THE DERIVE - NEWSLETTER #94/95

ISSN 1990-7079

THE BULLETIN OF THE

DERIVE®

USER GROUP

+ CAS-TI

C o n t e n t s:

September 2014

All my (Josef's) bk-teachware booklets can be downloaded for free from the ACDCA-website: http://rfdz.ph-noe.ac.at/acdca/materialien.html.

acdca

Materialien für den technologiegestützten Unterricht!

Freigegebene Skripten aus der Schriftenreihe von bk teachware von Josef Böhm! (Juli 2014)

- SR-06: Optimierungsaufgaben grafisch, numerisch und analytisch mit dem TI-92 lösen
- SR-13: Einführung des Integralbegriffs mit den TI-CAS-Rechnern
- SR-32: Programmieren mit DERIVE
- SR-33: Neue Aufgaben für das Unterrichten mit Derive & TI-89/92 /92+/Voyage200 Band 1
- SR-34: Neue Aufgaben für das Unterrichten mit Derive & TI-89/92 /92+/Voyage200 Band 2
- SR-57: Forensische Mathematik für den Unterricht
- SR-60: Programmieren mit TI-Nspire-CAS

Intuition und Zufall

Mathematische Experimente mit Tabellenkalkulation

Benno Grabinger, http://www.bennograbinger.de

Dieses Buch ist mit iBooks auf Ihrem Mac oder iPad und auf Ihrem Computer mit iTunes zum Download verfügbar. Multi-Touch-Bücher können mit iBooks auf Ihrem Mac oder iPad gelesen werden. Interaktive Features funktionieren u. U. am besten auf einem iPad. Für iBooks auf dem Mac ist OS X 10.9 oder neuer erforderlich.

Dieses interaktive Buch bietet eine Sammlung von Beispielen, bei denen die Intuition etwas anderes suggeriert, als die mathematische Theorie liefert.

Dear DUG Members,

Welcome to this double issue of our Newsletter. It was a busy summer and it is still a busy fall. TIME 2014 has passed and many of the delegates are expecting the proceedings. I am already finished with urging for the full papers and bringing many of them in the right format. The proceedings will be published on the website of the Pedagogical University of Lower Austria as special issue of its electronic journal. We expect that it will be ready for download by the second half of October. I will send an extra info-email.

I sad message came in during summer. Bert Waits one of the most important propagators of technology supported passed away in July. Many of us have best memories on Bert. I remember that he once visited my class in St. Pölten and attended a TI-92 lesson. You may imagine how motivating a visit of a famous American mathematician for my students – and for me, too, of course was. Many thanks Bert, for all what you did for the CAS-community (more on page 54).

The main part of this DNL consists of the User Forum. There were so many requests and answers, a couple of comments to earlier contributions. Some requests are still open and are waiting for advice ☺.

Besides that we have a few original contributions:

I wanted to settle an old debt – yes, I know there are so many debts finding on the list of articles to be published – by including Günter Schödl's "Caesar Multiplication" (page 40). It is a variation of the well known Caesar encryption and it is a secure method at all but I find it could be a nice application of modular arithmetic combined with string manipulations for students.

What concerns contributions from former times. Among my many papers I found a DERIVE file treating "GALERKIN'S METHOD". It is from DOS-times and it took me some time to adapt it for the latest DERIVE version. My problem is that I didn't find an appropriate "Galerkin-Method" in the web which seems to correspond with this file. It would be great if anybody could give some advice about GM.

I had a very intense email contact and files-exchange with our Swiss DUG-Member Alfred Roulier. Preparing my talk for Krems I tried programming the TI-Nspire with LUA. Alfred had contributed for the TI-News about LUA and I asked for some advice. Together we created very pretty graphs (You can find my first LUA-attempts with assistance from Steve Arnold in the revised DNL#32.)

Alfred sent a wonderful article about Julia-sets and how to bring them to a colourful life on the Nspire-screen – with LUA. There is a great LUA-script on the Flemish T3-website (in Dutch) and a lot of LUA-materials on Steve Arnold's website. I recommend all Nspire-programmers to have a look.

http://www.t3vlaanderen.be/fileadmin/t3-be/cahiers/cahier_35.pdf

and

http://compasstech.com.au/TNS_Authoring/Scripting/index.html

Finally I could not resist to add one of my CAS-solved Brain Teasers. I gave a workshop on this issue at TIME 2014 and we had great 90 minutes together.

With best wishes and regards until DNL#96 (then only 4 issues remaining until DNL#100 !!).

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 3136365
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 2014

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
Tools for 3D-Problems, P. Lüke-Rosendahl, GER
Hill-Encryption, J. Böhm, AUT
Simulating a Graphing Calculator in *DERIVE*, J. Böhm, AUT
Do you know this? Cabri & CAS on PC and Handheld, W. Wegscheider, AUT
An Interesting Problem with a Triangle, Steiner Point, P. Lüke-Rosendahl, GER
Graphics World, Currency Change, P. Charland, CAN
Cubics, Quartics – Interesting features, T. Koller & J. Böhm, AUT
Logos of Companies as an Inspiration for Math Teaching
Exciting Surfaces in the FAZ / Pierre Charland´s Graphics Gallery
BooleanPlots.mth, P. Schofield, UK
Old traditional examples for a CAS – what´s new? J. Böhm, AUT
Truth Tables on the TI, M. R. Phillips, USA
Where oh Where is It? (GPS with CAS), C. & P. Leinbach, USA
Embroidery Patterns, H. Ludwig, GER
Mandelbrot and Newton with *DERIVE*, Roman Hašek, CZK
Tutorials for the NSpireCAS, G. Herweyers, BEL
Some Projects with Students, R. Schröder, GER
Dirac Algebra, Clifford Algebra, D. R. Lunsford, USA
Treating Differential Equations (M. Beaudin, G. Piccard, Ch. Trottier), CAN
A New Approach to Taylor Series, D. Oertel, GER
Henon & Co; Find your very own Strange Attractor, J. Böhm, AUT
Rational Hooks, J. Lechner, AUT
Simulation of Dynamic Systems with various Tools, J. Böhm, AUT
Technical Problems solved with Secondary Maths, W. Alvermann, GER
Pickover's Mygalomorphs and Spiders, A. Roulier & J. Böhm, SUI/AUT

and others

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

A „Batch file“ for solving equations?

Fred J. Tydeman

I am having trouble figuring out how to code a batch file to be fed into *DERIVE*. I have a series of equations (the first three shown here).

```
InputMode := Word
 p3 = (3 + 2*p3)/6
 p4 = (6 + 8*p3 + 9*p4)/24
 p5 = (10 + 20*p3 + 45*p4 + 44*p5)/120
```

I would like Derive to solve for *p3*. Then using that value of *p3*, use it in the *p4* equation and solve for *p4*. Once *p3* and *p4* are known, use them to find *p5* (and so on). How do I code a *.mth file to do that?

I do not want to do manual soLve and Manage/Substitute actions from within Derive.

I would like to do

```
 Derive
   Transfer
     Load
       Derive (or Demo)
         my batch file
```

On March 31, 2014 8:41:50 PM Josef Böhm <nojo.boehm@pgv.at> wrote:

> Hi Fred,
>
> How do you expect the output of the solution, and the input of the equations, too, of course?
>
> Maybe that we then can find a satisfying answer.
>
> Best regards
> Josef

Fred J. Tydeman

Just need values of the j vars.

DNL:

Hi Fred, what about this:

Load `freds.mth` as a Utility file.

Then you can call function `fred(your system)` and you will receive the solution vector.

Examples:

`fred([2x+3y=2,4x-5z+y=10,2y-4z+u=0,x+y+z+u=10])=`

`fred_examples.dfw` presents your example together with another one.

You may enter as many equations as you like. The utility file switches *DERIVE* automatically into WORD-Mode.

Hope this is what you expected.

Best regards

Josef

`freds.mth`

```
#1:   InputMode := Word

      fred(system) := (SOLUTIONS(system, VARIABLES(system)))
#2:                                                          1
```

The examples:

```
#1:   LOAD(D:\DOKUS\DNL\DNL93\freds.mth)
```

Your example:

$$\#2:\quad \mathrm{fred}\left[p3 = \frac{3 + 2\cdot p3}{6},\ p4 = \frac{6 + 8\cdot p3 + 9\cdot p4}{24},\ p5 = \frac{10 + 20\cdot p3 + 45\cdot p4 + 44\cdot p5}{120}\right] = \left[\frac{3}{4}, \frac{4}{5}, \frac{61}{76}\right]$$

2nd example:

$$\#3:\quad \mathrm{given} := \left[p3 = \frac{3 + 2\cdot p3}{6},\ p4 = \frac{6 + 8\cdot p3 + 9\cdot p4}{24},\ p5 = \frac{10 + 20\cdot p3 + 45\cdot p4 + 44\cdot p5}{120},\ p6 = \frac{20 + 40\cdot p3 + 50\cdot p5 + 60\cdot p5 + 70\cdot p6}{150}\right]$$

$$\#4:\quad \mathrm{fred(given)} = \left[\frac{3}{4}, \frac{4}{5}, \frac{61}{76}, \frac{1051}{608}\right]$$

<u>Fred J. Tydeman</u>

On Wed, 30 Apr 2014 13:01:49 +0200 Josef Böhm wrote:
>
>some time ago I sent an idea how to solve your problem with the several >equations.
>I'd like to know if my "solution" was satisfying - I'd like to put your question - together with
>the possible solution in the next DERIVE Newsletter.

I finally found that email and attachments. They worked fine and that is what I was looking for.
Thanks.
Fred

Bug in Taylor Series?

Francisco M Fernandez, Argentina

Dear Derivians,

I am attaching a short dfw file that shows a problem with Taylor series. Am I missing anything?

Greetings

Francisco

$[r :\in \text{Real}\ (0, \infty),\ x0 :\in \text{Complex}]$

$$\text{TAYLOR}\left(x^2 + \frac{r^8}{3\cdot x^6},\ x,\ x0,\ 5\right)$$

$$-\frac{252\cdot r^8\cdot x^5 - 1386\cdot r^8\cdot x0\cdot x^4 + 3080\cdot r^8\cdot x0^2\cdot x^3 - 3\cdot x0^3\cdot x^2\cdot(1155\cdot r^8 + x0^8) + 1980\cdot r^8\cdot x0^4\cdot x - 462\cdot r^8\cdot x0^5}{3\cdot x0^{11}}$$

$$-\frac{84\cdot r^8\cdot x^5}{x0^{11}} + \frac{462\cdot r^8\cdot x^4}{x0^{10}} - \frac{3080\cdot r^8\cdot x^3}{3\cdot x0^9} + \frac{x^2\cdot(1155\cdot r^8 + x0^8)}{x0^8} - \frac{660\cdot r^8\cdot x}{x0^7} + \frac{154\cdot r^8}{x0^6}$$

$$\text{coe}(j) := \lim_{x\to x0} \frac{1}{j!}\cdot\left(\frac{d}{dx}\right)^j \left(x^2 + \frac{r^8}{3\cdot x^6}\right)$$

$$\text{coe}(1) = 2\cdot x0 - \frac{2\cdot r^8}{x0^7}$$

$$\text{coe}(2) = \frac{7\cdot r^8 + x0^8}{x0^8}$$

DNL: Dear Francisco,

I believe that you did not consider the powers of (x-x0). In the TAYLOR-result of *DERIVE* all products are simplified (expanded and added).

Only the summand of highest degree remains alone. (You get a lot x, x^2, x^3, ...) by evaluating all the (x-x0)^k.

Hope that I am right now.

Sorry for my silly rubbish which I sent earlier (which is not reprinted here).

Regards

Josef

$$\text{coecoe}(j) := \text{EXPAND}\left(\left(\lim_{x\to x0} \frac{1}{j!}\cdot\left(\frac{d}{dx}\right)^j \left(x^2 + \frac{r^8}{3\cdot x^6}\right)\right)\cdot(x - x0)^j\right)$$

$$\text{coecoe}(5) = \frac{84\cdot r^8\cdot(x0 - x)^5}{x0^{11}}$$

$$\left(\sum_{j=0}^{5} \text{coecoe}(j)\right) - \text{TAYLOR}\left(x^2 + \frac{r^8}{3\cdot x^6}, x, x0, 5\right) = 0$$

$$\text{POLY\_COEFF}(\text{coecoe}(5), x, 5) = - \frac{84\cdot r^8}{x0^{11}}$$

$$\text{POLY\_COEFF}\left(\text{TAYLOR}\left(x^2 + \frac{r^8}{3\cdot x^6}, x, x0, 5\right), x, 5\right) = - \frac{84\cdot r^8}{x0^{11}}$$

$$\text{coecoe}(j) := \text{EXPAND}\left(\left(\lim_{x\to x0} \frac{1}{j!}\cdot\left(\frac{d}{dx}\right)^j \left(x^2 + \frac{r^8}{3\cdot x^6}\right)\right)\cdot(x - x0)^j\right)$$

$$\text{coecoe}(5) = \frac{84\cdot r^8\cdot(x0 - x)^5}{x0^{11}}$$

$$\left(\sum_{j=0}^{5} \text{coecoe}(j)\right) - \text{TAYLOR}\left(x^2 + \frac{r^8}{3\cdot x^6}, x, x0, 5\right) = 0$$

$$\text{POLY\_COEFF}(\text{coecoe}(5), x, 5) = - \frac{84\cdot r^8}{x0^{11}}$$

$$\text{VECTOR}\left(\text{POLY\_COEFF}\left(\sum_{j=0}^{5} \text{coecoe}(j), x, i\right), i, 0, 5\right)$$

$$\left[\frac{154\cdot r^8}{x0^6}, -\frac{660\cdot r^8}{x0^7}, \frac{1155\cdot r^8 + x0^8}{x0^8}, -\frac{3080\cdot r^8}{3\cdot x0^9}, \frac{462\cdot r^8}{x0^{10}}, -\frac{84\cdot r^8}{x0^{11}}\right]$$

Dear Josef,

thank you very much. I apologize for my foolishness. I forgot to change the variable back to s = x – x0 in order to compare both expansions.

Francisco

Question for a Proof

David Halprin, Australia

Has any reader seen reference to, or use of, the "Conditions for Immobility of a Point" and/or "The Conditions for Immobility of a Straight Line", (as derived and used by Ernesto Cesaro), appearing anywhere other than in Cesaro's own book, "Lectures in Intrinsic (Natural) Geometry" and in some of his papers?

Especially, does any reader know of a better, (more rigorous, yet simpler in derivation), proof for these conditions? They appear to be a very useful mathematical tools, no longer in use, but with much untapped potential in geometry, differential geometry and calculus of variations, at least.

Making Matrices

Francisco Marcelo Fernández, Argentina

Dear Derivians,

I am attaching a short dfw file that shows a problem which I found when building unitary matrices. Is there any way to overcome the problem with `set_mat(v0,v1)`?

Greetings

Francisco

Makes the matrix representation m of the vector transformation v =m · u

```
#1:   mm(u, v) := VECTOR(VECTOR(∂(v_i, u_j), i, 1, DIM(v)), j, 1, DIM(u))
```

Two permutation operations of the elements of a vector v

```
#2:   P(i, j, v) := SWAP_ELEMENTS(v, i, j)

#3:   PR(v) := [v_3, v_1, v_2]
```

Builds the matrix representation for the transformation between v0 and all the permutations of v1

```
#4:   set_mat(v0, v1) := [mm(v0, v1), mm(v0, PR(v1)), mm(v0, PR(PR(v1))),
        mm(v0, P(1, 2, v1)), mm(v0, PR(P(1, 2, v1))), mm(v0, PR(PR(P(1, 2,
        v1))))]
```

Defines the order of a group element in its matrix representation

```
      order(m, j) :=
        Prog
          j := 0
#5:       Loop
            j :+ 1
            If m^j = IDENTITY_MATRIX(3) exit
          RETURN j
```

Shows the properties of the matrices in vm

```
#6:   test(vm) := VECTOR([DET(vm_i), TRACE(vm_i), order(vm_i)], i, 1, DIM(vm))`
```

The following example shows that set_mat(v0,v1) yields the correct answers for all the matrices except the fifth one.

#7: set_mat([x, y, z], [x, y, z])

#8: $$\left[\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}, \begin{bmatrix}0&1&0\\0&0&1\\1&0&0\end{bmatrix}, \begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}, \begin{bmatrix}0&1&0\\1&0&0\\0&0&1\end{bmatrix}, \begin{bmatrix}0&1&0\\0&0&1\\1&0&0\end{bmatrix}, \begin{bmatrix}1&0&0\\0&0&1\\0&1&0\end{bmatrix}\right]$$

Direct calculation reveals the error:

#9: [mm([x, y, z], [x, y, z]), mm([x, y, z], PR([x, y, z])), mm([x, y, z], PR(PR([x, y, z]))), mm([x, y, z], P(1, 2, [x, y, z])), mm([x, y, z], PR(P(1, 2, [x, y, z]))), mm([x, y, z], PR(PR(P(1, 2, [x, y, z]))))]

#10: $$\left[\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}, \begin{bmatrix}0&1&0\\0&0&1\\1&0&0\end{bmatrix}, \begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}, \begin{bmatrix}0&1&0\\1&0&0\\0&0&1\end{bmatrix}, \begin{bmatrix}0&0&1\\0&1&0\\1&0&0\end{bmatrix}, \begin{bmatrix}1&0&0\\0&0&1\\0&1&0\end{bmatrix}\right]$$

#11: set_mat([x, y, z], [x, y, z])

#12: $$\left[\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}, \begin{bmatrix}0&1&0\\0&0&1\\1&0&0\end{bmatrix}, \begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix}, \begin{bmatrix}0&1&0\\1&0&0\\0&0&1\end{bmatrix}, \begin{bmatrix}0&0&1\\0&1&0\\1&0&0\end{bmatrix}, \begin{bmatrix}1&0&0\\0&0&1\\0&1&0\end{bmatrix}\right]$$

#13: VECTOR([x, y, z]·M, M, set_mat([x, y, z], [x, y, z])) = $$\begin{bmatrix}x&y&z\\z&x&y\\y&z&x\\y&x&z\\z&y&x\\x&z&y\end{bmatrix}$$

#14: test(set_mat([x, y, z], [x, y, z])) = $$\begin{bmatrix}1&1&1&-1&-1&-1\\3&0&0&1&1&1\\1&3&3&2&2&2\end{bmatrix}$$

DNL: Dear Francisco,

here is a late reply to your request from April.

At the occasion of collecting the messages for the DNL#94 User Forum a inspected your matrix problem again.

In the file mentioned above I didn't find any problem.

Simplifying **set_math([x,y,z],[x,y,z])** gave the correct result.

As I needed the permutations for my TIME 2014 lecture I used the respective function and could find another way to generate the group of matrices.

I attach the DERIVE file.

Function **perm(v,k)** must be preloaded (for this function see page 15).

```
mm(u, v) := VECTOR(VECTOR(∂(v_i, u_j), i, 1, DIM(v)), j, 1, DIM(u))

set_math(v0, v1) := VECTOR(mm(v0, (perm(v1, DIM(v1)))_i), i, DIM(v1)!)

set_math([x, y, z], [x, y, z])
```

$$\left[\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix},\begin{bmatrix}0&1&0\\1&0&0\\0&0&1\end{bmatrix},\begin{bmatrix}1&0&0\\0&0&1\\0&1&0\end{bmatrix},\begin{bmatrix}0&0&1\\1&0&0\\0&1&0\end{bmatrix},\begin{bmatrix}0&1&0\\0&0&1\\1&0&0\end{bmatrix},\begin{bmatrix}0&0&1\\0&1&0\\1&0&0\end{bmatrix}\right]$$

```
set_math([x, y, z], [2·y, 3·z, x])
```

$$\left[\begin{bmatrix}0&0&1\\2&0&0\\0&3&0\end{bmatrix},\begin{bmatrix}0&0&1\\0&2&0\\3&0&0\end{bmatrix},\begin{bmatrix}0&1&0\\2&0&0\\0&0&3\end{bmatrix},\begin{bmatrix}0&1&0\\0&0&2\\3&0&0\end{bmatrix},\begin{bmatrix}1&0&0\\0&2&0\\0&0&3\end{bmatrix},\begin{bmatrix}1&0&0\\0&0&2\\0&3&0\end{bmatrix}\right]$$

$$\mathrm{VECTOR}\left([x,\ y,\ z]\cdot\left[\begin{bmatrix}0&0&1\\2&0&0\\0&3&0\end{bmatrix},\begin{bmatrix}0&0&1\\0&2&0\\3&0&0\end{bmatrix},\begin{bmatrix}0&1&0\\2&0&0\\0&0&3\end{bmatrix},\begin{bmatrix}0&1&0\\0&0&2\\3&0&0\end{bmatrix},\begin{bmatrix}1&0&0\\0&2&0\\0&0&3\end{bmatrix},\begin{bmatrix}1&0&0\\0&0&2\\0&3&0\end{bmatrix}\right]_i,\ i,\ 6\right)$$

$$\begin{bmatrix}2\cdot y&3\cdot z&x\\3\cdot z&2\cdot y&x\\2\cdot y&x&3\cdot z\\3\cdot z&x&2\cdot y\\x&2\cdot y&3\cdot z\\x&3\cdot z&2\cdot y\end{bmatrix}$$

Dear Joseph:

You are right, `set_math` gave me the right answer. Probably I did something when converting it from DFW5 to DFW6 (I am rather too fond of the former).

I am attaching an annotated DFW5 file with my attempts to construct the unitary transformations that leave a given polynomial invariant. This is just the 2D case and illustrates the problem by means of three simple examples. I hope it is clear enough. I did several others, but most of them partly on paper and partly on Derive. I do not have enough patience for programming properly. I am not a good Derivian, even though I like Derive very much.

I am also interested in greater dimensions and have obtained results for some cases also. I think that there is some mathematical bibliography on the more general problem of linear transformations, but I am mainly interested in group theory and, therefore, restricted myself to unitary transformations.

Best regards,

Francisco

Francisco Marcelo Fernández, Argentina

Dear Derivians,

I am attaching a short pdf file that shows what I think is an interesting problem. I have been doing such calculations with Derive step by step with a variety of strategies but I would like to have a program that does them automatically. Did anybody try something like it before? Or, is anybody willing to program it efficiently?

Greetings

Francisco

Given a polynomial function $P(x_1, x_2, \ldots, x_n)$ I am interested in finding all the unitary matrices $\mathbf{U}$ such that $P(x_1, x_2, \ldots, x_n) = P(x'_1, x'_2, \ldots, x'_n)$ where

$$\mathbf{U}\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} x'_1 \\ x'_2 \\ \vdots \\ x'_n \end{pmatrix} \quad (1)$$

Those matrices form a group. I did it painfully by hand for some examples with $n = 2$ and $n = 3$ but I would like having a Derive program that does it automatically. Does anybody do something like this?

Problem with TI-Nspire`s List & Spreadsheet

Robert Märki, Switzerland

Bei der Vorbereitung der t3-Regionaltagung in Bern bin ich auf ein Problem gestoßen.

Ich habe eine Funktion „cooling“ definiert für die Berechnung der Temperatur mit der Euler-Methode. Für das Studium der Konvergenz brauche ich die Werte cooling(10,1), cooling(10,0.1) etc. Die direkte Berechnung ist o.k., wenn ich die Werte aber in einem List&Spreadsheet darstellen will, ergibt sich ein Problem:

Spalte A: Liste {1, 0.1, 0.01, 0.001, ...}

Spalte B: cooling(10,a[]) ergibt falsche Werte!

Spalte C, 1.Zelle: cooling(10,a1) und dann mit „fill“ nach unten ausfüllen ergibt wieder korrekte Werte.

Die falschen Werte in der Spalte B sind offenbar die Werte nach dem ersten Durchgang der While-EndWhile-Schleife in der Definition der Funktion „cooling“. Ich verstehe nicht, wieso die Spalten B und C unterschiedliche Resultate liefern.

Im beiliegenden tns-file ist alles auf einer Seite zu finden.

Preparing for a regional T3-conference in Bern a came across the following problem:

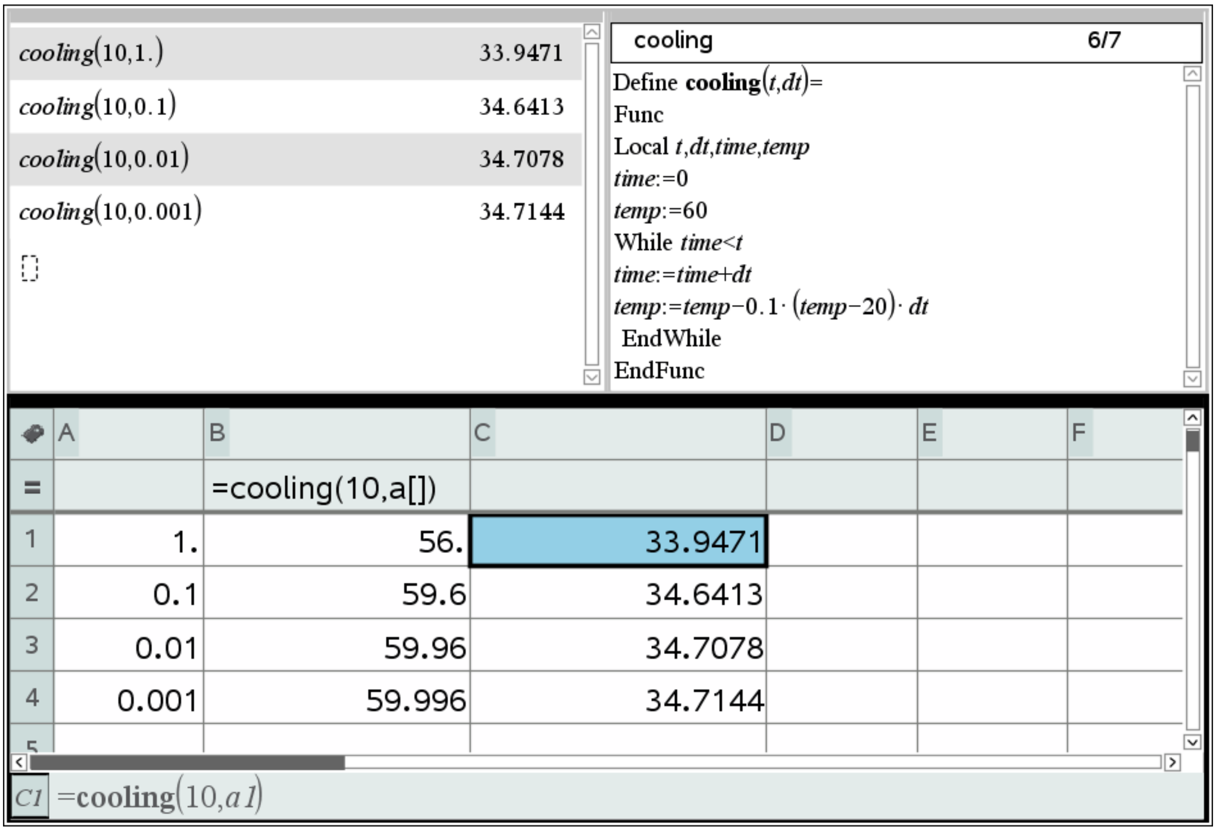
I defined a function "cooling" for calculation the temperature using Euler's method. For studying the convergence behaviour I need values cooling(10,1), cooling(10,0.1) etc. Direct calculation works but when presenting the values in a List&Spreadsheet application a problem appears:

Column *A*: List {1, 0.1, 0.01, 0.001, ...}

Column *B*: cooling(10,a[ ]) gives wrong values!

Column *C*, entering: cooling(10,a1) in cell *C1* and then accomplish downwards using "fill" works as expected.

The wrong values in column *B* are obviously the values obtained after the first run in the while-loop of function „cooling". I don't understand why columns *B* and *C* give different values. See the attached tns-file.



<u>DNL:</u>

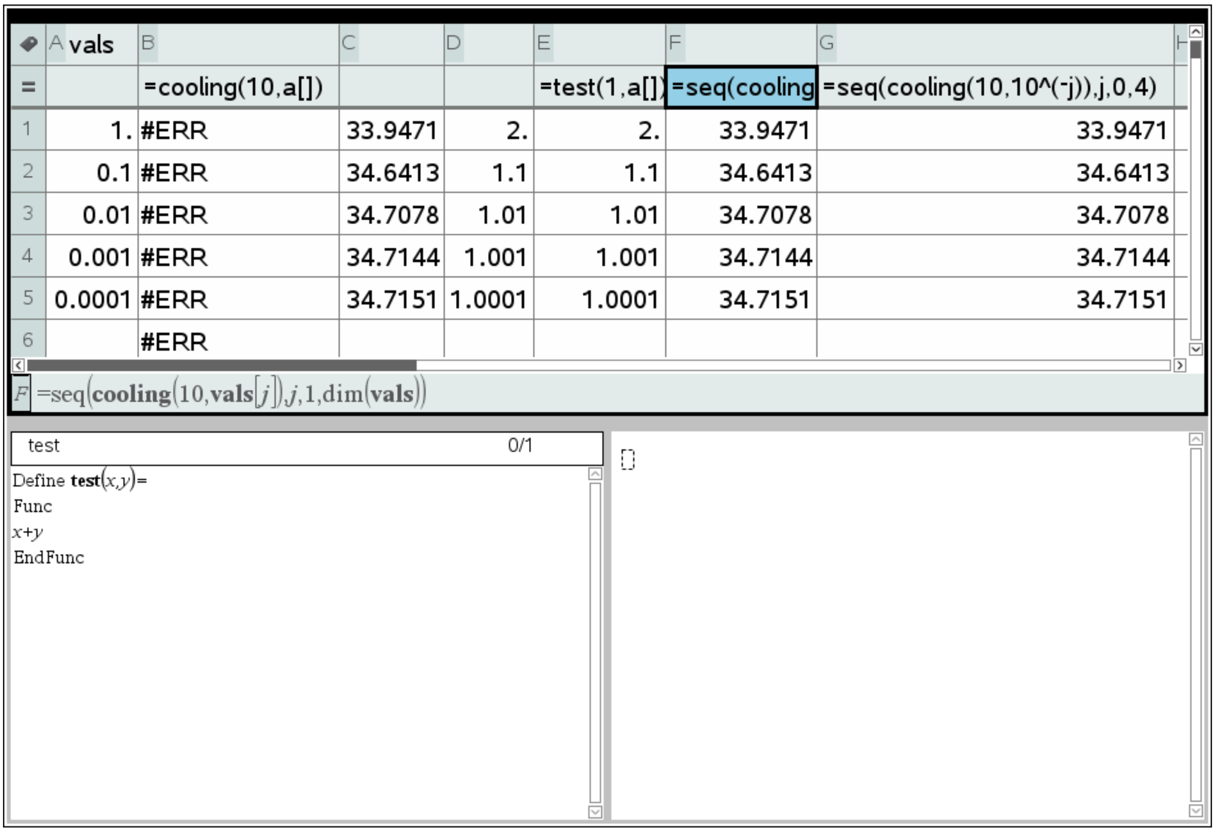
Inspecting YOUR file I can not find any mistake on the first glance – and not even on the next one. I exchanged the while-loop by a for-next-loop – didn't work: one change instead of wrong results I got an error message – even bad.

A test function – without any loop – did not make any problems.

I remember that I came across a similar problem with the Voyage 200 some years ago and David Stoutemyer admitted that it is not possible to work with a loop in such cases. As the CAS-engine of the Nspire is pretty the same in its core, I am not very surprised. What I did on the Voyage works with Nspire, too: I assign the values in col *A* a variable name, say vals. Using a sequence addressing the elements of list vals (see col *F*) gives the expected results.

I prefer the "copy-down-method" (your column *C*) because it corresponds with the method how to work usually in a spreadsheet.

I sent the problem to Nspire experts and am waiting for an answer.



	A vals	B	C	D	E	F	G
=		=cooling(10,a[])			=test(1,a[])	=seq(cooling	=seq(cooling(10,10^(-j)),j,0,4)
1	1.	#ERR	33.9471	2.	2.	33.9471	33.9471
2	0.1	#ERR	34.6413	1.1	1.1	34.6413	34.6413
3	0.01	#ERR	34.7078	1.01	1.01	34.7078	34.7078
4	0.001	#ERR	34.7144	1.001	1.001	34.7144	34.7144
5	0.0001	#ERR	34.7151	1.0001	1.0001	34.7151	34.7151
6		#ERR					

F =seq(cooling(10,vals[j]),j,1,dim(vals))

```
test                                        0/1
Define test(x,y)=
Func
x+y
EndFunc
```

The expert's answer:
Guido Herweyers wrote:
I don't understand this bug?? either. Obviously it is not allowed to use a loop.

seq(cooling(10,10^(-'k)),'k,0,4) works also.

Best regards
Guido

\- - -

MuMath on the PC

Robert Setif [robert.setif@gmail.com]

Betreff: Mumath 83 on Windows 8 64 bits ?

Dear Josef,
I used on a PC (Windows XP) several softwares : Derive, Mathematica 6, Maple 16, XMaxima, MuPad, XCAS, TN-Nspire, and of course MuMath for which I have a strong attachment. But unfortunately I cannot use MuMath on my new PC (portable Windows 8 and 64 bits).
Will it be a trick able to operate in a special session a software 16 bits on a PC Windows 8 (64 bits)?
Thank you very much.
With best regards.
Robert

Fred J. Tydeman

Consider in Derive 6.10 under Windows,

```
fmin := 2^-16382
sqrt(.25 + #i*fmin)
```

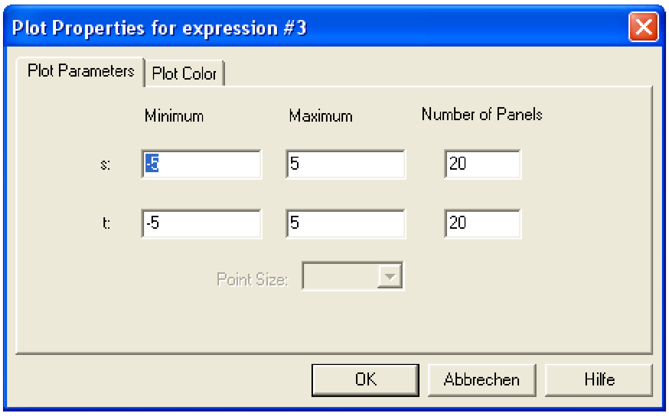
I have tried to approximate that using 200, then 2000, then 20000 digits of accuracy and all three come up with 0.5 (when 0.5 + #i*fmin is a much better approximation).

Is there some better way to have Derive compute the complex sqrt when the imaginary part is very small?

Here is a short contribution from our Iranian member:

Behrooz Khavari [khavari@hamoon.usb.ac.ir], Iran

You know that when we issue the *Edit > Plot* command in the 3D plot window, we encounter with the following window:



Role of "**s**" and "**t**" is depending on structure of equation that we are going to plot. I produced myself a table for various states. I never saw this table in the other sources but it may exist. I hope that it will be unique and useful. I am sorry if my English is not good.

Cylindrical (r,θ,z)

Structure of Equation	s	t
$r=r_0$	θ	z
$r=f(\theta)$	θ	z
$r=f(z)$	z	θ
$r=f(\theta,z)$	z	θ
$\theta=\theta_0$	r	z
$\theta=f(r)$	r	z
$\theta=f(z)$	z	r
$\theta=f(r,z)$	z	r
$z=z_0$	r	θ
$z=f(r)$	r	θ
$z=f(\theta)$	θ	r
$z=f(r,\theta)$	θ	r

Spherical (r,θ,ϕ)

Structure of Equation	s	t
$r=r_0$	θ	ϕ
$r=f(\theta)$	θ	ϕ
$r=f(\phi)$	ϕ	θ
$r=f(\theta,\phi)$	θ	ϕ
$\theta=\theta_0$	r	ϕ
$\theta=f(r)$	r	ϕ
$\theta=f(\phi)$	ϕ	r
$\theta=f(r,\phi)$	r	ϕ
$\phi=\phi_0$	r	θ
$\phi=f(r)$	r	θ
$\phi=f(\theta)$	θ	r
$\phi=f(r,\theta)$	θ	r

Rectangular (x,y,z)

Structure of Equation	x or s	y or t
$x=x_0$	y	z
$x=f(y)$	y	z
$x=f(z)$	z	y
$x=f(y,z)$	y	z
$y=y_0$	x	z
$y=f(x)$	x	z
$y=f(z)$	z	x
$y=f(x,z)$	x	z
$z=z_0$	x	y
$z=f(x)$	x	y
$z=f(y)$	x	y
$z=f(x,y)$	x	y

Using technology (software) in secondary and tertiary mathematics education is my main professional favorite so I am looking for a good position in developed countries to do my PhD in this topic.

I and my wife Somayyeh wrote the following book in Persian:

Teach yourself DERIVE, Dibagaran-e-Tehran publisher, Tehran,Iran, April 2009

Note: Dibagaran-e-Tehran is a famous and great publisher in Iran with following address: http://dibagaran.mft.info/05/En/

This is the first general Derive user's guide in Persian.

Best wishes

Behrooz

Asking for a Permuatations Program

Preparing one of my TIME 2014 talks (Brain Twisters) I wanted to solve one or the other problem with TI-NspireCAS. For this purpose I needed a program to generate permutations of a given set of elements – but not only the *n*! permutations of all elements but also the permutations of all subsets of *k* elements. So I wrote a mail to you all:

Permutations??

Subject: Permutations?

Dear DUG-Members,

is there anybody among you having a TI-program (TI-92, Voyage or Nspire) for generating all permutations of order *k* of a set of *n* elements?

e.g.

perm({1,2,3},2) = [1, 2; 2, 1; 1, 3; 3, 1; 2, 3; 3, 2]

perm({1,2,3}) returns all 6 permutations of the three numbers.

I have a *DERIVE*-function but unfortunately I cannot adjust it for the TIs.

Any advice is highly appreciated.

Best regards

Josef

```
perm(v, n, k_, n_ := 2, s_ := [[1]], t_) :=
  Prog
    Loop
      If n_ > n exit
      k_ := n_
      t_ := []
      Loop
        If k_ = 0 exit
        t_ := APPEND(t_, VECTOR(INSERT(n_, v_, k_), v_, s_))
        k_ :- 1
      s_ := t_
      n_ :+ 1
    v := POWER_SET(MAP_LIST(v↓j_, j_, {1, ..., DIM(v)}), n)
    v := VECTOR(SORT(v_), v_, v)
    APPEND(VECTOR(VECTOR(v_↓u_, u_, s_), v_, v))
```

Example:

```
DIM(perm([a, b, x, y], 3)) = 24

perm([a, b, x, y], 3)
```

$$\begin{bmatrix} b & x & y \\ x & b & y \\ b & y & x \\ x & y & b \\ y & b & x \end{bmatrix}$$

Many thanks to all of you who answered ☺

Danny Ross Lunsford

Pretty sure I copied the logic of that function from this one I found in DNL 41. The function does not do subsetting but should be easy to implement.

```
perms(n)
Func
Local r,s,p,m,k,i,j
{1}→r
For m,1,n-1
 {}→s
 m+1→k
 For i,1,m*m!-m+1,m
  mid(r,i,m)→p
  For j,Ø,m
   augment(s,augment(augment(left(p,m-j),{k}),right(p,j)))→s
  EndFor
 EndFor
 s→r
EndFor
Return list▸mat(r,n)
EndFunc
```

To be honest, it was not easy for me! Josef

Sergey Biryukov, Moscow, Russia

Dear Josef!

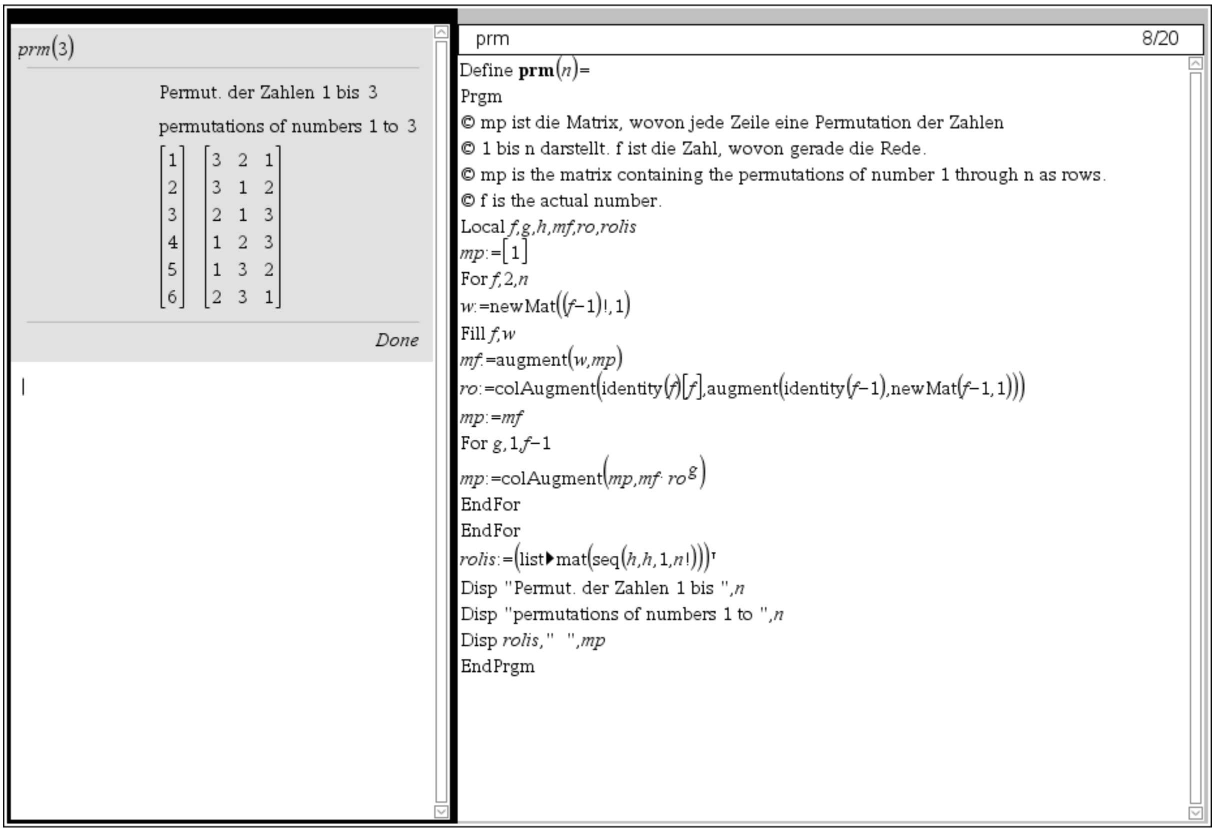
Three permutation algorithms are described at

http://www.cut-the-knot.org/do_you_know/AllPerm.shtml#Levitin

Sincerely, Sergey

Erik van Lantschoot, Weisel, Germany

Lieber Herr Böhm! Zufälligerweise habe ich so was für die TI-Nspire CAS. Ich schicke Ihnen das Programm sofort mit der Post zu. Eine Eigenschaft ist, dass die Lösung von prm(n) aus der Lösung von prm(n−1) abgeleitet wird. Dr. van Lantschoot, Weisel



```
prm(3)
        Permut. der Zahlen 1 bis 3
        permutations of numbers 1 to 3
        [1]  [3 2 1]
        [2]  [3 1 2]
        [3]  [2 1 3]
        [4]  [1 2 3]
        [5]  [1 3 2]
        [6]  [2 3 1]
                                   Done
```

```
prm                                                    8/20
Define prm(n)=
Prgm
© mp ist die Matrix, wovon jede Zeile eine Permutation der Zahlen
© 1 bis n darstellt. f ist die Zahl, wovon gerade die Rede.
© mp is the matrix containing the permutations of number 1 through n as rows.
© f is the actual number.
Local f,g,h,mf,ro,rolis
mp:=[1]
For f,2,n
w:=newMat((f-1)!,1)
Fill f,w
mf:=augment(w,mp)
ro:=colAugment(identity(f)[f],augment(identity(f-1),newMat(f-1,1)))
mp:=mf
For g,1,f-1
mp:=colAugment(mp,mf·ro^g)
EndFor
EndFor
rolis:=(list▸mat(seq(h,h,1,n!)))^T
Disp "Permut. der Zahlen 1 bis ",n
Disp "permutations of numbers 1 to ",n
Disp rolis,"  ",mp
EndPrgm
```

Mr Lantschoot sent an Nspire-program for generating the permutations of *n* elements. Program together with an extended explanation follows on the next pages. Many thanks to Mr. van Lantschoot.

Sehr geehrter Herr Boehm!

Ich weiß nicht, ob Sie mit meinem Programm viel weiter gekommen sind, weil ich Art und Umfang Ihres Vorhabens „Logelei" nicht kenne. Die Crux mit dem Programm permmatrix ist dass für n > 6 die Matrix nicht mehr gespeichert werden kann, da zu groß. Deshalb habe ich ein anderes Programm roperm(n,rw) entwickelt, das „nur" die Reihe *rw* von perm(n) berechnet. Ich lege beide Programme in Beilage. Entschuldigen Sie mich, dass die "Notes" dazu noch nicht ganz fertig sind. Aber die Berechnungen stimmen, so weit ich mit Beispielen herausfinden konnte. Übrigens: Die "Logelei", wie von Heinrich Heine besungen, ist ein Felsmassiv, das sich etwa 9 km von hier befindet. Schönen Gruss, EvL

Sehr geehrter Herr Boehm! Ich lege ein Programm roperm(rw,n) bei, das die Reihen der Permutationsmatrix *mp(n)* einzeln, Reihe für Reihe berechnen kann. Somit ist die Möglichkeit gegeben, mit recht großen Werten von *n* zu rechnen. Wenn das Programm unbeschadet bei Ihnen ankommt, hätte ich gerne eine Bestätigung. MfG EvL

A METHOD FOR GENERATING ALL PERMUTATIONS OF n OBJECTS

Dr,.Ing. E. van Lantschoot, e–mail: Lantschoot_Weisel@web.de

Let **mp(n)** be a matrix, the rows of which show all permutations of n objects. Such a matrix has n! rows, and if n>6, the storage capacity of TI–Nspire is exceeded. Therefore, a program **roperm(rw,n)** was developed, which computes each row **rw** of **mp(n)** individually.

Let us take a look at page 2 which shows **mp(4)** as developed using a particular, recursive method.

A first important statement is that **mp(n–1)...mp(1)** appear as nested submatrices of **mp(n).** The submatrix **mp(4)[1,4,1,4]** (the figures in square brackets are the indices of the upper left, resp. lower right corners of the submatrix) = **mp(1).** The submatrix **mp(4)[1,3,2,4] = mp(2).** The recursive method referred to is explained by showing how we proceed from **mp(2)** to **mp(3).** The generalization from **n** to **n+1** is then obvious.

We construct what we call a subblock, by taking **mp(2)** augmented by a new first column, which contains rows of 3s: thus $\begin{bmatrix}3 & 2 & 1\\3 & 2 & 1\end{bmatrix}$. Since **mp(3)** will have 3!=6 rows, three such subblocks under each other will be needed. The second subblock is obtained by applying a one–time shift to the left on each row, thus: $\begin{bmatrix}2 & 1 & 3\\1 & 2 & 3\end{bmatrix}$. In the same manner, the third subblock is obtained from the second. The three subblocks together constitute the submatrix **mp(4)[1,2,6,1] = mp(3).** In the argument which follows, a **block** will refer to a submatrix with dimensions **dim(block)={(st–1)!,n}**, **st** being the number of the step we are working at.

The program roperm(rw,n) implements this recursive method. But first, we have to occupy ourselves with the important function blobef(rw,st), a name which stands for "blocks of size **{(st–1)!,st}** before **rw** in **mp(st)**". "before" is to be taken in true sense, i.e. "not including rw". That is the reason why an "if" appears in the function.

Consider the row 21 of **mp(4)** as an example. Since **mp(3)** has 3!=6 rows, **blobef(21,4) = 3.** Then compute **rw := mod(rw,(4–1)!) = 3**, which is an indication that row 21 is row 3, shifted **blobef** times to the left. Thus, if we already had row 3 = [4 2 1 3], the transition to row 21 would be obvious. Let us remember that **blobef(rw,n)** computes the number of blocks of size **{(n–1)!,n}** in **mp(n)** before **rw.** Now that we have found that row 21 in **mp(4)** would be row 3 in **mp(3)** we can use **blobef(3,3) = 1** to ascertain what the row number of row 21 would be in **mp(2).** We find [2 1].

Finally, there is a trick to be accounted for. In the program the **blobef** function values should be called for in the succession **st = 1...n** by **blobef(<u>actual row in mp(st)</u>,st)**, but thery are rather called for by **blobef(<u>row in mp(n)</u>,st).** The statement "return **mod(h,st)**" takes care of all superfluous shifts.

The program roperm(rw,n) begins with the solution vector sol = {1} and applies the recursive method. Its operation is exemplified by the program **print()**, which shows how we proceed from the first solution vector to the next.

The performance of the program is spectacular. If n = 12, the number of rows of **mp(12)** is almost half a billion units. If we ask for **roperm(469341393)** we obtain the answer in no time: for rw = 469341393 solut. is $\{4,12,10,5,7,11,6,8,2,1,9,3\}$.

The function and the programs follow:

blobef 3/3

```
Define blobef(a,st)=
Func
Local h: h:=intDiv(a,(st−1)!)
If st=1 Then: Return 0: ElseIf remain(a,(st−1)!)=0 Then: h:=h−1: EndIf
Return mod(h,st)
EndFunc
```

print 3/4

```
Define print(n,st,rw,sol)=
Prgm
Disp "st = ",st,"  blobef = ",blobef(rw,st),"  sol = ",sol
If st=n Then: Disp "⬚": Goto raus: EndIf
Disp "Now put an ",st+1," in front and left−shift ",blobef(rw,st+1)," times"
Lbl raus
EndPrgm
```

roperm 9/9

```
Define roperm(rw,n)=
Prgm
Local rw,sol,st
© blobef(rw,st) is explained in the text
For st,1,n
If st=1 Then:{1}→sol: Goto raus: EndIf
sol:=rotate(augment({st},sol),blobef(rw,st))
Lbl raus:
print(n,st,rw,sol)
EndFor
Disp "for rw = ",rw," solut. is ",sol
EndPrgm
```

On the next page you can find two sample runs of roperm().

```
roperm(469341393,12)

st = 1  blobef = 0  sol = {1}
Now put an 2 in front and left−shift 0 times
st = 2  blobef = 0  sol = {2,1}
Now put an 3 in front and left−shift 1 times
st = 3  blobef = 1  sol = {2,1,3}
Now put an 4 in front and left−shift 1 times
st = 4  blobef = 1  sol = {2,1,3,4}
Now put an 5 in front and left−shift 1 times
st = 5  blobef = 1  sol = {2,1,3,4,5}
Now put an 6 in front and left−shift 0 times
st = 6  blobef = 0  sol = {6,2,1,3,4,5}
Now put an 7 in front and left−shift 2 times
st = 7  blobef = 2  sol = {2,1,3,4,5,7,6}
Now put an 8 in front and left−shift 3 times
st = 8  blobef = 3  sol = {3,4,5,7,6,8,2,1}
Now put an 9 in front and left−shift 3 times
st = 9  blobef = 3  sol = {5,7,6,8,2,1,9,3,4}
```

```
Now put an 12 in front and left−shift 11 times
st = 12  blobef = 11  sol = {4,12,10,5,7,11,6,8,2,1,9,3}

for rw = 469341393 solut. is {4,12,10,5,7,11,6,8,2,1,9,3}

Fertig

roperm(21,4)

st = 1  blobef = 0  sol = {1}
Now put an 2 in front and left−shift 0 times
st = 2  blobef = 0  sol = {2,1}
Now put an 3 in front and left−shift 1 times
st = 3  blobef = 1  sol = {2,1,3}
Now put an 4 in front and left−shift 3 times
st = 4  blobef = 3  sol = {3,4,2,1}

for rw = 21 solut. is {3,4,2,1}

Fertig
```

Until now there were some answers but no one could give the subset permutations. Then a mail from Benno Grabinger came in (see also the Information Page!!):

<u>**Benno Grabinger, Germany**</u>

Lieber Josef,

im Anhang findest du ein Nspire Dokument mit dem man k-Permutationen erzeugen kann. Interessant ist auch die dabei verwendete Funktion „nextperm“ mit der man die lexikografisch nächste Permutation (aus „beliebig“ vielen Elementen) erzeugen kann.

Was macht das Tennis?

Liebe Grüße,

Benno

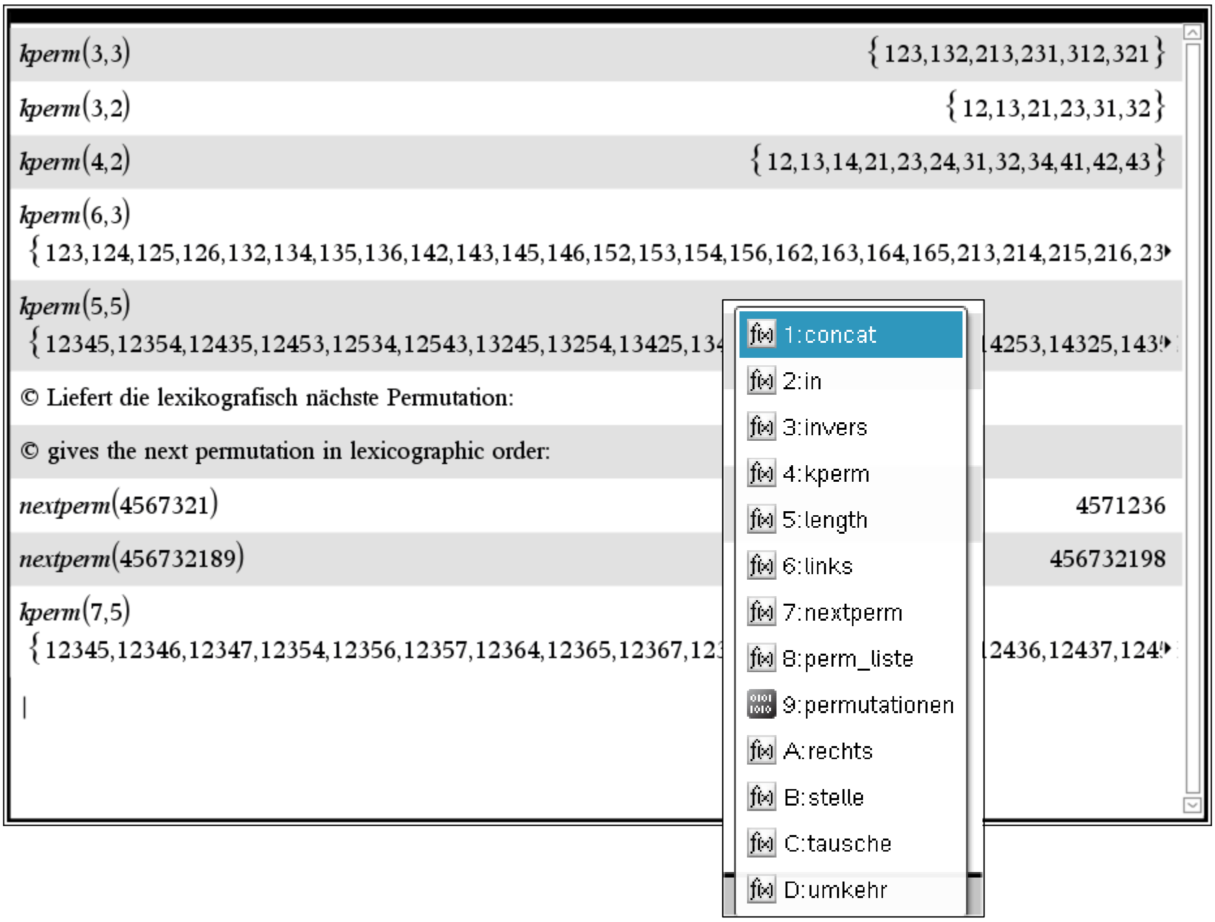
Dear Josef,

attached you will find an Nspire document which makes possible generating your requested permutations. Interesting is the function “nextperm” which creates the next permutation in lexicographic order (of “arbitrary” many elements).

What about your tennis playing?

Best regards,

Benno



As you can see Benno needs a lot of auxiliary functions. His program works but unfortunately it is restricted because of out of memory messages for a bit greater parameters *n* and – even on the PC.

And there was assistance from Carl Leinbach, too:

Carl L. Leinbach, USA

I've been thinking about your question on permutations. Unfortunately, I have been preoccupied with testing the programs on continued fractions and have not had time to write the program, but it should not be hard. I will work on it today, but just in case, here is the idea:

It will be a recursive program.

We all know the permutation(s) of a list containing one object, say [1]

Now to move to the permutations of a list containing two objects:

Write two copies of the list containing one object: [1] [1]

Expand each copy by placing a 2 in the i-th position of the i-th copy: [2,1] [1,2]

We now have a list of the permutations of a list containing two objects

Next we will repeat the process and create a list of the permutations of three objects (that is what I did in the .tns file showing the N-spire operations that I plan to use .

Write three copies of the permutations of the list containing two objects: [2,1] [1,2] [2,1] [1,2] [2,1] [1,2]

Expand the list generated by placing a 3 in the i-th position if the permutation came from the i-th copy [3,2,1] [3,1,2] [2,3,1] [1,3,2] [2,1,3] [1,2,3]

Here are the permutations of a list containing 3 objects - If I go any further in this this e-mail, I will never get it sent.

$\text{colAugment}\left(\begin{bmatrix}1&2\\2&1\end{bmatrix},\begin{bmatrix}3&3\end{bmatrix}\right) \quad \begin{bmatrix}1&2\\2&1\\3&3\end{bmatrix}$

$\begin{bmatrix}1&2\\2&1\\3&3\end{bmatrix}^T \quad \begin{bmatrix}1&2&3\\2&1&3\end{bmatrix}$

$\text{mat▸list}\left(\begin{bmatrix}1&2&3\\2&1&3\end{bmatrix}\right) \quad \{1,2,3,2,1,3\}$

$\text{rowSwap}\left(\begin{bmatrix}1&2\\2&1\\3&3\end{bmatrix},2,3\right) \quad \begin{bmatrix}1&2\\3&3\\2&1\end{bmatrix}$

$\begin{bmatrix}1&2\\3&3\\2&1\end{bmatrix}^T \quad \begin{bmatrix}1&3&2\\2&3&1\end{bmatrix}$

$\text{mat▸list}\left(\begin{bmatrix}1&3&2\\2&3&1\end{bmatrix}\right) \quad \{1,3,2,2,3,1\}$

$\text{augment}(\{1,2,3,2,1,3\},\{1,3,2,2,3,1\}) \quad \{1,2,3,2,1,3,1,3,2,2,3,1\}$

$\text{rowSwap}\left(\begin{bmatrix}1&2\\2&1\\3&3\end{bmatrix},1,3\right) \quad \begin{bmatrix}3&3\\2&1\\1&2\end{bmatrix}$

$\begin{bmatrix}3&3\\2&1\\1&2\end{bmatrix}^T \quad \begin{bmatrix}3&2&1\\3&1&2\end{bmatrix}$

$\text{mat▸list}\left(\begin{bmatrix}3&2&1\\3&1&2\end{bmatrix}\right) \quad \{3,2,1,3,1,2\}$

$\text{augment}(\{1,2,3,2,1,3,1,3,2,2,3,1\},\{3,2,1,3,1,2\}) \quad \{1,2,3,2,1,3,1,3,2,2,3,1,3,2,1,3,1,2\}$

$\text{list▸mat}(\{1,2,3,2,1,3,1,3,2,2,3,1,3,2,1,3,1,2\},3) \quad \begin{bmatrix}1&2&3\\2&1&3\\1&3&2\\2&3&1\\3&2&1\\3&1&2\end{bmatrix}$

next_perm

```
Define LibPub next_perm(pr,n)=
Func
  Local i,m,p,per,p1
  m:=dim(pr)[1]
  p:={⬚}
  For i,1,m
    p:=augment(p,{n})
  EndFor
  per:=colAugment(pr^T,list▸mat(p))
  p:=mat▸list(per^T)
  m:=dim(per^T)[2]
  For i,m−1,1,-1
     p1:=rowSwap(per,i,m)
     p:=augment(p,mat▸list(p1^T))
⬚
   EndFor
   Return list▸mat(p,n)
EndFunc
```

genperm

```
Define LibPub genperm(n)=
Func
   Local per,i
   per:=[1]
   If n=1 Then
      Return per
   Else
      For i,2,n
        per:=next_perm(per,i)
      EndFor
      Return per
    EndIf
EndFunc
```

```
genperm(7)[1022]                    "Error: Resource exhaustion"
genperm(6)[513]                     [1 6 4 5 2 3]
genperm(4)                          [1 2 3 4
                                     2 1 3 4
                                     1 3 2 4
                                     2 3 1 4
                                     3 2 1 4
                                     3 1 2 4
                                     1 2 4 3
                                     2 1 4 3
                                     1 3 4 2
                                     2 3 4 1
                                     3 2 4 1
                                     3 1 4 2
                                     1 4 3 2
                                     2 4 3 1
                                     1 4 2 3
                                     2 4 1 3
                                     3 4 1 2
                                     3 4 2 1
                                     4 2 3 1
                                     4 1 3 2
```

I expect that if you want all of the permutations of a list of n objects with n large, it will take a while. Recursive programs are that way – even when you write them as iterative programs. I hope this is helpful and I will start to work on the program. The model that I showed in the .tns file will be my guide.

Just to accomplish the issue I present the DERIVE function to generate the permutations with repetitions considering the order – which are also called variations with repetitions.

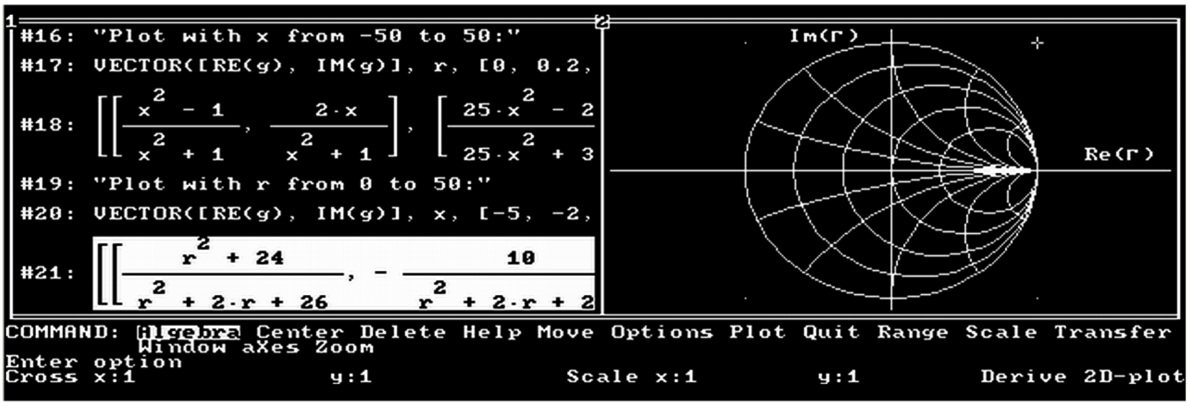
```
vars(v, k, b, k_, m_ := 0, n_, s_ := [], t_) :=
  Loop
    b := DIM(v)
    If m_ = b^k
       RETURN REVERSE(s_)
    k_ := k
    n_ := m_
    t_ := []
    Loop
      t_ := ADJOIN(v↓(MOD(n_, b) + 1), t_)
      n_ := FLOOR(n_, b)
      k_ :- 1
      If k_ = 0 exit
    s_ := ADJOIN(t_, s_)
    m_ :+ 1
```

```
vars([x, y, z, 6], 3)

DIM(vars([x, y, z, 6], 3)) = 64

x  x  x
x  x  y
x  x  z
x  x  6
  ...
  ...
6  6  z
6  6  6
```

What I did with the permutation program can be found on page 38, Josef

Rick Nungester (eDUG)

Because activity here is low, and because Math is Fun, and because I recently did this, here is a screenshot of Derive for DOS version 3.04 running on my early-1990s HP 200LX Palmtop that I still use daily, plotting a Smith Chart (used in radio frequency electronics design).



Welcome to a new member from Russia

Name: Sergey Kamenev
Institution: Bauman Moscow State Technical University (postgraduate)

Hello, Josef!

Please, join me Derive User Group.

I have very interesting materials about loading in Derive millions of rows data for analysis.

This is what he sent:

This is what he sent:

Hello Josef!

In this letter I share my experience with Derive 6.01 in loading big data: up to million rows of data.

Theoretically Derive is able to load 4*10^6 rows of data if 4GB memory installed.

1. You need PC with at least 4GB RAM. If you using Derive in virtual pc (I use Vmware Player) allocate these memory. Vmware player slower 28% on Derive task vs real PC.
   See used by Derive memory on "Help"->"About Derive" window.

   In Derive option set reserve 80% of system RAM on startup.

2. Prepare data. You need convert you data to text format.
   On every row of file you need 1 row of data.
   Delimiter - comma.

3. Use my program der_loader.php for convert you text file in DFW-file.
   Program tested on linux, but with minor fixes might work for Win.

   You need PHP installed. Your php might be able running from command line.
   Test: php - ver

   Be sure you set big memory for php in php.ini file. I set variable memory_limit=999M for converting big files.

   Program tested under php 5.3.8 and 5.3.28

   Command for conversion:
   ./der_loader.php you_file.txt > new_derive_file.dfw

   For Win you can drop first line from der_loader.php and run command:
   php -f der_loader.php you_file.txt > new_derive_file.dfw

   I did not test der_loader.php under Windows.

   Conversion internal logic: Derive very slowly loading very large variables.
   My converter splits big file in small variables s0_, s1_, ..., sN_.

   After loading you need simplify included function for join this variables to giant array.

4. Loader generated file to Derive.
   Loading attached example file USDEUR5.csv on my notebook 2.3Ghz under Vmware Player - 16.5 sec.

   On desktop PC 3.8Ghz under Vmware Player ~8 sec.

   Another test result from my PC (another file):
   50 000 rows 5.2 sec. File size 1.9M.
   200 000 rows 26 sec. File size 7.9M.
   400 000 rows 59 sec. DIM(s) - 1.6 sec. File size 15.5M.
   1 000 000 rows - 5 min 28 sec. DIM(s) - 4.7 sec. Memory used - 19% on PC with 3.5Gb RAM. File size 39M.

5. After loading simplify included in .dfw file function:
   SIMPLIFY_ME_BEFORE_WORK()

   It's needed for constructing giant array "s" from small variables s0_, s1_ and so on.

6. Before save simplify included function:

SIMPLIFY_ME_BEFORE_SAVE(s, "s")

If your data loading is lasting many minutes use next trick: after work suspend you Virtual PC with opened Derive.

Next time your virtual PC loaded with opened Derive with load the data within seconds.

Caution:

Do not try to display giant array on screen: Derive will hang up.
And do not try loading many thousand lines file from File->Load Data File. It's lowly and Derive hangs up trying to display data after loading.

Nice day
Sergey Kamenev

Here is another mail from Sergey. You are invited to contact him, Josef

Sergey Kamenev [derive14@slon.pp.ru]

Hello!

I use Derive from 1995 for learning.
Today I use a Derive to test their hypotheses in the areas of compression and multilevel marketing.

I 80% understand the format *.dwf files.

Do you know how to convert Derive files into Latex-files?

I do not know whether anyone interested in my problems. I will try to send you, if I find something interesting from my work.

A new version of Theorema can be downloaded!

publicity-bounces@risc.jku.at (RISC Secretary)

Dear friends and colleagues,

it is a great pleasure to announce that

*** Theorema 2.0 is available for download ***

since Friday, July 11, 2014. Please visit

http://www.risc.jku.at/research/theorema/software/

for any further information. You might also visit

https://www.facebook.com/mathematicsTheorema

We hope you enjoy,
Bruno Buchberger & The Theorema Group
RISC, JKU Linz, Austria.

Space Curves with curvature and torsion proportional to their arc length

Piotr-Andrzej Trebisz [Piotr-Andrzej.Trebisz@gmx.de]

Hallo Herr Böhm,

Mein Computer war kaputt, deshalb kommen die Datei mit den Raumkurven erst jetzt. Die Krümmung der Kurven „SPIRALE_2D“ und „SPIRALE_3D“ nimmt proportional zur Länge „s“ ab. Zusätzlich nimmt bei „SPIRALE_3D“ auch die Torsion proportional zur Länge „s“ ab. Bei den beiden Kurven „KLOTHOIDE_2D“ und „KLOTHOIDE_3D“ verhält es sich genau umgekehrt. „Kappa κ ist der Proportionalitätsfaktor der Krümmung, „Tau τ“ ist der Proportionalitätsfaktor für die Torsion.

Mit freundlichen Grüßen
Piotr Trebisz

Hello Mr. Böhm,
my computer was out of order, that is the reason why the space curves are coming late. Curvature of curves "SPIRALE_2D" and "SPIRALE_3D" decreases proportional to arc length "s". Additionally in "SPIRALE_3D" torsion also decreases proportional to “s”. In curves "KLOTHOIDE_2D" and "KLOTHOIDE_3D" it is just reverse. κ and τ are the constants of proportionality for curvature and torsion, respectively.
With best regards
Piotr Trebisz

```
#1:  [CaseMode := Sensitive, InputMode := Word]

#2:  [κ :∈ Real, τ :∈ Real, s :∈ Real [0, ∞), t :∈ Real]

#3:  ------------------------------------------------------------------
```

#4:

$$\text{SPIRALE\_2D}(\kappa, s) := \left[\frac{s\cdot(\text{COS}(\kappa\cdot\text{LN}(s)) + \kappa\cdot\text{SIN}(\kappa\cdot\text{LN}(s)))}{\kappa^2 + 1}, \frac{s\cdot(\text{SIN}(\kappa\cdot\text{LN}(s)) - \kappa\cdot\text{COS}(\kappa\cdot\text{LN}(s)))}{\kappa^2 + 1}\right]$$

#5:

$$\text{SPIRALE\_3D}(\kappa, \tau, s) := \left[\frac{s\cdot|\kappa|\cdot\left(\dfrac{\text{COS}(\sqrt{\kappa^2 + \tau^2}\cdot\text{LN}(s))}{\sqrt{\kappa^2 + \tau^2}} + \text{SIN}(\sqrt{\kappa^2 + \tau^2}\cdot\text{LN}(s))\right)}{\kappa^2 + \tau^2 + 1}, \frac{s\cdot\kappa\cdot\left(\dfrac{\text{SIN}(\sqrt{\kappa^2 + \tau^2}\cdot\text{LN}(s))}{\sqrt{\kappa^2 + \tau^2}} - \text{COS}(\sqrt{\kappa^2 + \tau^2}\cdot\text{LN}(s))\right)}{\kappa^2 + \tau^2 + 1}, \frac{s\cdot\tau\cdot\text{SIGN}(\kappa)}{\sqrt{\kappa^2 + \tau^2}}\right]$$

```
#6:  ------------------------------------------------------------------
```

Some plots to Piotr's 2D- and 3D-Curves

#7: $\text{FRESNEL\_2D}(\kappa, t) := \sqrt{\left(\frac{\pi \cdot i}{2 \cdot \kappa}\right)} \cdot \text{ERF}\left(\sqrt{\left(-\frac{i \cdot \kappa}{2}\right)} \cdot t\right)$

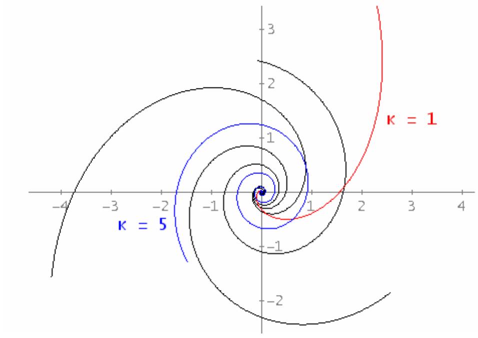
#8: $\text{FRESNEL\_3D}(\kappa, \tau, t) := \sqrt{\left(\frac{\pi \cdot i}{2 \cdot \sqrt{(\kappa^2 + \tau^2)}}\right)} \cdot \text{ERF}\left(\sqrt{\left(-\frac{i \cdot \sqrt{(\kappa^2 + \tau^2)}}{2}\right)} \cdot t\right)$

#9: --

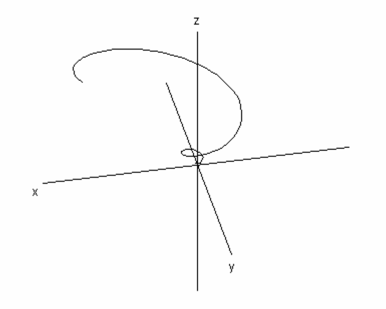
#10: $\text{KLOTHOIDE\_2D}(\kappa, t) := [\text{RE}(\text{FRESNEL\_2D}(\kappa, t)), \text{IM}(\text{FRESNEL\_2D}(\kappa, t))]$

#11: $\text{KLOTHOIDE\_3D}(\kappa, \tau, t) := \left[\frac{|\kappa| \cdot \text{RE}(\text{FRESNEL\_3D}(\kappa, \tau, t))}{\sqrt{(\kappa^2 + \tau^2)}}, \frac{\kappa \cdot \text{IM}(\text{FRESNEL\_3D}(\kappa, \tau, t))}{\sqrt{(\kappa^2 + \tau^2)}}, \frac{t \cdot \tau \cdot \text{SIGN}(\kappa)}{\sqrt{(\kappa^2 + \tau^2)}}\right]$

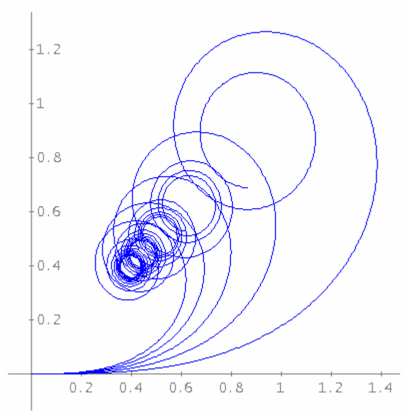
VECTOR(SPIRALE_2D(κ, s), κ, 1, 5)



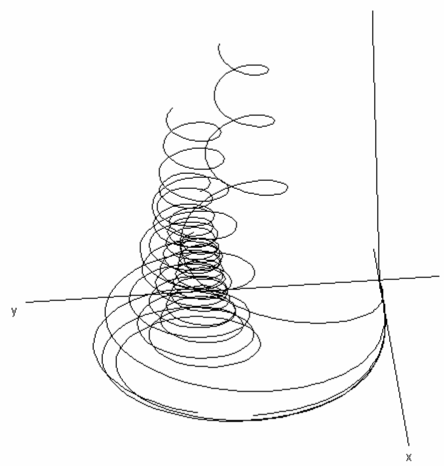
SPIRALE_3D(3, 1, s)



VECTOR(KLOTHOIDE_2D(κ, t), κ, 1, 5)



VECTOR(KLOTHOIDE_3D(κ, 2, t), κ, 1, 5)



Comments on earlier DNL articles from DownUnder.

David Halprin, Australia

Hi Josef

Firstly, I downloaded and read DNL93 cover to cover a few nights back, and there is much for me to say.

Once again, I thank you sincerely for all your mammoth effort of transcribing and editing all those equations. Your additions from Derive and TI-NspireCAS files really transformed the paper into so much more than I could have envisioned, thereby making the reading of it so much more fascinating.

When reading Adrian Oldknow's Keynote Address, there were two parts, that "hit the right chord" with me and my special interests:

On page 41 he mentions that "the human is always playing a supervisory role" etc.. That is exactly what I was intimating in my letter to you on page 21 re "checking out with pen and paper" and my letter to you on page 32 re "a degree of caution", with CAS.

Re the water-sprinkler problem, that I mentioned in my letter at the top of page 29 and Adrian's Exhibit 3 on page 36.

In January 1985, Bart Braden, a senior mathematics lecturer on the staff of Northern Kentucky University published "Design of an Oscillating Sprinkler" in The College Mathematics Magazine, Vol. 58, No.1.

I shall include a link to it:

http://www.maa.org/programs/maa-awards/writing-awards/design-of-an-oscillating-sprinkler[*]

Essentially, he presented it in 3 parts:-

1) The curvature of the arm.
2) The control of the rocking motion.
3) How to drive the sprinkler arm in the desired motion.

Adrian Oldknow describes the geometry of his system, therefore a comparison with Bart Braden's third part would be an interesting exercise, nicht wahr???

In 1985, I wrote to Bart and included my intrinsic solution to his first part, and as a bonus, I identified the curve as an epicycloid, which he had not done. He wrote me back and thanked me and asked about the intrinsic method. We exchanged a few letters over the years, during which time he became a professor. In 2000 he retired and became an emeritus professor. I wrote to him a few years later, asking his permission to write up and publish his paper, in part, in the unimelb MUMS journal, called Paradox. He wrote back, giving me permission, and I promised that I would send him a copy. Well, I never did write it up, so it "sat on the back-burner".

[*] You can find many excellent articles published by MAA (Mathematical Association of America) on http://www.maa.org/programs/maa-awards/writing-awards. Josef

So, after seeing Adrian's address, I do intend now to write it up and submit it to you for a future DNL, I hope. If you have the time to read Braden's paper, I would welcome your opinion. Maybe you, personally, could tackle parts 2 and 3, and we could combine them as one co-authored paper hoffentlich, vielleicht????

Thanks for the 3 examples of generating functions. I have never seen that before; I was incredulous when I read them.

BTW, I encountered some interesting generating functions for some special functions and polynomials elsewhere.

e.g. Bessel functions,(incl. modified BF),
Legendre functions, named for Rodrigue,
Hermite polynomials, named for Rodrigue,
Laguerre polynomials, named for Rodrigue,
Chebyshev polynomials, named for Rodrigue.

Harking back to early 2013 when I sent you the original copy of the GAS and said, it was not fit for publication “as-is” since it was flawed in some parts, so keep it for reading only. Well, the flawed parts were those two sets of equations, that you discovered to be missing from my 2014 submission, due to my deletion of them entirely. They were the above functions and polynomials, that apart from a generating function, they had a definition with a recursive relationship and it involved “n”, which I later found them non-amenable to my methods, schrecklich alas.

Re "Light in the Coffee Cup". I have written a short paper re the reflected rays on the surface, with a different approach than Roland Schröder, (ENVELOP2.PDF). Also some Derive files etc.

Roland mentioned a three-leaved clover, which reminded me of the Fractal paper, that demonstrates basins of stability ranging from cardioid, nephroid, thru to "8 leaved clover". So I am sending it to you to put in the queue for a DNL. Enjoy!!!

Herzlichst
David

ENVELOPES OF TANGENTS

A plane curve may be represented yet another way, by the definition, that it is the envelope of its tangents. This is a statement of the obvious, however one can give the equation to the tangents, and thereby a curve is defined. This also opens up a new way of grouping curves. This illustrates how a curve can be drawn with only straight lines, and has a practical application in the art of ‘curve-stitching’.

The equation of the tangents is given as a family of straight lines, with a variable parameter, *c*. viz:–

$H(x,y,c) = 0$ is the equation to the tangents.

$\frac{\partial H}{\mathrm{d}c} = H_c = 0$ is the partial derivative.

Solve these two to obtain the parametric equations for the curve.

Proof: Start with the curve $x = f(c)$ and $y = g(c)$, where c is the parameter.

We treat c as the only variable and we partially differentiate,

$$\frac{\partial x}{\mathrm{d}c} = \frac{\partial f}{\mathrm{d}c} = f' \quad \text{and} \quad \frac{\partial y}{\mathrm{d}c} = \frac{\partial g}{\mathrm{d}c} = g'$$

where

$$\frac{\partial^2 x}{\mathrm{d}^2 c} = \frac{\partial^2 f}{\mathrm{d}^2 c} = f'' \quad \text{and} \quad \frac{\partial^2 y}{\mathrm{d}^2 c} = \frac{\partial^2 g}{\mathrm{d}^2 c} = g''$$

Equation to tangents, Cartesian & Polar resp.:–

$$g' x - f' y = f g' - f' g \quad \text{and}$$

$$x(\dot{r}\sin\theta + r\cos\theta) - y(\dot{r}\cos\theta - r\sin\theta) = r^2.$$

Therefore

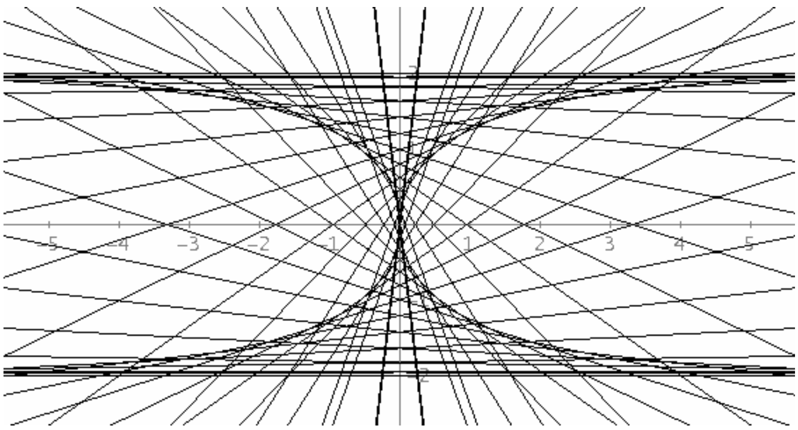
$$H = g' x - f' y - f g' + f' g = 0$$
$$H_c = g'' x - f'' y - f g'' + f'' g = 0$$

On solving these simultaneous equations for x and y we obtain $x = f$ and $y = g$. Q.E.D.

e.g. 1: Kappa Curve $\quad r = a \cdot \cot\theta \quad$ or $\quad y^2(x^2 + y^2) = a^2 x^2$

Tangents: $\quad x\sin^3\theta - y\cos\theta(1 + \sin^2\theta) + a\cos^2\theta = 0$

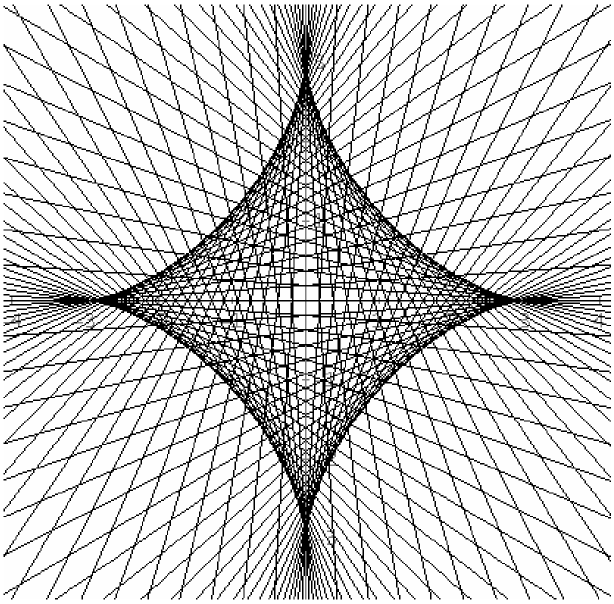
```
                   3                        2                2
SOLVE(x·SIN(θ)  - y·COS(θ)·(1 + SIN(θ) ) + 2·COS(θ)  = 0, y)

                                                  2             3
                                          2·COS(θ)  + x·SIN(θ)
                                    y = ───────────────────────
                                                          2
                                           COS(θ)·(SIN(θ)  + 1)

              2              3
       2·COS(θ)  + x·SIN(θ)                          π
VECTOR(───────────────────────, θ, 0, 2·π, ──)
                        2                           25
        COS(θ)·(SIN(θ)  + 1)
```



e.g. 2: Astroid $(x^2+y^2-a^2)^3+27a^2x^2y^2=0$ or $x^{\frac{2}{3}}+y^{\frac{2}{3}}=a^{\frac{2}{3}}$ or $x=a\cos^3\theta, y=a\sin^3\theta$

Tangents: $2x\sin\theta+2y\cos\theta-a\sin 2\theta=0$

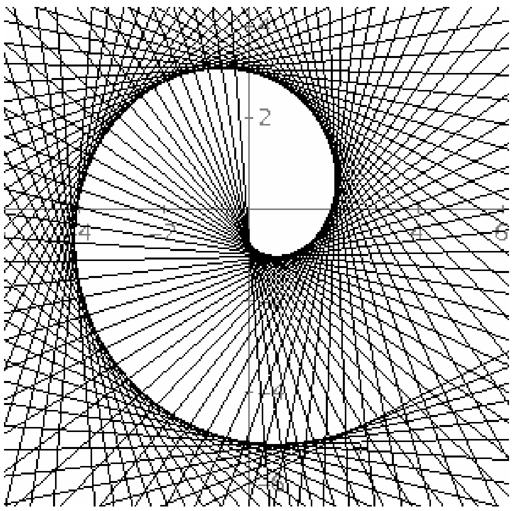
```
SOLVE(2·x·SIN(θ) + 2·y·COS(θ) = 3·SIN(2·θ) = 0, y)

SIN(2·θ) = 0 ∧ y = 3·SIN(θ) - x·TAN(θ)

VECTOR(3·SIN(θ) - x·TAN(θ), θ, 0, 2·π, π/50)
```



e.g. 3: Cayley's Sextic $(x^2+y^2-a^2)^3+27a^2x^2y^2=0$ or $x^{\frac{2}{3}}+y^{\frac{2}{3}}=a^{\frac{2}{3}}$ or

Tangents: $2x\sin\theta+2y\cos\theta-a\sin 2\theta=0$

```
SOLVE(x·COS(4·θ/3) + y·SIN(4·θ/3) - 4·COS(4)·θ/3 = 0, y)

y = 4·θ·COS(4)/(3·SIN(4·θ/3)) - x·COT(4·θ/3)

VECTOR(4·θ·COS(4)/(3·SIN(4·θ/3)) - x·COT(4·θ/3), θ, 0, 2·π, π/50)
```



e.g. 4: Lamé Curves $\left(\frac{x}{a}\right)^n + \left(\frac{y}{b}\right)^n = 1$ or $x = a\cdot\cos^{\frac{2}{n}}\theta,\ y = b\cdot\sin^{\frac{2}{n}}\theta$

Algebraic when n is rational, otherwise transcendental.

$n = ½$ a very special case of parabola

$n = -2$ Cross Curve

$n = 2/3$ Evolute of a central conic (Astroid if $a = b$)

$n = 5/2$ 'Super-Ellipse' (discovered by Piet Hein).

Tangents: $b\,x\cos\theta\,\sin^{\frac{2}{n}-1}\theta + a\,y\sin\theta\,\cos^{\frac{2}{n}-1}\theta - a\,b\sin^{\frac{2}{n}-1}\cos^{\frac{2}{n}-1} = 0.$

Parameter form of Lamé-Curves:

#1: $\left[a\cdot \cos(\theta)^{2/n},\ b\cdot \sin(\theta)^{2/n}\right]$

#2: $\left[f(\theta) := a\cdot \cos(\theta)^{2/n},\ g(\theta) := b\cdot \sin(\theta)^{2/n}\right]$

the tangents:

#3: $\left(\frac{d}{d\theta} g(\theta)\right)\cdot x - \left(\frac{d}{d\theta} f(\theta)\right)\cdot y - f(\theta)\cdot\frac{d}{d\theta} g(\theta) + g(\theta)\cdot\frac{d}{d\theta} f(\theta)$

#4: $\cos(\theta)^{(2-n)/n}\cdot\left(\frac{2\cdot a\cdot y\cdot \sin(\theta)}{n} - \frac{2\cdot a\cdot b\cdot \sin(\theta)^{(2-n)/n}}{n}\right) + \frac{2\cdot b\cdot x\cdot \sin(\theta)^{(2-n)/n}\cdot \cos(\theta)}{n}$

#5: $\mathrm{SOLVE}\left(\cos(\theta)^{(2-n)/n}\cdot\left(\frac{2\cdot a\cdot y\cdot \sin(\theta)}{n} - \frac{2\cdot a\cdot b\cdot \sin(\theta)^{(2-n)/n}}{n}\right) + \frac{2\cdot b\cdot x\cdot \sin(\theta)^{(2-n)/n}\cdot \cos(\theta)}{n},\ y\right)$

#6: $y = b\cdot \sin(\theta)^{2\cdot(1-n)/n} - \frac{b\cdot x\cdot \sin(\theta)^{2\cdot(1-n)/n}\cdot \cos(\theta)^{2\cdot(n-1)/n}}{a}$

the respective DERIVE-function:

#7: $y = \mathrm{PARA\_TANGENT}\left(\left[a\cdot \cos(t)^{2/n},\ b\cdot \sin(t)^{2/n}\right],\ t,\ \theta,\ x\right)$

#8: $y = b\cdot \sin(\theta)^{2\cdot(1-n)/n} - \frac{b\cdot x\cdot \sin(\theta)^{2\cdot(1-n)/n}\cdot \cos(\theta)^{2\cdot(n-1)/n}}{a}$

#9: $a\cdot y = a\cdot b\cdot \sin(\theta)^{2\cdot(1-n)/n} - b\cdot x\cdot \sin(\theta)^{2\cdot(1-n)/n}\cdot \cos(\theta)^{2\cdot(n-1)/n}$

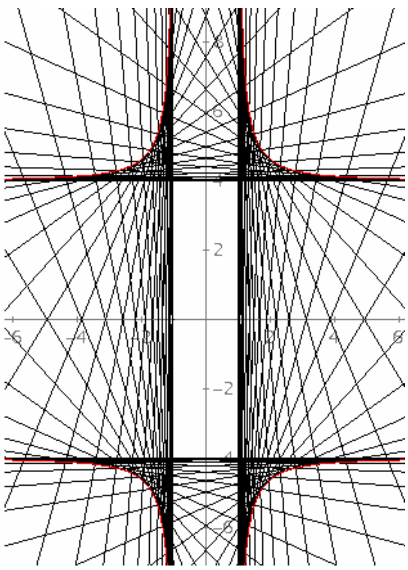
take n = -2, a = 1 and b = 4 (Cross Curve)

#10: $y = \frac{4}{\sin(\theta)^3} - 4\cdot x\cdot \cot(\theta)^3$

#11: $\mathrm{VECTOR}\left(y = \frac{4}{\sin(\theta)^3} - 4\cdot x\cdot \cot(\theta)^3,\ \theta,\ 0,\ 2\cdot\pi,\ \frac{\pi}{50}\right)$

#12: $\left[1\cdot \cos(\theta)^{2/(-2)},\ 4\cdot \sin(\theta)^{2/(-2)}\right]$

#13: $\left[\frac{1}{\cos(\theta)},\ \frac{4}{\sin(\theta)}\right]$



take n = 5/2, a = 5, b = 2

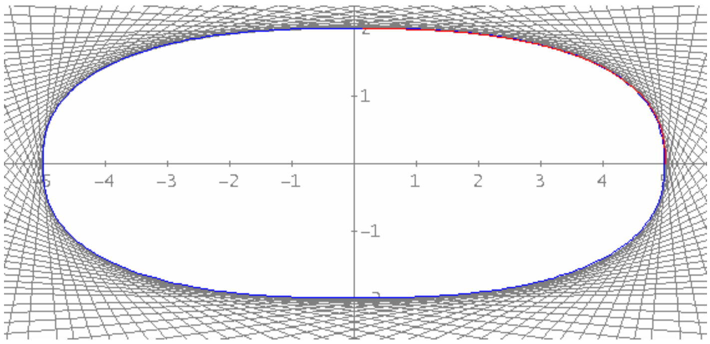
$$\#14:\quad y = 2\cdot SIN(\theta)^{2\cdot(1-5/2)/(5/2)} - \frac{2\cdot x\cdot SIN(\theta)^{2\cdot(1-5/2)/(5/2)}\cdot COS(\theta)^{2\cdot(5/2-1)/(5/2)}}{5}$$

$$\#15:\quad y = \frac{2}{SIN(\theta)^{6/5}} - \frac{2\cdot x\cdot COS(\theta)^{6/5}}{5\cdot SIN(\theta)^{6/5}}$$

$$\#16:\quad VECTOR\left(y = \frac{2}{SIN(\theta)^{6/5}} - \frac{2\cdot x\cdot COS(\theta)^{6/5}}{5\cdot SIN(\theta)^{6/5}},\ \theta,\ 0,\ 2\cdot\pi,\ \frac{\pi}{50}\right)$$

$$\#17:\quad \left[5\cdot COS(\theta)^{4/5},\ 2\cdot SIN(\theta)^{4/5}\right]$$

This gives only one quarter of the curve. The whole curve is defined by the absolute value (see # 21)



$$\#18:\quad VECTOR\left(y = \frac{2}{SIN(\theta)^{6/5}} + \frac{2\cdot x\cdot COS(\theta)^{6/5}}{5\cdot SIN(\theta)^{6/5}},\ \theta,\ 0,\ 2\cdot\pi,\ \frac{\pi}{50}\right)$$

$$\#19:\quad VECTOR\left(y = -\frac{2}{SIN(\theta)^{6/5}} + \frac{2\cdot x\cdot COS(\theta)^{6/5}}{5\cdot SIN(\theta)^{6/5}},\ \theta,\ 0,\ 2\cdot\pi,\ \frac{\pi}{50}\right)$$

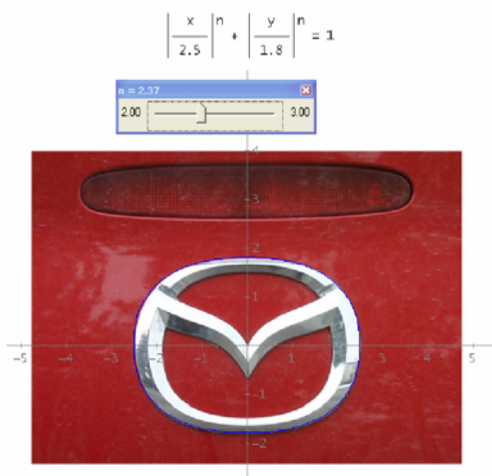
$$\#20:\quad VECTOR\left(y = -\frac{2}{SIN(\theta)^{6/5}} - \frac{2\cdot x\cdot COS(\theta)^{6/5}}{5\cdot SIN(\theta)^{6/5}},\ \theta,\ 0,\ 2\cdot\pi,\ \frac{\pi}{50}\right)$$

$$\#21:\quad \left|\frac{x}{5}\right|^{5/2} + \left|\frac{y}{2}\right|^{5/2} = 1$$

Many years ago when I was busy with background pictures and among others tried modelling various comapany's symbols I took the picture of a Mazda sign.

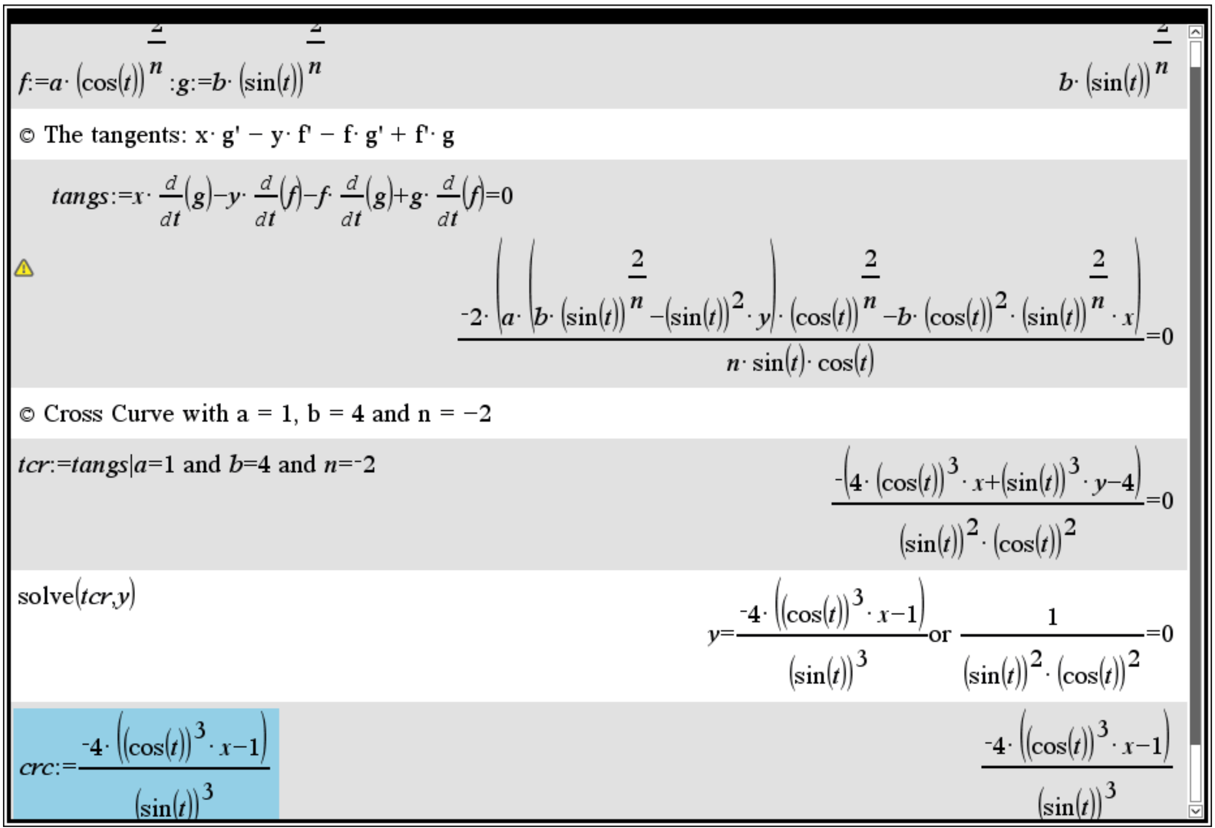
I had the idea to start with an ellipse and then change the parameters.

Using a slider for the exponent I found a curve which fit excellent. Then I was very surprised to discover this "Super Ellipse" as a Lamé Curve in Wikipedia ...



Josef

See the Cross Curve produced with TI-Nspire:



$f:=a\cdot(\cos(t))^{\frac{2}{n}} : g:=b\cdot(\sin(t))^{\frac{2}{n}}$ $\quad b\cdot(\sin(t))^{\frac{2}{n}}$

© The tangents: x· g' − y· f' − f· g' + f'· g

$tangs:=x\cdot\frac{d}{dt}(g)-y\cdot\frac{d}{dt}(f)-f\cdot\frac{d}{dt}(g)+g\cdot\frac{d}{dt}(f)=0$

$$\frac{-2\cdot\left(a\cdot\left(b\cdot(\sin(t))^{\frac{2}{n}}-(\sin(t))^{2}\cdot y\right)\cdot(\cos(t))^{\frac{2}{n}}-b\cdot(\cos(t))^{2}\cdot(\sin(t))^{\frac{2}{n}}\cdot x\right)}{n\cdot\sin(t)\cdot\cos(t)}=0$$

© Cross Curve with a = 1, b = 4 and n = −2

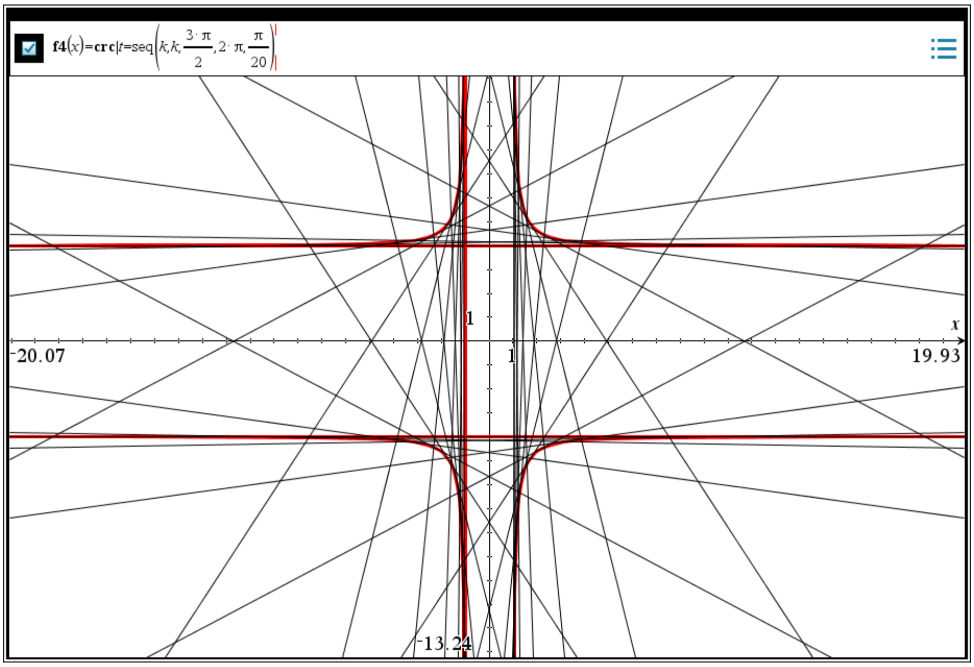
$tcr:=tangs|a=1 \text{ and } b=4 \text{ and } n=^{-}2$

$$\frac{-\left(4\cdot(\cos(t))^{3}\cdot x+(\sin(t))^{3}\cdot y-4\right)}{(\sin(t))^{2}\cdot(\cos(t))^{2}}=0$$

solve(tcr,y)

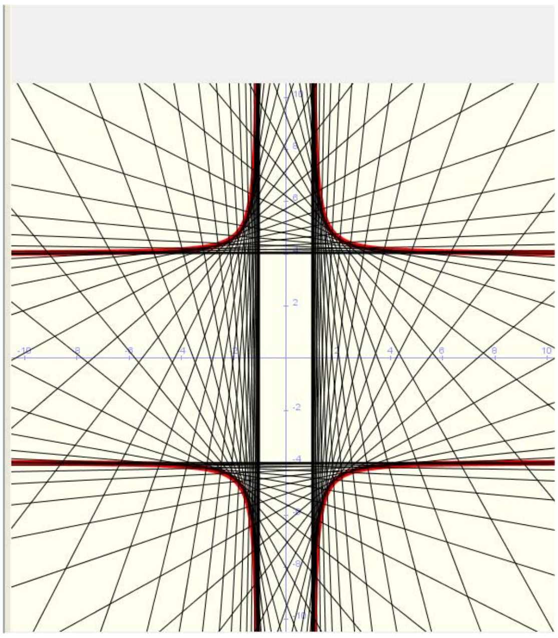
$$y=\frac{^{-}4\cdot\left((\cos(t))^{3}\cdot x-1\right)}{(\sin(t))^{3}} \text{ or } \frac{1}{(\sin(t))^{2}\cdot(\cos(t))^{2}}=0$$

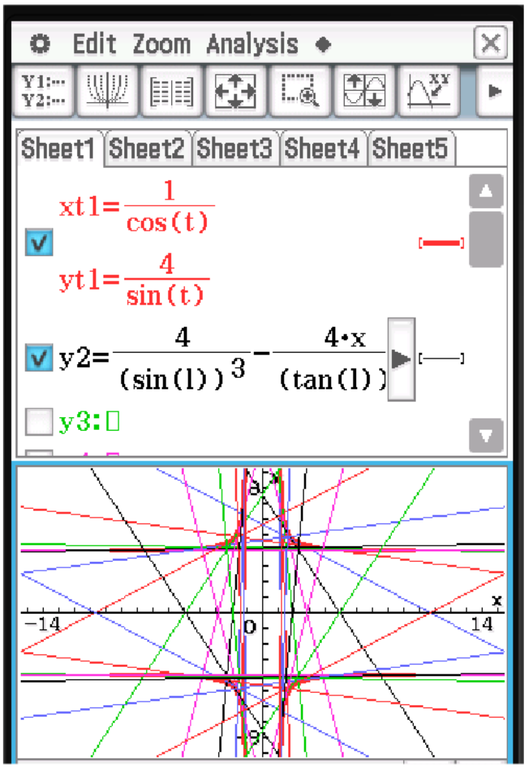
$crc:=\frac{^{-}4\cdot\left((\cos(t))^{3}\cdot x-1\right)}{(\sin(t))^{3}}$ $\quad \frac{^{-}4\cdot\left((\cos(t))^{3}\cdot x-1\right)}{(\sin(t))^{3}}$



Calculation is no problem. Plotting is not so easy because of the horizontal and vertical straight lines appearing in the graph of the Cross Curve. Additionally I have to split the family of tangents. Plotting many tangents (with *t* from 0 to 2π step π/20) does not work properly.

```
plot((x/1)^-2+(y/4)^-2=1,{color=red,line_width=6}) → plotter1
tangs:={y= 4/(sin(t))^3-4x·(cotan(t))^3 with t in 0.1..2π..π/50}
  → {y= 4/sin(t)^3 -4·x·cotan(t)^3 with t in 0.1..2·π..π/50}
for i in 1..length(tangs) do
        plot(tangs_i, {color=gray,line_width=1}) → plotter1
end
```





Let's compare how other systems are performing:

WIRIS (above) presents a fine graph of cross curve and tangents as well. As you can see I started with t = 0.1 because WIRIS refuses plotting for t = 0 (division by zero!)

Left hand side is a screen shot of ClassPad. ClassPad does not provide the cotangens function, the values for t are provided in a separate list.

All tangents can here be plotted in a single step – but I could not give them all the same colour!

e.g. 5: Nephroid (on surface of coffee in cup)

Let the inner surface of the cup be represented by the unit circle and let the incident rays be parallel to the x-axis. If a ray is incident on the cup at the point P,(cos θ, sin θ), then since the angle of reflection is equal to the angle of incidence, the equation to the reflected ray from P is

$$(y-\sin\theta)\cdot\cos 2\theta=(x-\cos\theta)\cdot\sin 2\theta$$

which is the faily of all reflected rays, with θ as the parameter.

The envelope of this family has the general name, *catacaustic*.

The equation of the envelope of a one-parameter family of curves $f(x,y,\theta)=0$ is found by eliminating the parameter θ from the equations $f=0,\ \frac{\partial f}{\partial\theta}=0$ or by solving for x and y as functions of θ to result in a parametric pair of equations, which, in this case, is a nephroid.

viz:- $$x=\cos\theta-\frac{\cos\theta\cos 2\theta}{2},\quad y=\sin\theta-\frac{\cos\theta\sin 2\theta}{2}.$$

Another representation of a nephroid as an envelope

$$x = a\,(3\cos\theta - \cos 3\theta),\quad y = a\,(3\sin\theta - \sin 3\theta) \qquad \text{Parametric}$$

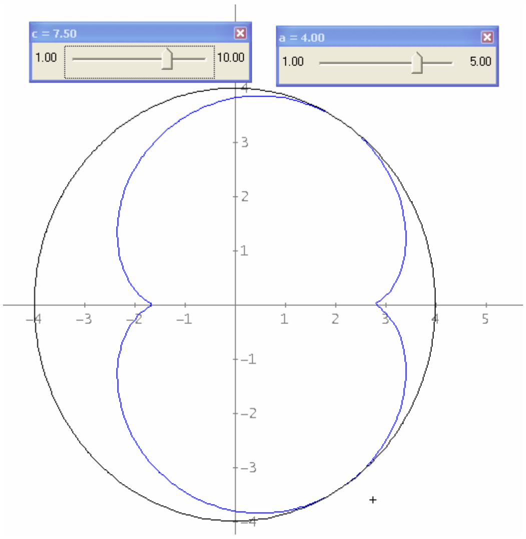
$$(x^2 + y^2 - 4a^2)^3 = 108a^4y^2 \qquad \text{Cartesian}$$

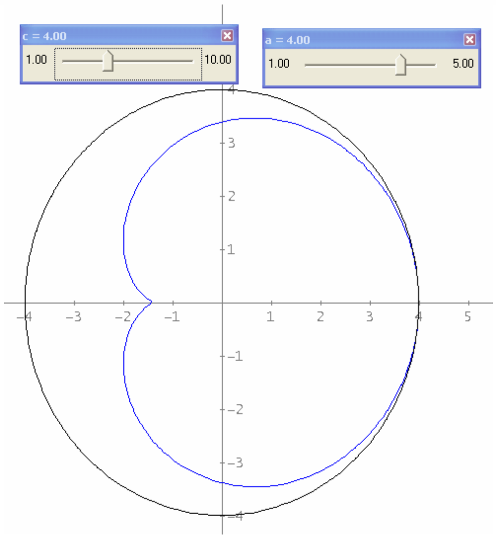
$$\left(\frac{r}{2a}\right)^{\frac{2}{3}} = \left(\sin\frac{\theta}{2}\right)^{\frac{2}{3}} + \left(\cos\frac{\theta}{2}\right)^{\frac{2}{3}} \qquad \text{Polar}$$

Tangents: $x(\cos\theta - \cos 3\theta) + y(\sin\theta - \sin 3\theta) - 8a^2\cdot\sin^2\theta = 0$

$$((4\cdot c^2 - a^2)\cdot(x^2 + y^2) - 2\cdot a^2\cdot c\cdot x - a^2\cdot c^2)^3 - 27\cdot a^4\cdot c^2\cdot y^2\cdot(x^2 + y^2 - c^2)^2 = 0$$

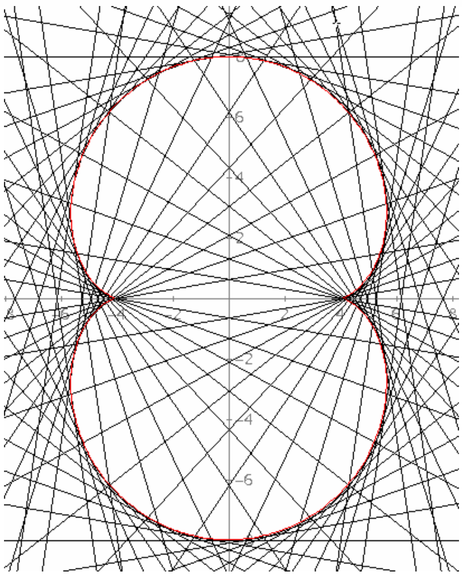
This is supposed to be a catacaustic of a circle x^2 + y^2 = a^2, where the radiant point of the light is at (c,0).





$$(x^2 + y^2 - 16)^3 = 108\cdot 16\cdot y^2$$

$$\text{VECTOR}\left(\frac{x\cdot\text{COS}(3\cdot\theta) - x\cdot\text{COS}(\theta) + 16\cdot\text{SIN}(\theta)^2}{\text{SIN}(\theta) - \text{SIN}(3\cdot\theta)},\ \theta,\ \frac{\pi}{40},\ 2\cdot\pi,\ \frac{\pi}{40}\right)$$



CATASTROPHE THEORY

There is also a catastrophic approach to the reflection in the coffee cup

One can ignore the wave nature of the light and merely consider the energy being transported along the light rays. The intensity of the light is inversely proportional to the cross-section area of a pencil of light rays.

Initially, the mathematics is different, but the final result is precisely as above.

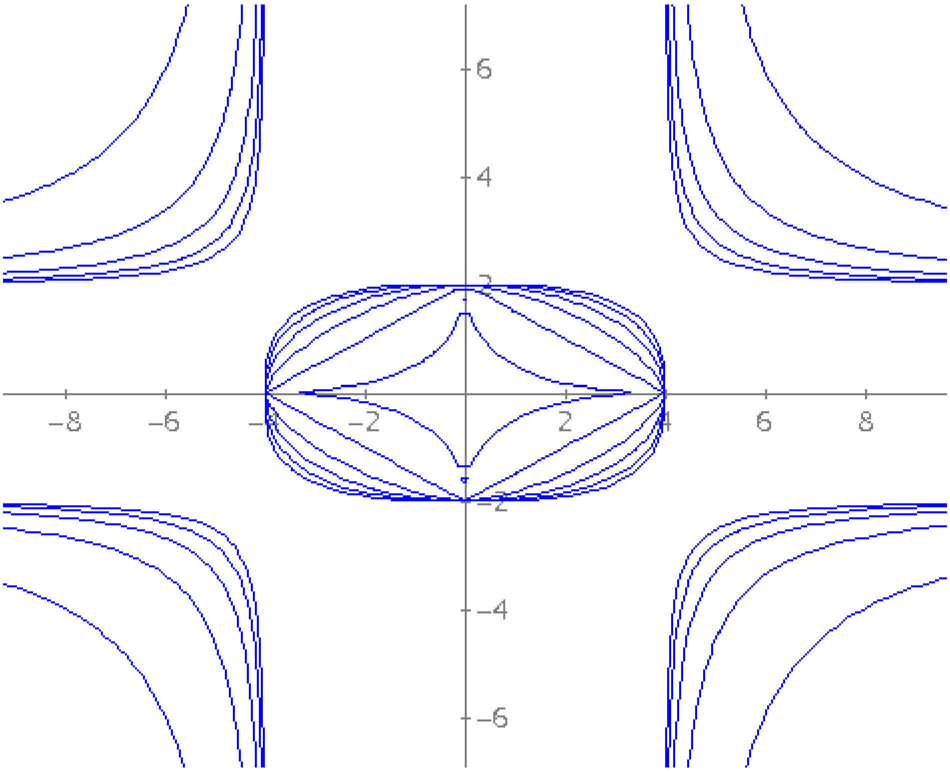
The world of Lamé Curves

It is interesting investigating the various forms of Lamé Curves. You can either plot the family of them varying the exponent (graph is below) or you introduce a slider for the exponent and inspect how the form is changing. Josef

#1: $\text{lame}(a, b, n) := \left|\left(\frac{x}{a}\right)^n\right| + \left|\left(\frac{y}{b}\right)^n\right| = 1$

#2: $\text{VECTOR}\left(\text{lame}(4, 2, k), k, -3, 3, \frac{1}{2}\right)$

#3: lame(4, 2, n)

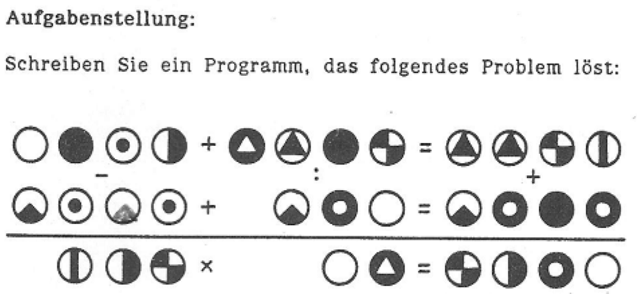


Twister 02 – Da raucht der Kopf [1] – The Head Nearly Splits

Josef Böhm, Würmla, Austria

This “Logical” inspired me to solve logic problems CAS-assisted. I found many nice problems in books and journals. This Brain Twister was the initiator for one of ma TIME 2014 Workshops.

I enjoyed collecting and solving various provlems enormously.



Function perm(n,k) (Page 15) is loaded as expression #1.

The numbers and their relations (equations) are defined (by characters instead of the symbols):

```
#2:  [z1 := 1000·i + 100·j + 10·k + l, z2 := 1000·m + 100·n + 10·j + o]
#3:  [z3 := 1000·n + 100·n + 10·o + p, z4 := 1000·q + 100·k + 10·q + k]
#4:  [z5 := 100·q + 10·r + i, z6 := 1000·q + 100·r + 10·j + r]
#5:  [z7 := 100·p + 10·l + o, z8 := 10·i + m, z9 := 1000·o + 100·l + 10·r + i]
#6:  [eq1 := z1 + z2 - z3, eq2 := z4 + z5 - z6, eq3 := z7·z8 - z9]
#7:  [eq4 := z1 - z4 - z7, eq5 := z2 - z5·z8, eq6 := z3 + z6 - z9]
```

all is the list of all 10! permutations of the 10 different characters (=3 628 800), The characters are assigned to the elements of the permutations to be generated and then to be investigated.

```
#8:  all := perm([0, 1, 2, 3, 4, 5, 6, 7, 8, 9], 10)
#9:  [i := v_1, j := v_2, k := v_3, l := v_4, m := v_5]
#10: [n := v_6, o := v_7, p := v_8, q := v_9, r := v_10]
#11: (SELECT(eq1 = 0 ∧ eq2 = 0 ∧ eq3 = 0 ∧ eq4 = 0 ∧ eq5 = 0 ∧ eq6 = 0, v, all))_1
#12: needed 2700 seconds calculation time
#13: [2, 0, 7, 5, 4, 6, 8, 3, 1, 9]
#14: v := [2, 0, 7, 5, 4, 6, 8, 3, 1, 9]
#15: [z1, z2, z3, z4, z5, z6, z7, z8, z9]
#16: [2075, 4608, 6683, 1717, 192, 1909, 358, 24, 8592]
```

Solution:

2075	+	4608	=	6683
−		÷		+
1717	+	192	=	1909
358	×	24	=	8592

```
#17:  we help by decreasing the number of unknowns by varying i from 1 to 9:

#18:  v :=

       ⎡i := 1, j := v , k := v , l := v , m := v ⎤
#19:   ⎣               1        2        3        4⎦

       ⎡n := v , o := v , p := v , q := v , r := v ⎤
#20:   ⎣      5        6        7        8        9⎦

#21:  all := perm([0, 2, 3, 4, 5, 6, 7, 8, 9], 9)

      (SELECT(eq1 = 0 ∧ eq2 = 0 ∧ eq3 = 0 ∧ eq4 = 0 ∧ eq5 = 0 ∧ eq6 = 0, v, all))
#22:                                                                            1

      []
#23:    1

#24:  no solution for i = 1

#25:  i := 2

#26:  all := perm([0, 1, 3, 4, 5, 6, 7, 8, 9], 9)

      (SELECT(eq1 = 0 ∧ eq2 = 0 ∧ eq3 = 0 ∧ eq4 = 0 ∧ eq5 = 0 ∧ eq6 = 0, v, all))
#27:                                                                            1

#28:  [0, 7, 5, 4, 6, 8, 3, 1, 9]

#29:  needs 253 seconds only

#30:  v := [0, 7, 5, 4, 6, 8, 3, 1, 9]

#31:  [z1, z2, z3, z4, z5, z6, z7, z8, z9]

#32:  [2075, 4608, 6683, 1717, 192, 1909, 358, 24, 8592]
```

Additional problem: Design a similar problem by your own. Use characters instead of symbols.

The tricky (unintentionally) problem was that there is a typo in the book. You can see in the scan that I corrected one symbol in the second row by hand. Fortunately the solution was given in the book. The author provided some more or less extended programs in BASIC, Pascal and C to solve the problems (it was from 1989!!).

Similar problems are so called "*Alphametics*" which can be found in many variations in a column of the weekly German newspaper *Die Zeit*.

Example: Try to solve MAI + JUNI + JULI = ALPIN (*Die Zeit,* Mai 2014)

I finished my workshop saying:

I don't intend to substitute logical reasoning by trial & error methods.

I believe that it can bring another quality into the solving process when forming a mathematical model using mathematical language and expressions.

[1] Gerd Kebschull, *Computer Knobeleien*, Heise1989

I have a collection of more than 50 Brain Twisters (German and English) which can be downloaded with the DLN9495-files.

*In cryptography, a **Caesar cipher**, also known as **Caesar's cipher**, the **shift cipher**, **Caesar's code** or **Caesar shift**, is one of the simplest and most widely known encryption techniques. It is a type of substitution cipher in which each letter in the plaintext is replaced by a letter some fixed number of positions down the alphabet.* (*Wikipedia*)

Günter Schödl changes the encoding algorithm by multiplying the code of every character by a fixed value instead of adding it. This procedure does not make the cipher more secure – but more interesting from the programming point of view. Josef

(At http://www.guenter-schoedl.at/informatik/basics/caesar-multiplikativ.htm you can find the original contribution in German, Josef).

Caesar Multiplication

Günter Schödl, Austria

For improving the encoding one can multiply each single code by a fixed value (*fa*) instead of adding one. As one needs the inverse function for decoding it is necessary to use as module (*mo*) a prime number which includes the complete set of characters (codes).

Encoding:
The character with code *x* is encoded according to `f(x):=MOD(fa*x,mo)`. In order to shift it back into the used font set we add the code corresponding with the first character of the font set (*kor*). This gives our encoding function

```
f(x):=MOD(fa*x,mo)+kor
```

Example: Take the alphabet which starts with A (code 65) and comprises 26 characters. We choose a prime greater 26, e.g. 31, as module. Factor *fa* shall be 4 and we adjust with *kor* = 65.

```
f(x) := MOD(4·x, 31) + 65
f(65) = 77
A → M
```

Decoding:
For decoding we need the inverse modular function which is of form $g(x):=a*x + b$. Coefficient a is obtained by

```
mo:=
fi:=INVERSE_MOD(fa,mo)
```

b is found by the equation:

```
MOD(fi * f(x) + b, mo) + kor = x
```

This gives in our example:

```
fi := INVERSE_MOD(f(a), mo)
INVERSE_MOD(4, 31) = 8
SOLUTIONS(MOD(8·77 + z, 31) + 65 = 65, z) = [4, -27, -58]
```

This is in all cases the residual class 4!
Then the decoding function can be defined as

```
g(x) := MOD(8·x + 4, 31) + 65
g(77) = 65
M → A
```

We write a function which encodes a whole string and test it:

```
Caesar1(string) := CODES_TO_NAME(VECTOR(f(i), i, NAME_TO_CODES(string)))
Caesar1(Franz) = BWQGX
```

Decoding the encoded string shows the problem: the module comprises the upper case letters only! (All the strings are under quotes!)

```
KlarCaesar1(string) := CODES_TO_NAME(VECTOR(g(i), i, NAME_TO_CODES(string)))
KlarCaesar1(BWQGX) = FSBO[
```

Encoding FRANZ and then decoding works correctly:

```
Caesar1(FRANZ) = BSMCT
KlarCaesar1("BSMCT") = "FRANZ"
```

In order to include the lower case letters, too, we adjust both functions:

```
factor = 3, mo = 31

f2(x) :=
  If x < 97
     MOD(3·x, 31) + 65
     MOD(3·x, 31) + 96

f2(65) = 74

INVERSE_MOD(3, 31) = 21

SOLUTIONS(MOD(21·74 + z, 31) + 65 = 65, z) = [-4, 27, -35]

factor = 21, konst = 27, mo = 31

g2(x) :=
  If x < 97
     MOD(21·x + 27, 31) + 65
     MOD(21·x + 27, 31) + 96

Caesar2(string) := CODES_TO_NAME(VECTOR(f2(i), i, NAME_TO_CODES(string)))

KlarCaesar2(string) := CODES_TO_NAME(VECTOR(g2(i), i, NAME_TO_CODES(string)))

Caesar2(Franz FRANZ) = YaltyDY^JRW

KlarCaesar2(YaltyDY^JRW) = Franz^FRANZ

Caesar2(This message is secret.) = EbedDqxddl~xDedDdxraxgO

KlarCaesar2(EbedDqxddl~xDedDdxraxgO) = This^message^is^secretM
```

There is one problem left: encoding and decoding the characters with ASCII Code between 31 and 65, e.g. the space, comma, colon, ...

Space and full stop are treated incorrectly! We solve this problem by choosing an appropriate value for *kor* and together with a module which is great enough. So we receive a generalized encoding and decoding function finally:

For encoding:

```
f3(x, fa, mo, kor) :=
  If x < mo + kor
     MOD(fa·x, mo) + kor
     MOD(fa·x, mo) + kor + mo
```

For decoding:

```
MOD(3·65, 97) + 32 = 33

INVERSE_MOD(3, 97) = 65

SOLUTIONS(MOD(65·33 + z, 97) + 32 = 65, z) = [22, -75, 119]

g3(x, fa, konst, mo, kor) :=
  If x < mo + kor
     MOD(fa·x + konst, mo) + kor
     MOD(fa·x + konst, mo) + kor + mo

For encoding: fa = 3, mo = 97, kor = 32

For decoding: fa = 65, konst = 22, mo = 97, kor = 32

Caesar3(string, fa, mo, kor) := CODES_TO_NAME(VECTOR(f3(i, fa, mo, kor), i,
NAME_TO_CODES(string)))

KlarCaesar3(string, fa, konst, mo, kor) := CODES_TO_NAME(VECTOR(g3(i, fa, konst,
mo, kor), i, NAME_TO_CODES(string)))
```

Example: We encode a five lines text which is saved as variable *text*.

```
text := Zunächst schreibt man das normale Alphabet auf- das ist das Klartextal-
phabet. Darunter schreibt man nochmals das Alphabet das Geheimtextalphabet. Mit
diesem fangt man jedoch nicht unter dem A an, sondern unter einem beliebigen
Buchstaben. Man schreibt das Alphabet, bis man unter dem Buchstaben Z angelangt
ist, und schreibt den Rest vorne hin
```

We choose *fa* = 3 (multiplication factor), *mo* = 97 (width of the font set), *kor* = 32 (start of the font set) as given above.

```
a := Caesar3(text, 3, 97, 32)

a := 1\GzOO,[&5VY□V&5S,8#Y□D G□) V□GJSD A,□!AM5 #,Y□ \/F□) V□8VY□) V□?A SY,eY AM5
#,YI□* S\GY,S□V&5S,8#Y□D G□GJ&5D AV□) V□!AM5 #,Y□) V□3,5,8DY,eY AM5
#,YI□E8Y□)8,V,D□/ G2Y□D G□;,)J&5□G8&5Y□\GY,S□),D□!□

KlarCaesar3(a, 65, 22, 97, 32)

Zunächst schreibt man das normale Alphabet auf- das ist das Klartextalphabet.
Darunter schreibt man nochmals das Alphabet das Geheimtextalphabet. Mit diesem
fangt man jedoch nicht unter dem A an, sondern unter einem beliebigen Buchsta-
ben. Man schreibt das Alphabet, bis man unter dem Buchstaben Z angelangt ist,
und schreibt den Rest vorne hin.
```

We add a second example with an English text to encode and decode:

For encoding we choose: *fa* = 5, *mo* = 101, *kor* = 32

```
text2 := First of all we write down the common alphabet - this is the clear al-
phabet. Then write the secret alphabet below but don't start below the A, but
under any other letter ...

aa := Caesar3(text2, 5, 101, 32)

aa := O4afk[R%[qCC[z [za4k [□RzM[k/ [{RHHRM[qCW/qv k[7[k/4f[4f[k/ [{C qa[qCW/qv
k<[O/ M[za4k [k/ [f {a k[qCW/qv k[v CRz[vpk[□RM~k[fkqak[v CRz[k/ [62[vpk[pM□
a[qM□[Rk/ a[C kk a[<<<
```

```
MOD(5·65, 101) + 32 = 54

INVERSE_MOD(5, 101) = 81

SOLUTIONS(MOD(81·54 + z, 101) + 32 = 65, z) = [2, -99, 103]
```

For decoding we need: *fa* = 81, *konst* = 2, *mo* = 101, *kor* = 32

```
KlarCaesar3(aa, 81, 2, 101, 32)

First of all we write down the common alphabet – this is the clear alphabet.
Then write the secret alphabet below but don't start below the A, but under any
other letter ...
```

We could investigate the Caesar Cipher combining multiplication and addition in a similar way.

Colour Gradient and LUA Scripts with TI-Nspire

Alfred Roulier, Switzerland

Der Reiz von Julia-Bildern liegt in den schönen Formen und reichen Farben dieser Gebilde. Seit beim TI-Nspire die Möglichkeit für LUA-Skripte gegeben ist, kann man damit nun ebenfalls farbige Punktgrafiken erstellen.

Ein Julia-Bild wird wie folgt erstellt :
In der komplexen Zahlenebene wählt man einen Punktbereich (ein Rechteck) und gibt eine konstante Zahl c vor. Bei jedem Punkt z im Bereich führt man nun die Rekursion $z_{n+1} = z_n^2 + c$ aus und prüft dabei, ob $z_{n+1}^2 > 2$ geworden ist. Wenn dies geschieht, hält man die Rekursionstiefe k fest und bricht die Rekursion ab. Wenn k eine gegebene Grenze erreicht, z.B. 200, bricht man ebenfalls ab. Solche Punkte heissen Gefangenenpunkt und werden schwarz gezeichnet. Die anderen sind Fluchtpunkte und werden in Funktion von k gefärbt. Wenn man in das Bild hineinzoomt, sieht man, wie sich die filigranen Strukturen immer weiter verästeln – das ist die fraktale Eigenschaft dieser Punktmenge.

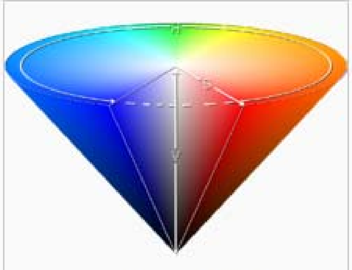
The fascination of Julia graphs lies in their beautiful forms and the rich colours of these objects. Since LUA scripts are possible with TI-Nspire coloured point graphs can be produced (in a reasonable time).

We choose a region in the complex plane (usually a rectangle) and a constant (complex) number *c*. For every point *z* within the region the recursion $z_{n+1}=z_n^2 + c$ is performed checking if $z_{n+1} > 2$. If so, the recursion depth *k* is fixed and the recursion is stopped. If *k* becomes greater than a certain boundary, e.g. 200, then we also stop and plot the initial point black ("prisoner point"). The other points are called "escape points" and are coloured depending on the value of *k*. Zooming in gives always new graphs – which is the fractal nature of this point set.

Farbverlauf / Colour Gradient

Es geht also darum, den Farbton der Punkte in Funktion der Iterationstiefe k zu berechnen. Dies geschieht am besten im HSV Farbraum. (Hue = Farbton, Saturation = Sättigung, Value = Dunkel-/Hellwert))

The problem is to calculate the colour shade as a function of the number of iterations *k*. This can be performed in the best way in the HSV colour model (Hue, Saturation, Value).



Farbwert *H*, Sättigung *S*, Dunkelstufe *V*

Farbtonskala

In diesem Wikipedia-Bild ist das HSV-Konzept erklärt.
Der Farbton läuft von 0 bis 360 Grad.
Sättigung und Dunkelstufe von 0 bis 1.
Wir wählen ein Farbtonfenster im Bereich H_1 bis H_2 und unterteilen diesen in k_{max} Schritte (k_{max} = Max Iterationstiefe).
Die Sättigung wählen wir 1.
Zur Steigerung des Kontrast wählen wir für gerade k eine Dunkelstufe von z.B. 0.5 und für ungerade k die Stufe 1.

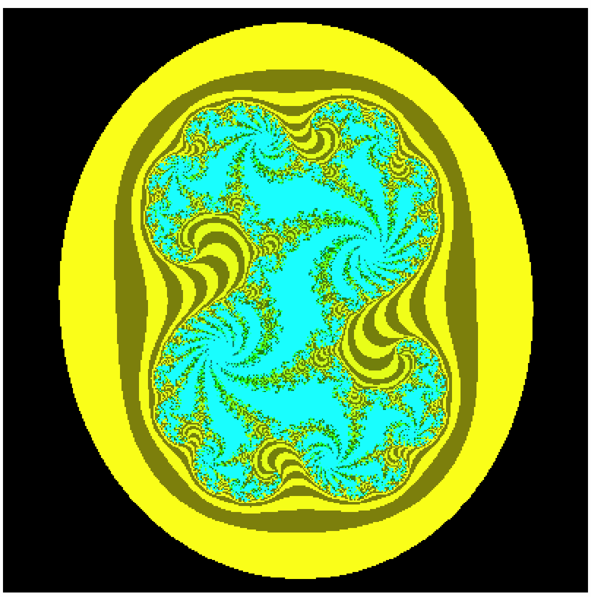
0° ≤ Hue ≤ 360°, we choose H_1 ≤ hue ≤ H_2 and divide into k_{max} = max iterations.
Saturation = 1, value = 0.5 for even *k* and 1 else.

LUA arbeitet aber im RGB-Farbraum, weshalb eine Umrechnung HSV → RGB notwendig wird. Den Algorithmus findet man in https://de.wikipedia.org/wiki/HSV-Farbraum und ist im Anhang im Skript einzusehen.
LUA supports the RGB color model, which makes necessary to convert from HSV to RGB. The algorithm can be found in http://en.wikipedia.org/wiki/HSL_and_HSV#From_HSV.

Programmierung

Die Berechnung der k-Wert-Matrix in einem TI-Nspire-Programm und der anschliessende Transfer in ein Bild via LUA beansprucht bei 100 x 100 Punkten ca 45 Sekunden.

Wenn sowohl die Berechnung der Punktwerte (ohne Zwischenspeicherung als Matrix) und Bild im LUA Skript programmiert sind, geht es wesentlich schneller. 400 x 400 Punkte werden in einer knappen Minute dargestellt. LUA Skript im Anhang.



Indem die Helligkeit laufend zwischen zwei Werten pendelt, wird ein besserer Kontrast erzielt.

Eingaben	
qd:=1 ▸ 1	Seitenlänge Bildpunktquadrat in Pixel
npt:=400 ▸ 400	Maximale Anzahl Punkte pro Dimension
bereich:=1.6 ▸ 1.6	Bildbereich
tiefe:=200 ▸ 200	Iterationstiefe
cre:=0.32 ▸ 0.32	Realteil der Konstanten c
cim:=0.043 ▸ 0.043	Imaginärteilteil der Konstanten c
z0re:=0 ▸ 0	Realteil der Bildmitte
z0im:=0 ▸ 0	Imaginärteil der Bildmitte
Farben	
f1:=60 ▸ 60	Untere Farbtongrenze in Grad (HSV)
f2:=180 ▸ 180	Obere
s:=0.9 ▸ 0.9	Sättigung
v1:=127.5 ▸ 127.5	kleine Helligkeit
v2:=255 ▸ 255	grosse Helligkeit

Calculation of the *k*-values matrix by a Nspire program followed by the transfer into a LUA-generated picture needs about 45 seconds for a 110 x 100 points rectangle.
Calculating the *k*-values also within the LUA script (without intermediate storing the matrix) works much faster 440 x 400 points needs one minute.

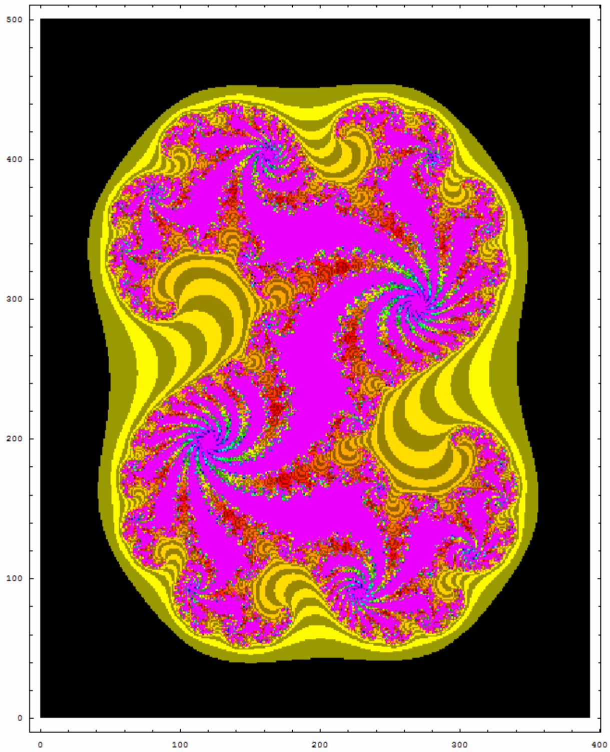
Mathematica bietet weitere Vorteile / Mathematica offers some advantages:

- es rechnet noch schneller / calculation is faster
- es verfügt über den Befehl „DensityPlot“ / a command “Density Plot“ is available
- die Farbgebung erfolgt im HSV-Raum / colouring works within the HSV model

Das Mathematica Programm ist kurz :

```
z0 = 0.01 + .02 * I;
c = 0.32 + 0.043 * I;
npt := 500; tiefe = 301;
seitere = 2.2; seiteim = 2.8;
step = Max[seitere, seiteim] / npt;
werte = Table[{n = 0, z = r0 + q0 * I, While[Abs[z] < 2 && n < tiefe, z = z^2 + c; n++;]}; n,
   {q0, Im[z0] - seiteim / 2, Im[z0] + seiteim / 2, step}, {r0, Re[z0] - seitere / 2, Re[z0] + seitere / 2, step}];
a = .18; farbe[h_] := h - a /; h ≥ a ; farbe[h_] := a - h /; h < a
ListDensityPlot[werte, AspectRatio → Automatic, ColorFunction → (Hue[farbe[#], 1, If[# < .01, 0, 0.6 + 0.4 * Mod[Floor[tiefe * #], 2]]] &), Mesh → False]
```

Die Werte werden in einem einzigen Table-Befehl ermittelt. Zur Bestimmung der Pixelfarbe wird eine „pure function“ verwendet. # ist eine Zahl im Intervall (0,1), abgeleitet aus werte/tiefe. Der Farbgang beginnt bei a = 0.18, d.h. tiefe Werte sind gelb. Die Sättigung ist immer voll. Die Helligkeit liefert Farbe schwarz, wenn werte < 0.01. Darüber pendelt die zwischen 0.6 und 1 je nachdem werte gerade oder ungerade ist.

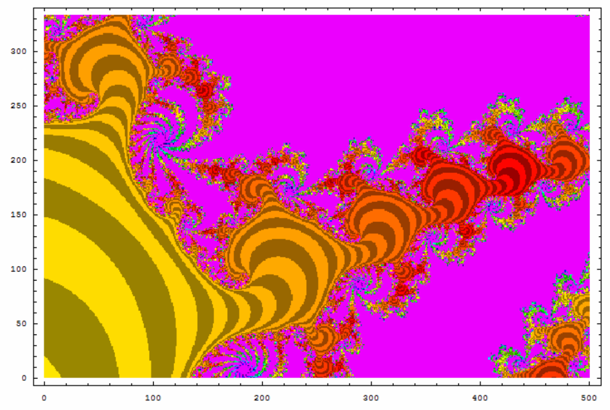


Three *MATHEMATICA* Plots

c = 0.32 + 0.043 ***i***

z = 0

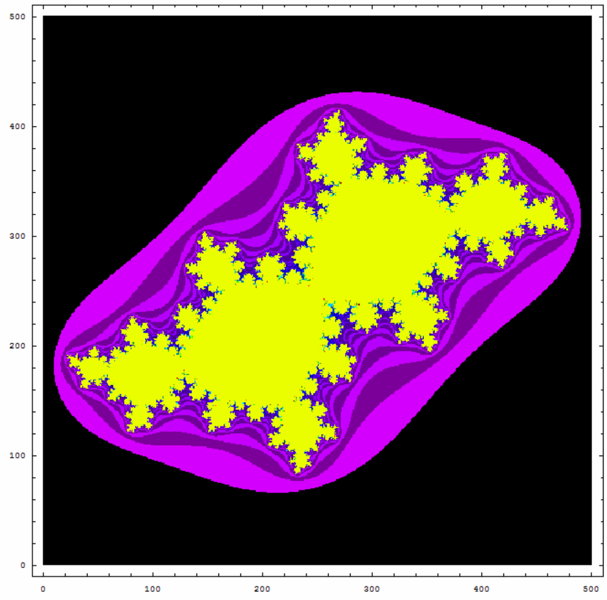
500 Punkte, Tiefe 300



Zoom auf Punkt z0=-0.1+0.4 ***i***

Bildbreiten 0.6 bzw 0.4

500 Punkte, Tiefe 300.



c = -.39054 – 0.58679 ***i***
z = 0
500 Punkte, Tiefe 300

This is the LUA script for calculating and plotting

```
platform.apilevel = '1.0'
-- Schritt 1 : Bildschirmdimensionen h und w in Pixel erfassen
function on.create()
  h=platform.window:height()
  w=platform.window:width()
end
function on.resize(width,height)
  h=height
  w=width
end
-- Figur neu zeichnen, wenn die Enter-Taste gedrückt wird (ein Eingabewert verändert wird)
function on.enterKey()
 platform.window:invalidate()
end
function on.paint(gc)
-- Die Eingabewerte von der Noteseite übernehmen
  qd=(var.recall("qd") or  1)              -- Seitenlänge des Bildpunktquadrats in Pixel
  npt=(var.recall("npt") or  1)            -- maximale Anzal Punkte pro Dimension
  tiefe=(var.recall("tiefe") or 1)         -- Iterationstiefe
  cre=(var.recall("cre") or 1)             -- Realteil der Konstanten c
  cim=(var.recall("cim") or 1)             -- Imaginärteil der Konstanten c
  z0re=(var.recall("z0re") or 1)           -- Realteil des Zentrums des Punktbereichs
  z0im=(var.recall("z0im") or 1)           -- Imaginärteil des Zentrums des Punktbereichs
  bereich=(var.recall("bereich") or 1)     -- Seitenlänge des Punktbereichs(quadrat)
-- Die Anzahl Bildpunkte pro Dimension berechnen
  npt=math.min(npt,math.floor(0.9*h/qd))
-- Die Farbparameter übernehmen
  f1=(var.recall("f1") or 0)               -- untere Grenze des Farbtonbereichs in Grad
  f2=(var.recall("f2") or 360)             -- obere Grenze des Farbtonbereichs in Grad
  s=(var.recall("s") or 1)                 -- Sättigung
```

```
  v1=(var.recall("v1") or .5)              -- unterer Helligkeitswert
  v2=(var.recall("v2") or 1)               -- oberer Helligkeitswert
-- Schlaufe über alle Bereichspunkte
  for j=1,npt do
    for i=1,npt do
      zre1=z0re-bereich+(j-1)*2*bereich/npt ; zim1=z0im+bereich-(i-1)*2*bereich/npt
-- Punktkoordinaten
      wert=tiefe
      for k=1,tiefe do
        zre2=zre1^2-zim1^2+cre              -- Iteration durchführen
        zim2=2*zre1*zim1+cim
        if math.sqrt(zre2^2+zim2^2)>2 then    -- Iteration testen
          wert=k
          break
        end
        zre1=zre2 ; zim1=zim2
      end
-- RGB Farben berechnen
      hh=f1+wert/tiefe*(f2-f1) ; hi=math.floor(hh/60) ; f=hh/60-hi
      v=v1
      if math.fmod(wert,2)==0 then
        v=v2
      end
      p=v*(1-s) ; q=v*(1-s*f) ; t=v*(1-s*(1-f))
      if wert==1 then
        rot=0 ; gruen=0 ; blau=0
      elseif hi==0 then
        rot=v ; gruen=t ; blau=p
      elseif hi==1 then
        rot=q ;gruen=v ; blau=p
      elseif hi==2 then
        rot=p; gruen=v ; blau=t
      elseif hi==3 then
        rot=p ; gruen=q ; blau=v
      elseif hi==4 then
        rot=t ; gruen=p ; blau=v
      elseif hi==5 then
        rot=v ; gruen=p ; blau=q
      end
-- Farbe zuweisen und Quadrat der Seitenlänge qd zeichnen
      gc:setColorRGB(rot,gruen,blau)
      gc:fillRect(w/2-.95*h/2+(j-1)*qd,.05*h+(i-1)*qd,qd,qd)
     end
  end
 end
```

The next pages show the notes page for entering the data connected with the julia() program followed by the program itself and two sample runs of julia().

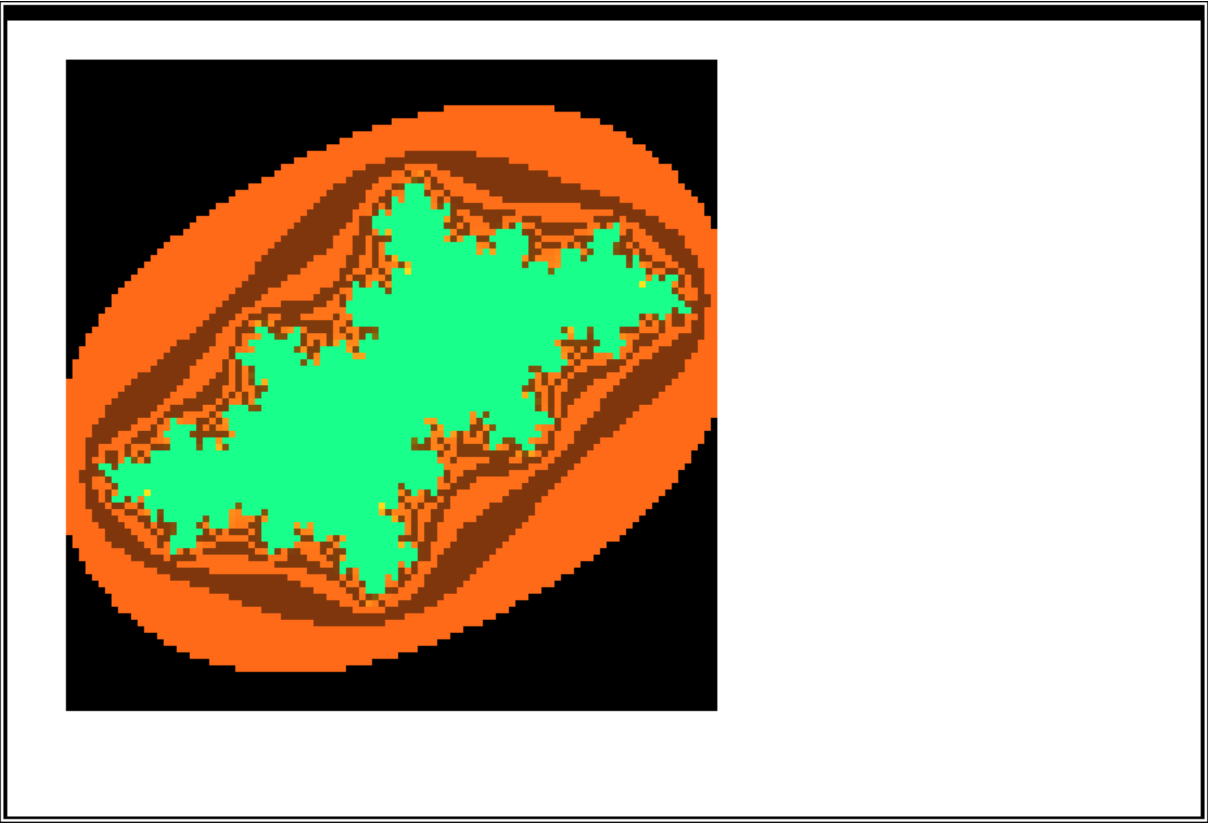
Eingaben / Inputs

c_:=-0.39–0.58·*i* ▸ -0.39–0.58·*i* Konstante in der Iteration $z_{n+1} = z_n + c$
z0_:=0 ▸ 0 Mittelpunkt / Centre
npt:=100 ▸ 100 Anzahl Punkte pro Dimension / points per dimension
seitere:=3 ▸ 3 Bereich Realteil / region real part
seiteim:=3 ▸ 3 Bereich Imaginärteil / region imaginary part
tiefe:=200 ▸ 200 Iterationstiefe / depth of iteration
qd:=5 ▸ 5 Seitenlänge Bildpunktquadrat in Pixel / side of pixel square

julia() ▸ *Done*

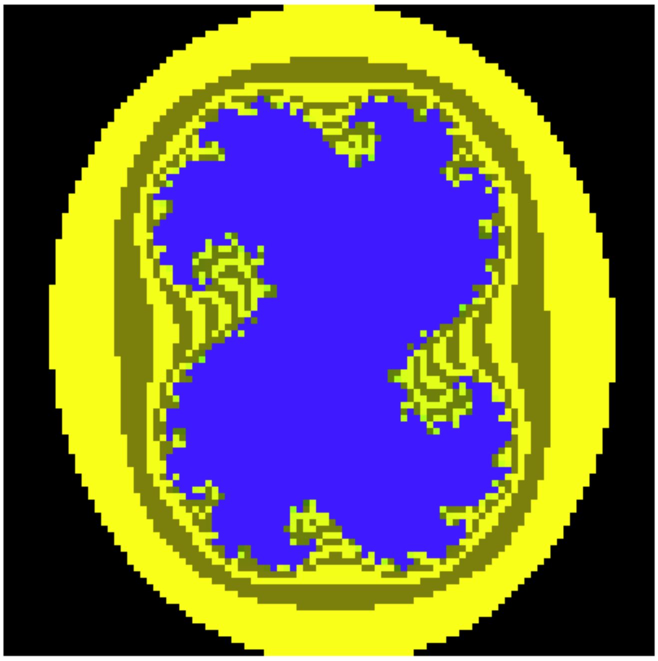
Farben / Colours

f1:=20 ▸ 20 Untere Farbtongrenze in Grad (HSV) / lower HSV boundary
f2:=150 ▸ 150 Obere / upper HSV boundary
s:=0.9 ▸ 0.9 Sättigung / saturation
v1:=127.5 ▸ 127.5 kleine Helligkeit / little brightness
v2:=255 ▸ 255 grosse Helligkeit / strong brightness



julia 7/15

```
Define julia()=
Prgm
Local i,j,k,schrittre,schrittim
wert:=newMat(npt,npt)
schrittre:=seitere/npt:schrittim:=seiteim/npt
For i,1,npt
  For j,1,npt
  wert[i,j]:=tiefe
    z_:=real(z0_)-seitere/2+schrittre·(i-1)+(imag(z0_)+seiteim/2-schrittim·(j-1))·i
    For k,1,tiefe
      z_:=z_^2+c_
      If |z_|>2 Then
        wert[i,j]:=k: Exit
      EndIf
     EndFor
   EndFor
EndFor
EndPrgm
```



Eingaben / Inputs

c_:=0.31+0.043·i ▸ 0.31+0.043·i

z0_:=0 ▸ 0

npt:=100 ▸ 100

seitere:=3 ▸ 3

seiteim:=3 ▸ 3

tiefe:=250 ▸ 250

qd:=5 ▸ 5

julia() ▸ *Done*

Farben / Colours

f1:=60 ▸ 60

f2:=250 ▸ 250

s:=0.9 ▸ 0.9

v1:=127.5 ▸ 127.5

v2:=255 ▸ 255

TIME 2014

Lectures and Workshops

The Proceedings will be published on the website of the Pedagogical University of Lower Austria as soon as possible. I will keep you informed. Some lectures may be added.

Keynotes

Technologiegestützt intelligentes Wissen und Handlungskompetenzen fördern
Technology in Teaching Mathematics: Looking Back and Looking Forward
The Future of Mathematics: A personal View and Comments on Math Education
The International Baccalaureate External Examination Model
The Potential of the Internet for Mathematic Education, MicroLearning, MOOC, OER, ePortfolio and other Virtual Monsters
The Use of DGS and CAS in Proving Theorems

Long Lectures

3D & TI-NspireCAS: What is Available, What is Missing, and How to Adapt
Analyse von Prüfungsaufgaben bezüglich der Rolle der Technologie
Assessment with Access to a Computer Algebra System
CAS and Dynamic Geometry Activities that Integrate Algebra & Geometry: Investigate, Discover, Prove
CASIO Class Pad II-Praxis
CATO – a general User Interface for CAS
Changing How Math is Taught and Learned Using MyMathLab
Conceptualizing a Pedagogical CAS for Algebraic Manipulation of Expressions
Differential Equations and Dynamical Systems, a dynamic approach with TI-NspireCAS
Explorations of Mathematical Models in Life Sciences with Maple
Explorations with the Barycentric Formula for Polynomial Interpolation
Funktionen – immer gut für eine Überraschung
GeoGebra 3D
Gleichförmige Bewegungen – ein Unterrichtskonzept für die 9. Schulstufe
Inspirations from Outside
Integrate iPad Math & Presentation Apps into your 6-12 Math Classes
Maplets for Calculus: A Model for Multi-Use Mathematical Software
Mathcad im Unterricht und bei der neuen Reife- und Diplomprüfung

MathCad in der Industrie

New Ways to Enhance the Classroom Using Symbolic Computation: Automated Assessment, Modeling and Simulation, and More

Online Mathematics for Bachelor Students – it really works!

Prospective Mathematics Teachers' Use of the Symbolic Manipulation Features of a CAS in their Lesson Planning

Representing Numbers as Continued Fractions and a TI-Nspire Package to do some Basic Continued Fraction Arithmetic

Teaching Mathematics with CAS SAGE

Techniques of Morphing to Stimulate the Teaching with Technology Examples within the TI-Nspire Environment and Cabri 3D

The Study of Envelopes in a CAS Environment

Using TI-Nspire 2D Graphs in a CAS Environment

Verschiedene Zugänge zur Zahl *e*

Zeitgemäßer Mathematikunterricht und zentrale Matura? Passt das zusammen?

Short Lectures

A stepwise CAS Course for Solving First-Order Partial Differential Equations

Advanced Techniques to compute Improper Integrals using a CAS

CAS in Teaching Linear Algebra: From Diagnosis, Connection, Deepening to Application

Computer, Tablet or Graphing Calculator

Designing Spatial Visualization Tasks for Middle School Students with a 3D Modelling Software

Differences between Expected Answers and the Answers given by CAS to School Equations

Dynamic Visual Proofs in Mathematics Education

Elektronische Prüfungsumgebungen

Erfahrungen mit CAS/GTR-SchülerInnen an der Fachhochschule

From Calculus to Dynamical Systems through CAS

Math with Programming – Shaken or Stirred

Modelling the Motion of an Elevator

Modern Mathematics Lessons with Technology and Central School-leaving exam? Does this go together

Motivating Students in an Introductory Matrix Algebra Course

Opinion of Teachers on the Use of the Wiris CAS for Teaching and Learning Mathematics

Question Types for Assessment of Mathematics Education in Dynamic Geometry Environments

Saving Private Goldbach

Students' Comparison of their Trigonometric Answers with the Answers of a CAS in Terms of Equivalence and Correctness

Study of Historical Geometric Problems by Means of CAS and DGS

The Derivation of Kepler's Three Laws using Newton's Law of Gravitation and the Law of Force

The Future of eTextbooks

The Task we do, the Software we choose

The Use of Graphical Materials in Teaching of Mathematics: Effects on Students' Understanding and Performance

TI-Nspire CAS and Laplace Transforms

Using a Mathematical Package for Modeling of Sleep Disorders in Patients with Traumatic Brain Injury

Using Computable Document Format in Teaching Mathematics

Using Fourier Series to control Mass Imperfections in Vibratory Gyroscopes

Using TI-Nspire in a Modelling Teacher's Training Course

Wie viel bzw. welches CAS benötigt man für die Zentralmatura Angewandte Mathematik an BHS

Working with Nonverbal Elements using DGS in the Mathematics

Workshops

"Verstehensorientiert" Unterrichten mit Blick auf die neue Reifeprüfung

Analytische Geometrie im Raum - visualisiert und technologiegestützt mit TI-Nspire

CA and Secondary School Mathematics: Mathematical Tasks in Principle become Tasks in Practice

Exploring Mathematics through Multiple Representations

From Exploration to Assessment using Maple T.A.

GINI-Coefficient and GOZINTO-Graph

Interactive Mathematics-work-book on a Pen-enabled Device

Interaktivität & Technologieeinsatz im Unterricht

Parametric 3D-Plot, ein zeitgemäßes mächtiges Werkzeug für die Raumgeometrie

Solving Brain Teasers/Twisters – CAS Assisted

Teaching Aids for CAS-compliant Lessons

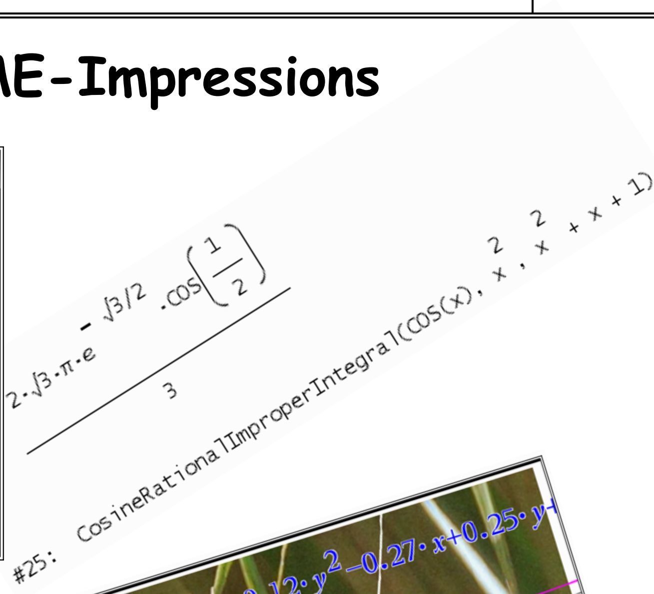
Technical Problems – Solved by Sec 2 Mathematics

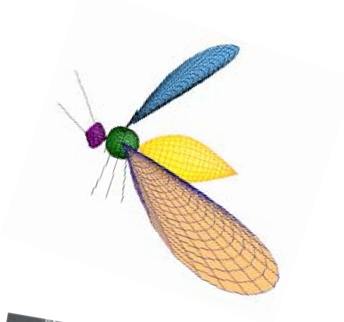
Wachstumsmodelle diskret und kontinuierlich

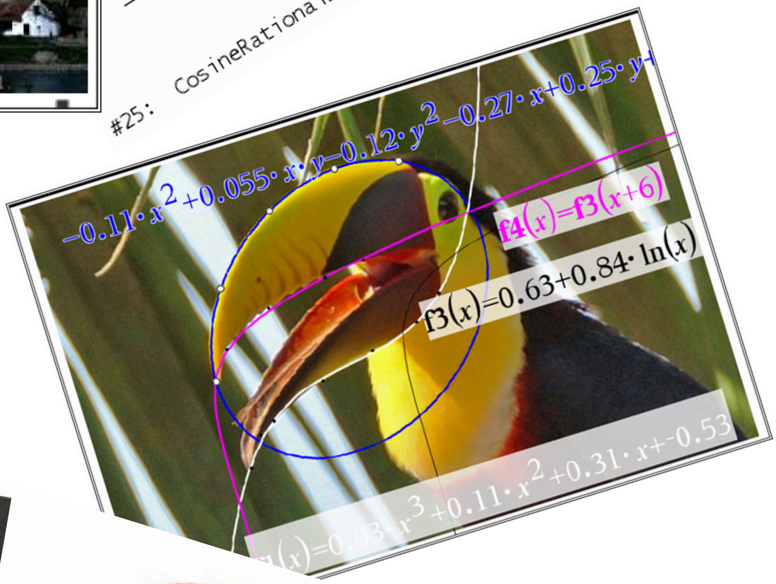
Wirtschaftliche Anwendungen in der Schulmathematik

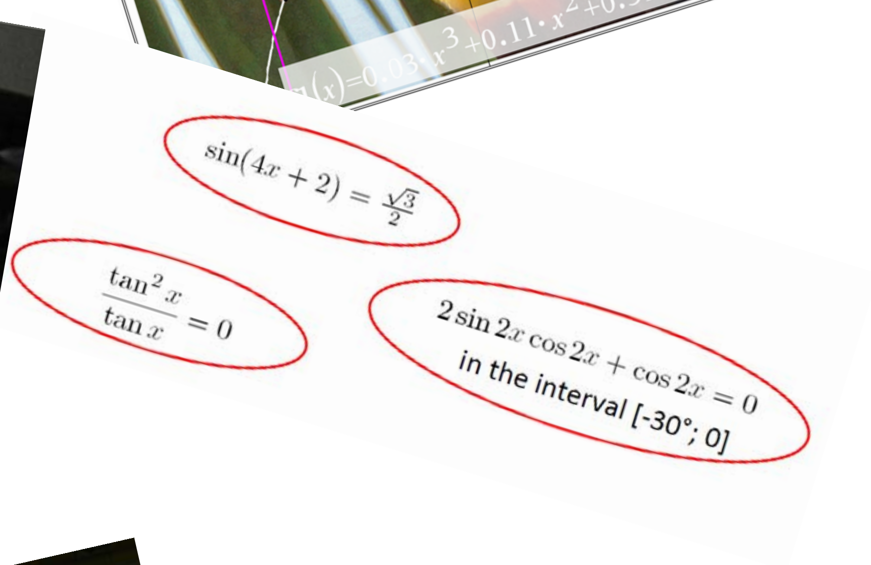
Workshop für CASIO Class Pad II

Some TIME-Impressions









I received a sad message at the end of July: Bert Waits passed away on 27 July. Bert was a passionate fighter for the use of technology in mathematics education and particulurly for the use of CAS.

Bert was very deep involved with the development of CAS on handheld devices as the TI-92 and Voyage 200. Numerous papers and textbooks – many of them written together with his colleague Frank Demana – were dedicated to technology in the teaching of mathematics. Whoever had attended a lecture given by Bert will never forget his enthusiasm and his energy.

All of us who had the privilege to meet Bert personally will remember fondly his warm hearted and homorous way to meet people. Several years ago my wife an I were honoured by his invitation to visit him and his wife Barb in their "paradise" on Sea-brook Island where he gave us a sand dollar. This fragile object has a place of honour in our living room.

Our deep sympathy is with Barb Waits and her family. We will miss Bert as an excellent teacher and as a wonderful man and friend.

Josef and Noor for the DUG and T^3-Community

You can find a collection of Bert's papers at: http://mathforum.org/library/view/12429.html