math, the bad news was my problem variable's value was an open problem: how to best find the exponent of A Mod B.

I thanked him for his help, and started researching again, mainly by brute force, rather than reading... A few more years passed and a friend in here was released from prison and kept in touch. He was willing to order books and stuff for me as long as I sent him the money to pay for it. But since information is limited in here, I didn't know what books I should try to get.

This led me to contact Dr. Haas again for advice. His book recommendations were always on target, and he could always find something that addressed my questions. His advice and encouragement always pushed me to go a little further. He became my mentor.

When I signed up for a calculus class through U.N.C-C.H. (I was finally allowed...), prison staff had to allow me to receive a calculator. I sent money to my

P.4

friend who helped me order books, and asked him to order me the best calculator he could find that had a paper manual. After all, manuals on the internet or CDs would not help me... And then I received my TI-89! It was wonderful to have a mini computer to use in my research studies! No longer would I have to hope I could sign up for a computer class for a few months (every few years) and write macros in VBA to generate data for me to study. All I had to do was spend a few minutes, or hours, writting a TI-BASIC function for the TI-89 and let it run.

Of course, with my heavy use beyond what the average user does, it crashed a few times... which is how the precautions on P.2 of my paper made themselves known to me.

Dr. Haas printed out several articles from your DERIVE newsletter for me, and encouraged me to share my functions with other users. Dr. Haas devoted many hours to typing up the functions and the article. He was very patient as it underwent numerous revisions and expansions. At one point it seemed like the TI-89 would crash between each of our letters, and in retyping the functions into the TI-89, I would find several improvements that made the code better, which led to more revisions. And of course, as I wrote functions to aid my research, I added them. I probably could have added enough more for a Part 3...

P. 5

While all this was going on, I was accepted as a student for this prison's 3-year UNCA program which will allow me to earn half the credits needed for a bachelors degree. A contributing factor in my acceptance was a copy of my manuscript in process. And our first semester helped me improve my writing with an academic research and writing class.

Right now we are on hold because the virus has been found at this prison. Since we do not know when we can have classes again, I am thinking about how to take classes by mail from other colleges, but I am not sure if I can get scholarship money to pay for them.

Thankfully I will be done with prison mid 2022, and can then get a job to work my way through college. I will be 42 years old. I have been locked up since the first day of summer break after 10TH grade when I was 17. But I have not wasted the time I have spent in prison, I have used it.

Sincerely,

Matthew A. Myers
0551079 A.M.C.I.

Part of one original page of Matthew's code, which was typed by Prof. Haas for Matthew:

```
:DIV_SUM(CA)
:FUNC©RETURNS THE SUM OF ALL CA`S DIVISORS
:LOCAL VA,VB,VC,VD,VE,VF,VG,VH,VI,VJ,VK,VL
:F_2IL(CA)→VA
:IF GETTYPE(VA)="STR":RETURN VA
:DIM(VA)→VD:{}→VK
:FOR VF,1,VD
:STANDARD(MID(VA,VF,1))→VC
:COLDIM(VC)→VE:{1}→VB
:FOR VG,1,VE:{}→VL
```

TI-89 Keystroke Functions, Part One

by
Matthew A. Myers, U.S.A.

Smallest, fastest, newest, best... these are qualities most people look for in electronics. But "best" can vary by person. While the newest calculators tend to only make their user manuals available as a digital copy on a website, or on a CD, the original TI-89 has a paperback user's manual. A user may gain full knowledge and use of its abilities in the many places in the world that do not have computers or the internet. This prevents the TI-89 from becoming obsolete, even in today's increasingly internet-only world. Another TI-89 feature is "Keystroke Functions," which are installable on a TI-89 just by typing them into the calculator, no downloads required.

The 29 keystroke functions shared in this two-part series are user-friendly, and perform similar in fashion to factory-standard TI-89 functions. Functions accept a range of input types, and output matches the first input data type as appropriate. Argument matching is improved (for data types and dimensions), and functions simplify input to base type if input data types are "nested" (a number, as a text string, inside a single item list, can be formatted to a number). Input errors generate an error message string (STR), or a TI-89 error of appropriate type (see *TI-89 Guidebook* pages 516-521 for information on Syntax, Command, and other errors).

The functions are divided into two parts, and four categories:

Part 1		**Part 2**		
1 Utilities		**2 Math**	**3 Number Theory**	**4 Congruences**
f_2i()	**inlist**()	**div_c**()	**carmiclλ**()	**exp_any**()
f_2il()	**inlist_a**()	**divisors**()	**eulerφ**()	**exp_m**()
f_2l()	**s_a_list**()	**ord_p**()	**liouvilλ**()	**exp_rrs**()
f_2n()	**s_a_mat**()	**pow_mod**()	**mangoldt**()	**pr_class**()
f_2nl()	**s_a_set**()	**syn_div**()	**möbius**()	**pr_min**()
f_2s()	**standard**()			**r_r_sys**()
f_2set()				

Utilities are addressed in Part 1 of this series, as they are used by later functions, and are the most universally useful to programmers. The seven functions that start with "**f_2**" handle "formatting to" with few restrictions on input format. The function **standard()** shares purposes with the TI-89 function factor(), but puts the information into a matrix available for computation. Other basic tasks covered are sorting data (the three functions starting with "**s_a_**"), and positional information of items in a list (**inlist**() and **inlist_a**() functions).

Part two applies these Utility functions for Math, Number Theory, and Congruence functions. The second category covers the Math functions, including informational tasks like counting (or listing) divisors of a number, finding a number's ordinality, computational tasks like finding the remainder of a number raised to a power, and synthetic division. Third is Number Theory, and these are specialized to the point that it is best to have prior knowledge of them or read the individual descriptions and textual references provided below. Fourth, and last, are Congruence functions (which is a field of number theory, and the average user may need background information from the references noted), that return the results of computations (**exp_any**(), **exp_m**(), and **exp_rrs**() functions), informational data (**pr_class**() and **pr_min**() functions), or sets of numbers (the **r_r_sys**() function) as needed. The author welcomes reader suggestions for future articles for other desired keystroke functions.

Function code text in Italic is for input error-handling, or comments, and experienced users may omit it. Text in **Highlighted Bold** are functions that must be in the same active folder, or have their full path name included in the code, for proper execution (see below). A list of all new functions (from this series) required for each keystroke function, is in their respective descriptions below. This should allow for the user to copy (or type) only the desired functions, and their required sub-functions, rather than all 29. The code for three functions (**f_2i**(), **f_2nl**(), and **s_a_list**() functions) was omitted to save space due to their similarity to others. Instructions to create the omitted functions are in their description entry. For purposes of space, many lines of programming code were condensed by fitting more than one instruction on a line. This is allowed by the TI-89, as each colon signals the start of a new command line. When two or more colons are on a line, it may be split into more lines for clarity.

The following abbreviations and definitions are used in this series:

- ANY: Any data type that the user can input, and the calculator can manipulate.
- GEN: EXPR, LIST, MAT, NUM, or STR data types that can convert to a single NUM item.
- GENs: As GEN, but with the option of more than one NUM item.
- INT: As GEN, but convertible to a single Integer NUM item without rounding.
- INTs: As INT, but with the option of more than one Integer item.
- Type Matched: Output data type matches the first Input argument data type, or is a LIST.

Because not everyone will want all 29 functions, a list of required functions is provided in each description below. For ease of usage, create all desired functions in one folder (MAIN is the default, but any will do). If the user may access these functions from a "current folder" other than the folder the functions are in, add the folder name to each instance of a function "called" by another function (which are **Bold** in this series, or **Highlighted Bold** in the code sections).

For example, **f_2il**() calls both **f_2l**() and **f_2n**(). If these functions are in a folder named "myers" then modify lines 5 and 9 to the following:

```
Line 5    :myers\f_2l(va)→vb:dim(vb)→vd:{}→ve
Line 9    : If getType(vc)≠"NUM":myers\f_2i(vc)→vc
```

Use the TI-89's search function to find each instance that needs updating inside all functions typed from this series. Note that if a function has a remark that it requires another function, it may not directly call upon that function itself, rather, one of the functions it calls may use it.

To prevent data loss due to the calculator crashing, here are some precautions:

- Be aware of how much available memory remains (less is available inside an application).
- When possible, compute data outside of the Data/Matrix application, then copy and paste.
- Change batteries by taking all the batteries out first, and then replacing all of them.
- Do not "garbage collect" when the batteries are low, replace the batteries first.
- If in the middle of a computation, the "very-low battery" indicator is showing, wait until the screen turns off. Once the screen is off, change the batteries, and continue.

Examples presented for each function have all single letter variables empty (or undeclared) unless noted, and the following settings (in "MODE"): "Exact/Approx" set to "AUTO", "Angle" set to "RADIAN", and "Display Digits" set to "FLOAT". After using a variable for a given example, it should be deleted to re-empty it before trying the next example.

1 Utility Functions

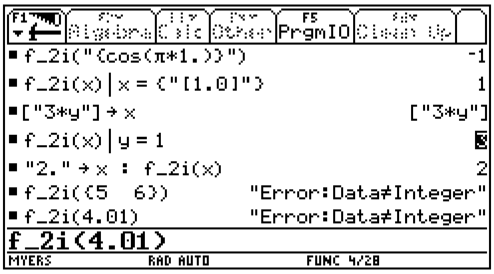
f_2i(A)	**f_2nl**(A)	**inlist_a**(A,B,C)	**standard**(A)
f_2il(A)	**f_2s**(A)	**s_a_list**(A)	
f_2l(A)	**f_2set**(A)	**s_a_mat**(A,B)	
f_2n(A)	**inlist**(A,B,C)	**s_a_set**(A)	

These functions reformat input as desired (the seven functions starting with "**f_2**" and the **standard**() function), provide information about input (**inlist** and **inlist_a** functions), or sort data in ascending order (the three functions that start with " **s_a_**"). Their primary usefulness is not at the MAIN command line as user-level functions, but as tools used in other functions, as demonstrated in following sections (often handling user input). Interested readers are encouraged to review Texas Instrument's Chapter 17: Programming, and Appendix A: TI-89 Functions and Instructions. With proper techniques, and when the data is known to be formatted in specific ways, sorting functions like **s_a_list(), s_a_mat(),** and **s_a_set()** can be sped-up. Unfortunately, the built-in functions SortA and SortD may not be used on Local function variables.

f_2i(INT)

Returns a single integer formatted to datatype NUM from the given input, but will not round to the nearest integer. Useful when the format of an integer being supplied by a function or user may vary.

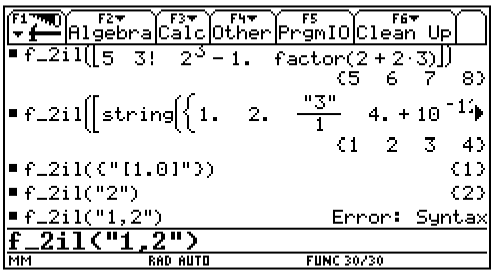
Examples:

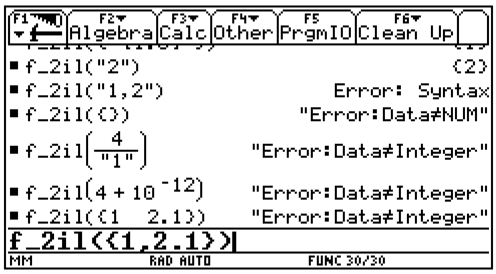


f_2il(INTs)

Returns an integer LIST datatype from the given input, but will not round to the nearest integer. Useful when the format of an integer being supplied by a function or user may vary, or to reformat integers for sequential processing in a function.

Examples:





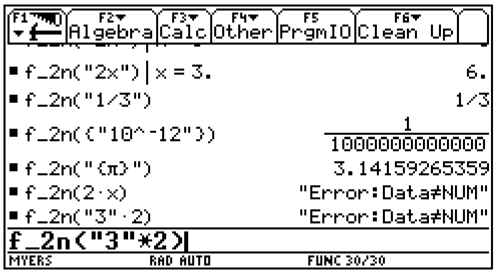
f_2n(GEN)

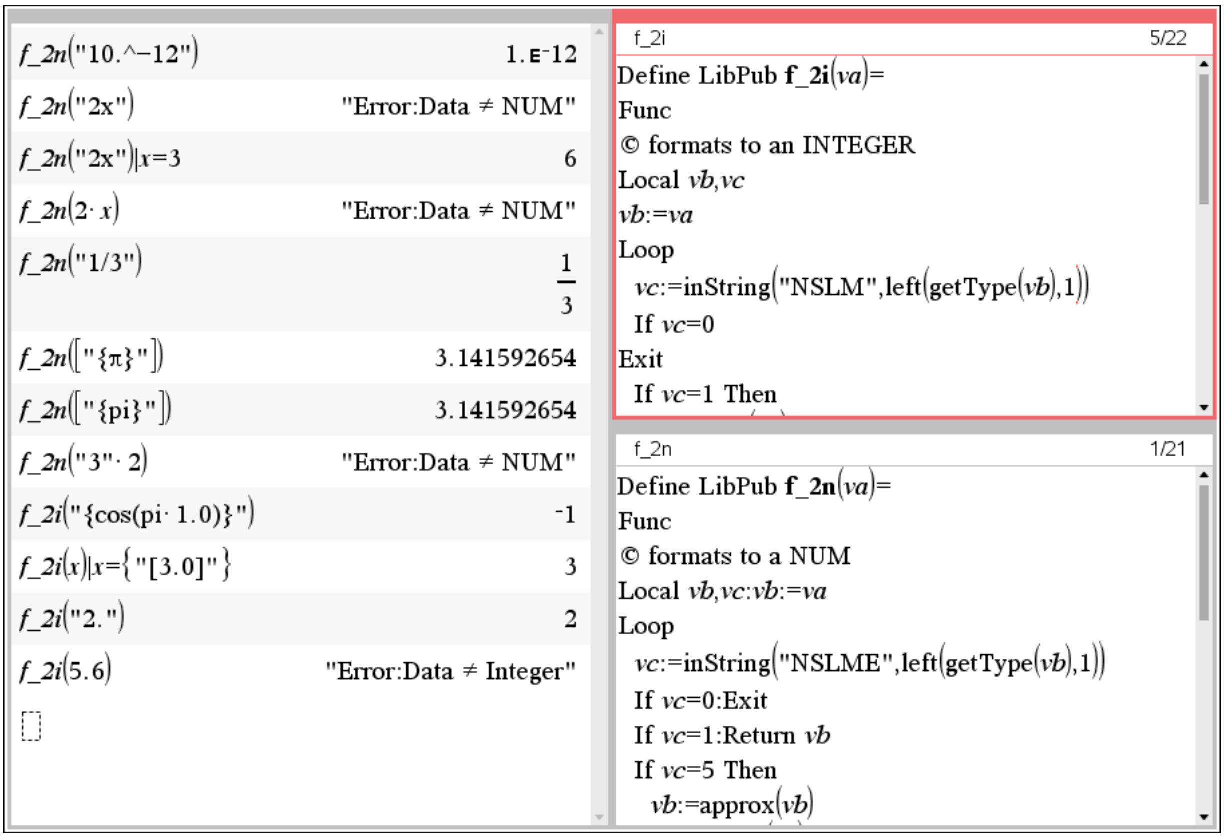
Returns a NUM.

In this form all function codes are contained in the contribution typed by Dr. Haas. The TI-89, TI-92 and V200-files are part of MTH119.zip. One or the other function are ported on TI-Nspire-platform, too.

```
:f_2n(va)
:Func©formats to a NUM©line 2
:Local vb,vc:va→vb
:Loop
:  inString("NSLME",left(getType(vb),1))→vc©line 5
:  If vc=0:Exit
:  If vc=1:Return vb©line 7
:  If vc=5 Then
:    approx(vb)→vb
:    If getType(vb)="EXPR":Exit
:    Cycle
:  EndIf©line 12
:  If vc=2 Then
:    when(vb≠"",expr(vb),∞)→vb
:  ElseIf vc=3 Then
:    when(dim(vb)=1,sum(vb),∞)→vb
:  ElseIf dim(vb)={1,1} Then
:    sum(mat▸list(vb))→vb
:  Else:Exit
:  EndIf
:EndLoop
:Return "Error:Data≠NUM"©line 22
:EndFunc
```

Examples:

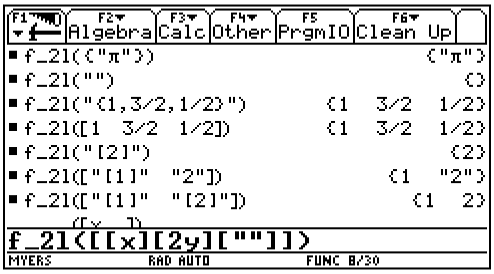


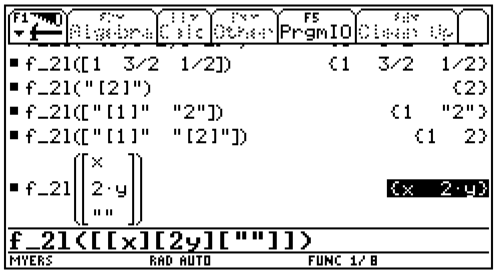


f_2l(ANY)

Returns a LIST, never an error STR, and will "un-nest" to strip away other datatype formatting when feasible. To prevent errors, it will sometimes not strip away STR formats.

Examples:

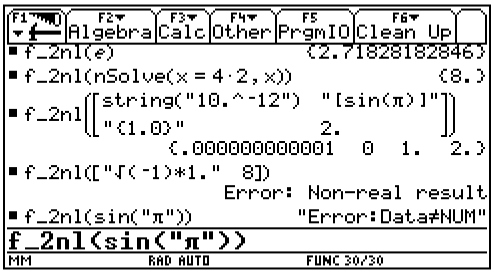


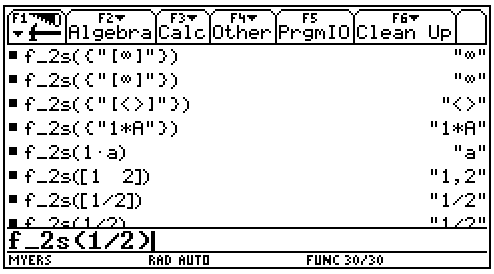


f_2nl(GENs) and f_2s(ANY)

Returns a NUM LIST and returns a STR, never an error message. Note that it may simplify more than desired.

Examples:

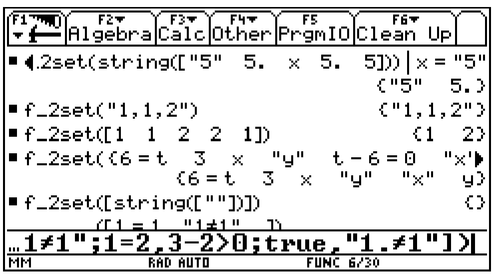


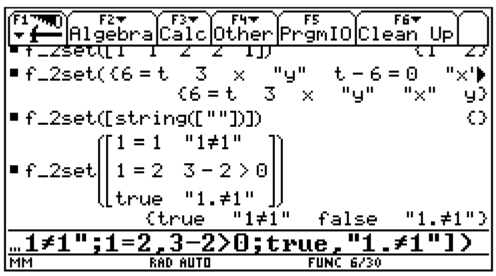


f_2set(ANY)

Returns an unsorted reduced set (no duplicates) LIST, never an error STR. When the input is known to be sorted (ascending) before being passed to **f_2set**(), changing **inlist**() to **inlist_a**() on line 9 speeds things up. See also **s_a_set**() description.

Examples:



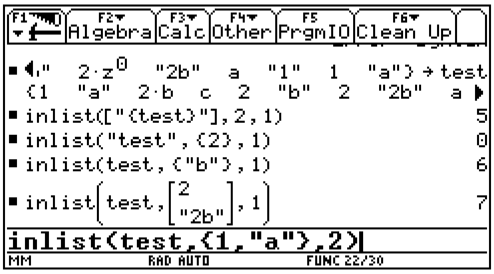


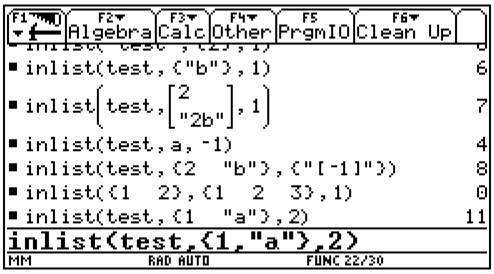
inlist(ANY_A, ANY_B, INT_C)

Returns as **inlist**() above, except this function assumes (but does not verify) search_LIST_A is in ascending order. When search_LIST_A is not sorted, it may give incorrect, or useless, output as a result, see the last 4 examples on the right. This is due to how the function searches a list, comparing data types and values, to see if it should give up before searching the whole list.

Examples: (make sure single letter variables are empty before trying these)

With {1,"a",2*b,c,2,"b",2z^0,"2b",a,"1",1,"a"}→test



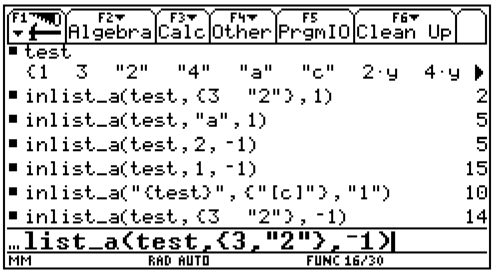


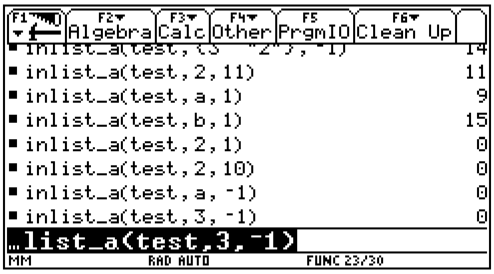
inlist_a(ANY_A, ANY_B, INT_C)

Returns as **inlist**() above, except this function assumes (but does not verify) search_LIST_A is in ascending order. When search_LIST_A is not sorted, it may give incorrect, or useless, output as a result, see the last 4 examples. This is due to how the function searches a list, comparing data types and values, to see if it should give up before searching the whole list.

Examples:

With {1,3,"2","4","a","c",2y,4y,a,c,2,"3","b",3y,b}→test

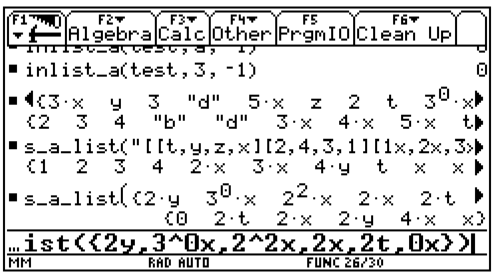




s_a_list(ANY)

Returns LIST items of type EXPR, NUM, STR, &VAR as a sorted ascending LIST. Data types are sorted in the following order: NUM, STR, EXPR, VAR. An EXPR may be simplified, and if one simplifies to a VAR (i.e. 1*x to x), it will be sorted as a VAR. Otherwise, list items are not simplified to a base type if "nested", but the input is converted to a LIST. This function is very slow compared to the built-in SortA function, and should only be used where SortA cannot be.

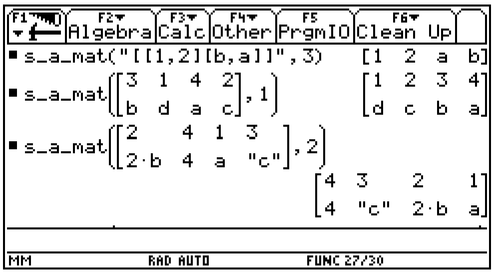
Examples: (make sure single letter variables are empty before trying these):

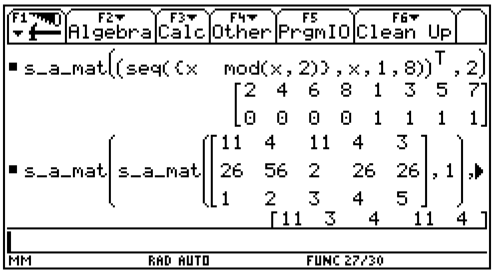


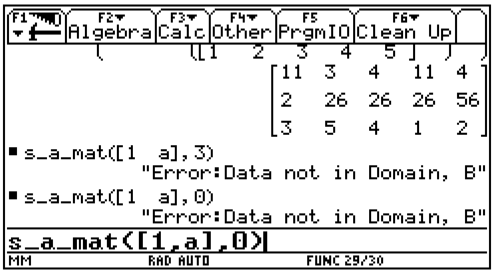
s_a_mat(ANY_A, INT_B)

Returns matrix_ANY_A as a sorted ascending (by row_INT_B) MAT. It follows the same data type sorting rules as **s_a_list**() above. When matrix_ANY_A is not a MAT then it is passed to **s_a_list**() for sorting, disregarding nested data types and row_INT_B input. The built-in TI-89 SortA function allows a matrix's rows to be separated into lists, sorted, and recombined into a sorted MAT. But, as mentioned above, SortA is not available for use inside functions, and **s_a_mat**() may be nested (as in the last example presented here) without "forgetting" the order of previous sorting (as with SortA). This is an advantage over SortA, aside from being able to be used in places where SortA cannot be. If results would be the same from SortA and **s_a_mat**() (like for sorting the first row), SortA should be used as it is faster.

Examples:



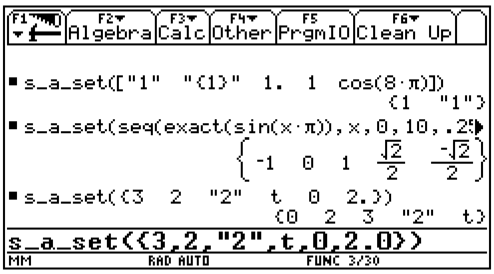




s_a_set(ANY)

Returns a sorted ascending "reduced set" (no duplicates) LIST following the same sorting rules as **s_a_list**() above. While **s_a_set**() is faster than using **s_a_list**() and then **f_2set**(), it is slower than first using the built-in TI-89 SortA function and then **f_2set**(), when using SortA is possible (see the above **f_2set**() description for more details).

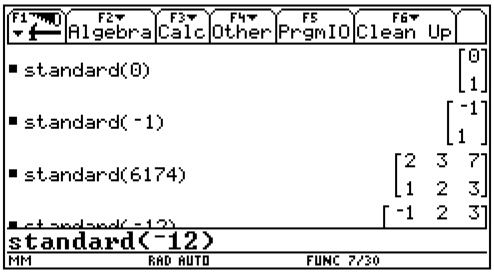
Examples:

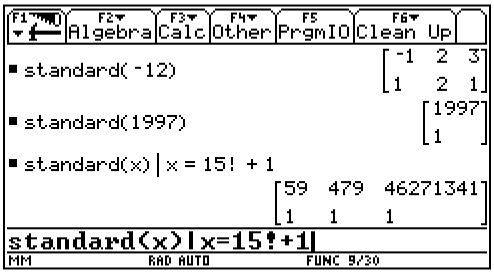


standard(INT)

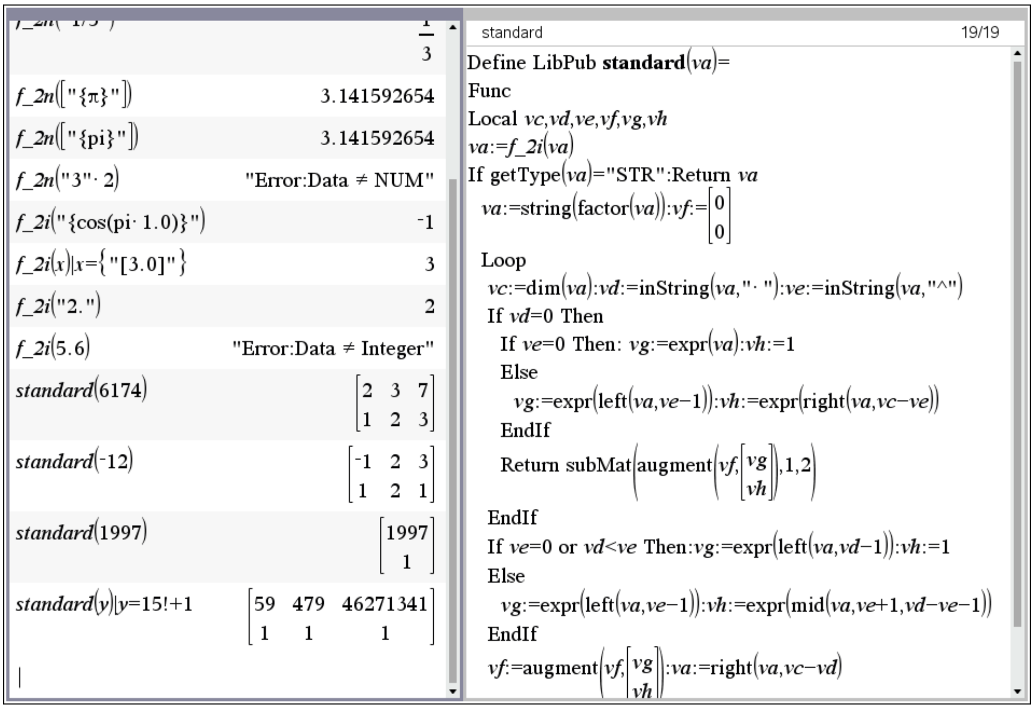
Returns INT in Standard form as a 2 row MAT. Row 1 shows the factors, Row 2 is their powers. This changes the temporary format of the TI-89 built-in function factor() (which is an EXPR that evaluates to a NUM upon manipulation, unless first converted to a STR data type that is not readily computable) into a fixed format usable for computation and comparison of factors and powers.

Examples:





standard(*n*) as a TI-Nspire-function:



```
f_2n({"{π}"})                 3.141592654
f_2n({"{pi}"})                3.141592654
f_2n("3"·2)                   "Error:Data ≠ NUM"
f_2i("{cos(pi·1.0)}")         -1
f_2i(x)|x={"[3.0]"}           3
f_2i("2.")                    2
f_2i(5.6)                     "Error:Data ≠ Integer"
standard(6174)                [2 3 7; 1 2 3]
standard(-12)                 [-1 2 3; 1 2 1]
standard(1997)                [1997; 1]
standard(y)|y=15!+1           [59 479 46271341; 1 1 1]
```

```
standard                                         19/19
Define LibPub standard(va)=
Func
Local vc,vd,ve,vf,vg,vh
va:=f_2i(va)
If getType(va)="STR":Return va
 va:=string(factor(va)):vf:=[0;0]
 Loop
  vc:=dim(va):vd:=inString(va,"· "):ve:=inString(va,"^")
  If vd=0 Then
   If ve=0 Then: vg:=expr(va):vh:=1
   Else
    vg:=expr(left(va,ve-1)):vh:=expr(right(va,vc-ve))
   EndIf
   Return subMat(augment(vf,[vg;vh]),1,2)
  EndIf
  If ve=0 or vd<ve Then:vg:=expr(left(va,vd-1)):vh:=1
  Else
   vg:=expr(left(va,ve-1)):vh:=expr(mid(va,ve+1,vd-ve-1))
  EndIf
  vf:=augment(vf,[vg;vh]):va:=right(va,vc-vd)
```

In DERIVE function `factors` does the job, but presents the result originally as a 2-column matrix.

$$\text{FACTORS}(-12)' = \begin{bmatrix} -1 & 2 & 3 \\ 1 & 2 & 1 \end{bmatrix}$$

$$\text{FACTORS}(15! + 1)' = \begin{bmatrix} 59 & 479 & 46271341 \\ 1 & 1 & 1 \end{bmatrix}$$

Bibliography

Texas Instruments. *TI-89 Guidebook*. Harrisonburg, VA: Banta, 1998.

This series is dedicated to those who have no access to libraries or computers, and to those that kindly help them get books, copies, printouts, advice, and new ideas.

Special Thanks to
Jeffrey Bosch, Richard Cadwalader, Dr. Robert Haas, and UNCA staff