THE DERIVE - NEWSLETTER #84

ISSN 1990-7079

THE BULLETIN OF THE

USER GROUP

+ CAS-TI

Contents:

December 2011

!!!

I'd like to remind you that TIME 2012 is approaching. Please consider participation and don't forget to submit a lecture and/or workshop. Marina Lepp and Eno Tönisson are working on the social program. Fortunately we could win a prominent lady from Estonia for giving an additional keynote lecture.

Bitte überlegen Sie die Teilnahme an der TIME 2012 und vergessen Sie nicht, einen Vortrag oder Workshop anzumelden. Die erste Einreichung ist bereits eingelangt (aus Australien!!). Marina Lepp und Eno Tönisson arbeiten bereits am Begleitprogramm. Erfreulicherweise konnte auch eine prominente Dame aus Estland für einen zusätzlichen gewonnen werden.

Attend TIME2012 in Tartu, Estonia
July 11-14, 2012
http://time2012.ut.ee/

The next announcement is for now for our German speaking members:

Vorerst ergeht diese Einladung an unsere deutschsprachigen Mitglieder:

Mitgliedschaft bei der ACDCA

Das ACDCA (Austrian Center for Didactics of Computer Algebra) wurde Ende der 80er Jahre gegründet und ist seither eine der treibenden Kräfte für den Einsatz von CAS in der M-Ausbildung. Dieser Verein wurde kürzlich umstrukturiert und bietet allen Interessierten die Mitgliedschaft und die Gelegenheit zur Mitwirkung an. Bitte informieren Sie sich auf der Homepage des ACDCA über die Aufgaben und Ziele des ACDCA.
Info und Beitrittserklärung finden Sie auf

http://www.acdca.ac.at

Our DUG member Klaus Körner is asking for a FORTRAN Compiler. It needs not to be a fancy one. There are several Freeware Compilers offered on the web. Can anybody provide a special recommendation? Please write and I will forward your mail to Klaus.
Many thanks in advance

Klaus has an interesting proposal for a CAS application. There is not enough space in this DNL to write more. I will present his idea in the next issue,
Josef

Liebe DUG-Mitglieder,

Ich bin sehr froh, dass es mir dieses mal gelungen ist, den DNL rechtzeitig fertig zu stellen.

Dieser DNL besteht zum größten Teil aus Reaktionen unserer Mitglieder auf frühere Beiträge und auf Anfragen zu DERIVE und zum TI-Nspire. Für den Herausgeber ist es natürlich sehr erfreulich, dass die DNL-Artikel zum Weiterforschen anregen.

Stefan Welke lieferte eine tiefgründige Behandlung eines Aufsatzes von Carl Leinbach und mir und unser Freund David Halprin aus „Downunder" erinnert sich an seine Beschäftigung mit Flugzeugprofilen.

Eine Anfrage von Robert Setif traf sich mit einer Frage von Rob Gough zum populären Thema „Mandelbrotmenge". Dies war auch für mich Anlass genug, mich mit DERIVE daran zu versuchen.

Ich darf berichten, dass die DUG auch heuer wieder neue Mitglieder gewinnen konnte. Erik von Lantschoot – ein neues Mitglied - hat sich auch gleich mit einem trickreichen TI-Nspire-Programm zur Dreiecksauflösung eingestellt. Vielen Dank dafür.

Piotr Trebisz zeigt die dritte Folge seiner Schneckenhäuser und schließlich wünscht Roland Schröder mit einem vierblättrigen Klee den DUG-Mitgliedern ein erfolgreiches neues Jahr 2012. Diesen Wünschen schließe ich mich gerne an.

Schöne Beiträge aus Kolumbien (Nelson Urrego) und USA (Phil Todd) sind wieder eingelangt.

Beachten Sie bitte die Informationen auf der gegenüberliegenden Seite.

Ich verbleibe mit den besten Grüßen und Wünschen und mit der Hoffnung auf ein Treffen in Tartu (TIME 2012).

Dear DUG Members,

I am very happy that I could finish DNL#84 in time.

The major part of this DNL consists of reactions of our members on earlier contributions and on requests to DERIVE and TI-NspireCAS. It is very enjoyable for the editor that DNL articles inspire for further investigations and for providing comments.

Stefan Welke sent a profound treatment of an earlier paper presented by Carl leinbach and me. Our friend David Halprin from Downunder remembers his work with airplane profiles "some" years ago.

A request from Robert Setif met a question posed by Rob Gough both dealing with the popular issue "Mandelbrot Set". This was a welcome occasion for me to make a try with DERIVE.

I can report that the DUG could win again new members in 2011. Erik van Lantschoot – a new member - presents a tricky TI-Nspire-program for solving triangles. Many thanks for this.

Piotr Trebisz shows the 3rd part of his snail shells and finally Roland Schröder sends a four-leaf clover wishing all DUG members a successful New Year 2012. I would like to join him wishing you and your families the very best.

Great contributions from Colombia (Nelson Urrego) and USA (Phil Todd) have arrived.

Please notice the information on the opposite page.

I remain with best regards and wishes and hope to meet you in Tartu (TIME 2012).

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: March 2012

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-Encription, 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
Overcoming Branch & Bound by Simulation, J. Böhm, AUT
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
Statistics with TI-Nspire, G. Herweyers, BEL
Cesar Multiplication, G. Schödl, AUT
Henon & Co; Find your very own Strange Attractor, J. Böhm, AUT
Rational Hooks, J. Lechner, AUT
Cubus Simus, H. Ludwig, GER
Simulation of Dynamic Systems with various Tools, J. Böhm, AUT
Using DERIVE to simulate the basic steps of the Q. Shor's algorithm, N. Urrego, COL
Approximating Circular Arcs with Cubic Splines, Ph. Todd, USA

and others

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

Using Linear Algebra to Explain Some Unexpected Results or Re: What Josef and Carl Saw

Stefan Welke, Bonn, Germany, stefanwelke@web.de

(1) Introduction

Josef Böhm and Carl Leinbach developed in their article [1] a purely arithmetic proof for the convergence of a recursively defined sequence of points in $\mathbb{R}^2$:

(0.1) $p_{n+3} := \frac{1}{2} p_n + \frac{1}{2} p_{n+1} + 0 \cdot p_{n+2}$ for $n \in \mathbb{N}_0$ and $p_0 := (0,0), p_1 := (0,1), p_2 := (1,0)$

We shall slightly change the point of view and utilize complex numbers by identifying points $p = (x, y) \in \mathbb{R}^2$ with complex numbers $z = x + i \cdot y \in \mathbb{C}$ and $x, y \in \mathbb{R}$. Then (1.1) reads as

(0.2) $z_{n+3} := \frac{1}{2} z_n + \frac{1}{2} z_{n+1} + 0 \cdot z_{n+2}$ for $n \in \mathbb{N}_0$ and $z_0 := 0, z_1 := i, z_2 := 1$

This is neither ingenious nor new, but it enables us to use a different approach to the problem, which will show close relations to the treatment of Fibonacci and Lucas sequences. We shall develop a *Binet*-formula, or better, a *Binet*-representation for such sequences. We note at first, that (1.2) is a special case of the following general situation of a linear recurrence:

(0.3) $z_{n+3} := p \cdot z_n + q \cdot z_{n+1} + r \cdot z_{n+2}$ for $n \in \mathbb{N}_0$ with initial values $z_0, z_1, z_2 \in \mathbb{C}$

where p, q, r are complex numbers. Our next step is to rewrite the definition (1.2) as a recurrence of vectors in $\mathbb{C}^3$:

(0.4) $$\begin{pmatrix} z_{n+1} \\ z_{n+2} \\ z_{n+3} \end{pmatrix} := \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ p & q & q \end{pmatrix} \begin{pmatrix} z_n \\ z_{n+1} \\ z_{n+2} \end{pmatrix}$$ for $n \in \mathbb{N}_0$ and initial vector $\begin{pmatrix} z_0 \\ z_1 \\ z_2 \end{pmatrix} \in \mathbb{C}^3$

If we set $M := \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ p & q & r \end{pmatrix}$ and $v_n = \begin{pmatrix} z_n \\ z_{n+1} \\ z_{n+2} \end{pmatrix}$, then (1.4) is equivalent to

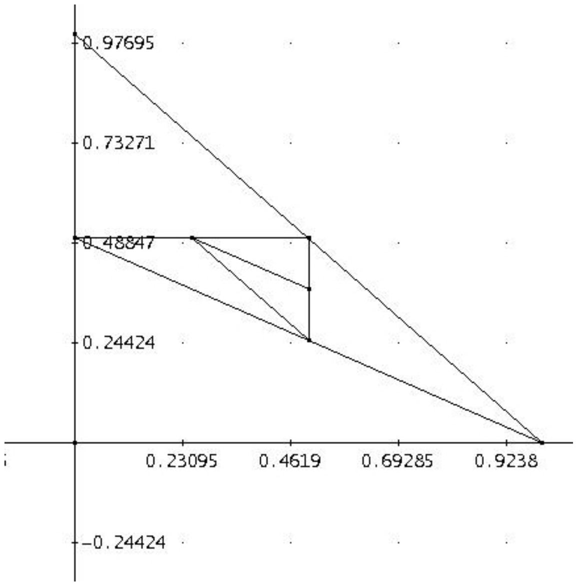
(0.5) $v_{n+1} = Mv_n$ and as a consequence: $v_n = Mv_{n-1} = M^2 v_{n-2} = ... = M^n v_0$.

This last formula unveils the true nature of the recurrence (1.3) : It has the same structure as the geometric sequence for numbers, which is an one-dimensional version of our three-dimensional recurrence. The following *Derive* function realizes (1.4):

```
#1: g(p,n,v):=
      PROG(w:=[[0,1,0],[0,0,1],[p SUB 1,p SUB 2,p SUB 3]],
      b:=VECTOR((w^t*v) SUB 1,t,n SUB 1,n SUB 2),[RE(b),IM(b)]`)
```

Here `p` ist he vector of coefficients `p=[p,q,r]` from (1.3), `n=[n1,n2]` gives the first and last exponent in $\{M^{n1}v_0, M^{n1+1}v_0, ..., M^{n2}v_0\}$, and `v=[z1,z2,z3]` the initial vector.

```
#2: g([0.5,0.5,0],[0,7],[0,#i,1])

#3:
[[0,0],[0,1],[1,0],[0,0.5],[0.5,0.5],[0.5,0.25],[0.25,0.5],[0.5,0.37
5]]
```



The objective of Carl's and Josef's article [1] was the proof of convergence of (1.1) respectively. (1.2). We will conclude this introduction with two simple propositions about limits of recurrences.

Proposition 1.1

If the limit $z_\infty := \lim_{n\to\infty} z_n \neq 0$ of (1.2) *exists, then* $p+q+r=1$.

Proof: If this limit exists, then obviously $z_\infty := \lim_{n\to\infty} z_n = \lim_{n\to\infty} z_{n+1} = \lim_{n\to\infty} z_{n+2} = \lim_{n\to\infty} z_{n+3} \neq 0$. Thus $z_\infty = p\cdot z_\infty + q\cdot z_\infty + r\cdot z_\infty = (p+q+r)z_\infty$. Division by $z_\infty \neq 0$ gives the result. □

As we shall see, $p+q+r=1$ is necessary but not sufficient for convergence. The next proposition is about boundedness of the recurrence.

Proposition 1.2

1.2.1. If under the conditions (1.2) $|p|+|q|+|r| \leq 1$, *then the sequence* $\{z_n\}_{n\in\mathbb{N}}$ *is bounded by* $M := \max\{|z_1|,|z_2|,|z_3|\}$ *and has at least one accumulation point.*

1.2.2. If under the conditions (1.2) $|p|+|q|+|r| = Q < 1$, *then* $\lim_{x\to\infty} z_n = 0$.

Proof of 1.2.1.: From the triangle inequality follows:

$$|z_{n+3}| \leq |pz_{n+2}| + |qz_{n+1}| + |rz_n| \leq (|p|+|q|+|r|)\cdot \max\{|z_{n+2}|,|z_{n+1}|,|z_n|\} \leq \max\{|z_{n+2}|,|z_{n+1}|,|z_n|\} =: M_n$$

An immediate consequence is $M_{n+1} = \max\{|z_{n+3}|,|z_{n+2}|,|z_{n+1}|\} \leq \max\{|z_{n+2}|,|z_{n+1}|,|z_n|\} = M_n$. We conclude by induction: $|z_n| \leq M_n \leq M_1 = M$. This proves the first part.

The boundedness implies, that the sequence stays within the closed polycylinder $D := \{(z^{(1)}, z^{(2)}, z^{(3)}) \in \mathbb{C}^3 \mid |z^{(j)}| \leq M, j \in \{1,2,3\}\}$, which is a compact subset of $\mathbb{C}^3$. This proves the second part.

Proof of 1.2.2.: The same reasoning as above gives $M_{n+3} \le Q \cdot M_n$, so by induction $|z_{3n+k}| \le M_{3n+k} \le Q^n \cdot M_1 = Q^n \cdot M$ for $k \in \{0,1,2\}$. Since $0 \le Q < 1$, 1.2.2. is proved. □

The third proposition is about the convergence of the sequence of quotients $\left\{ \frac{z_{n+1}}{z_n} \right\}_{n \in \mathbb{N}}$. As we know from the Fibonacci numbers, the sequence of quotients may converge, even if the sequence itself does not.

Proposition 1.3

If the sequence of quotients of (1.2) converges, and if the limit $q_\infty := \lim_{n\to\infty} \frac{z_{n+1}}{z_n}$ *is not equal to zero, then* q_∞ *satisfies the algebraic equation* $q_\infty^{\;3} - r \cdot q_\infty^{\;2} - q \cdot q_\infty - p = 0$.

Proof: Assume, the limit exists, then clearly $q_\infty := \lim_{n\to\infty} \frac{z_{n+1}}{z_n} = \lim_{n\to\infty} \frac{z_{n+2}}{z_{n+1}} = \lim_{n\to\infty} \frac{z_{n+3}}{z_{n+2}}$ follows, because almost all numbers of the recurrence are not equal to zero. We now divide (1.3) by z_n and obtain the equivalence $\frac{z_{n+3}}{z_n} = p + q\frac{z_{n+1}}{z_n} + r\frac{z_{n+2}}{z_n} \Leftrightarrow \frac{z_{n+3}}{z_{n+2}} \cdot \frac{z_{n+2}}{z_{n+1}} \cdot \frac{z_{n+1}}{z_n} = p + q\frac{z_{n+1}}{z_n} + r\frac{z_{n+2}}{z_{n+1}} \cdot \frac{z_{n+1}}{z_n}$. Passing to the limit, the equation becomes $q_\infty^{\;3} = r \cdot q_\infty^{\;2} + q \cdot q_\infty + p$. This is equivalent to the statement of the proposition. □

Remark: If we choose $z_0 = 0, z_1 = 1, z_2 = 1$, this three-step recurrence produces with $p = 1$, $q = 2$, and $r = 0$, as well as with $p = 0$, $q = 1$, and $r = 1$ and with any linear combination of these two parameter vectors $(p,q,r) = \alpha \cdot (1,2,0) + (1-\alpha) \cdot (0,1,1)$ with $\alpha \in \mathbb{R}$ the Fibonacci numbers. We give an example:

```
#4:   w ≔ 0.7·[1, 2, 0] + 0.3·[0, 1, 1]

#5:   g(v, [0, 10], [0, 1, 1])`
                               1

#6:          [0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55]
```

(2) The *Binet*-Representation

We compute the determinant and the characteristic polynomial of the matrix M from (1.5):

```
#7:   M ≔ [0,1,0;0,0,1;p,q,r]

#8:   DET(M)

#9:                 p

#10:  CHARPOLY(M,λ)

                3       2
#11:         - λ  + r·λ   + q·λ + p
```

We notice, that the possible limit of the sequence of quotients from proposition 1.3 must be a root of the characteristic polynomial of M .

Now let M have three distinct roots, $\lambda_1, \lambda_2, \lambda_3$, possibly one equal to zero. These roots are the eigenvalues of M . Let λ be any eigenvalue, then clearly $\lambda^3 = p + q\lambda + r\lambda^2$. This explains the following identities:

(0.6)
$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ p & q & r \end{pmatrix} \begin{pmatrix} 1 \\ \lambda \\ \lambda^2 \end{pmatrix} = \begin{pmatrix} \lambda \\ \lambda^2 \\ p + q \cdot \lambda + r \cdot \lambda^2 \end{pmatrix} = \begin{pmatrix} \lambda \\ \lambda^2 \\ \lambda^3 \end{pmatrix} = \lambda \cdot \begin{pmatrix} 1 \\ \lambda \\ \lambda^2 \end{pmatrix}$$

Therefore the corresponding eigenvectors are, up to a factor:

(0.7)
$$e_1 = \begin{pmatrix} 1 \\ \lambda_1 \\ \lambda_1^2 \end{pmatrix}, e_2 = \begin{pmatrix} 1 \\ \lambda_2 \\ \lambda_2^2 \end{pmatrix}, e_3 = \begin{pmatrix} 1 \\ \lambda_3 \\ \lambda_3^2 \end{pmatrix}$$

We assume, that all eigenvalues are distinct, then Vandermonde's determinant gives

(0.8)
$$\det \begin{pmatrix} 1 & 1 & 1 \\ \lambda_1 & \lambda_2 & \lambda_3 \\ \lambda_1^2 & \lambda_2^2 & \lambda_3^2 \end{pmatrix} = (\lambda_1 - \lambda_2)(\lambda_2 - \lambda_3)(\lambda_3 - \lambda_1) \neq 0 \ .$$

This result proves the linear independence of the three eigenvectors (0.7) for distinct roots, $\lambda_1, \lambda_2, \lambda_3$, and we can represent every initialvector $v_0 \in \mathbb{C}^3$ as a linear combination of (0.7), because the vectors (0.7) form a basis of $\mathbb{C}^3$:

(0.9)
$$v_0 = \alpha_1 e_1 + \alpha_2 e_2 + \alpha_2 e_2, \alpha_k \in \mathbb{C}^3$$

We call (0.9) the *Binet*-decomposition of v_0 . If we now calculate (1.5) with this decomposition, then we obtain by linearity: $Mv_0 = M(\alpha_1 e_1 + \alpha_2 e_2 + \alpha_2 e_2) = \alpha_1 \lambda_1 e_1 + \alpha_2 \lambda_2 e_2 + \alpha_2 \lambda_3 e_2$ and hence

(0.10)
$$v_n = M^n v_0 = M^n (\alpha_1 e_1 + \alpha_2 e_2 + \alpha_2 e_2) = \alpha_1 \lambda_1^n e_1 + \alpha_2 \lambda_2^n e_2 + \alpha_2 \lambda_3^n e_2$$

At this point, after passing to the first coordinates of the vectors v_0 and v_n , the proof of the first part of the following proposition is completed:

Proposition 2.1

Assume, that the characteristic polynomial $c(\lambda) = \lambda^3 - r\lambda^2 - q\lambda - p$ has three distinct roots $\lambda_1, \lambda_2, \lambda_3$, and $\alpha_1, \alpha_2, \alpha_3$ are the coefficients of the Binet-*decomposition of the initialvector, then the linear recurrence (1.3) has the following representation:*

(0.11)
$$z_n = \alpha_1 \lambda_1^n + \alpha_2 \lambda_2^n + \alpha_2 \lambda_3^n \ , \ n \in \mathbb{N}$$

This representation is unique.

The uniqueness of this representation is a consequence of the uniqueness of the eigenvalues of a matrix and of the uniqueness of the representation of a vector with respect to a given basis of the vector space.
We call (2.6) the *Binet*-formula, or the *Binet*-representation of the recurrence. Let us look at our example with $p = q = \frac{1}{2}$, $r = 0$, and the initial values $z_0 = 0, z_1 = i, z_2 = 1$. The roots of the characteristic equation and the coefficients of the *Binet*-decomposition are:

```
#12:  InputMode ≔ Word

#13:  SOLVE(λ^3  - 0.5·λ - 0.5 = 0, λ)

#14:  λ = -0.5 - 0.5·i OR λ = -0.5 + 0.5·i OR λ = 1

#15:  SOLVE([0, 1, i] = α1·[1,-0.5 - 0.5·i,(-0.5 - 0.5·i)^2] +
      + α2·[1,-0.5 + 0.5·i, (-0.5 + 0.5·i)^2] + α3·[1, 1, 1], [α1, α2, α3])

#16:  α1 = 0.4 + 0.2·i AND α2 = -0.8 - 0.6·i AND α3 = 0.4 + 0.4·i
```

Thus the *Binet*-representation of Carl's and Josef's recurrence is

(0.12) $$z_n := (0.4 + 0.2i)(-0.5 - 0.5i)^n + (-0.8 - 0.6i)((-0.5 + 0.5i)^n + 0.4 + 0.4i \quad .$$

The limit of this sequence is easily found, because $\left|-0.5 - 0.5i\right| = \left|-0.5 + 0.5i\right| = \frac{1}{\sqrt{2}} < 1$. We get:

$$\lim_{n\to\infty} z_n := (0.4 + 0.2i)\cdot\lim_{n\to\infty}(-0.5 - 0.5i)^n + (-0.8 - 0.6i)\cdot\lim_{n\to\infty}(-0.5 + 0.5i)^n + 0.4 + 0.4i = 0.4 + 0.4i$$

A close examination of the *Binet*-representation (0.11) tells us that

- the convergence or divergence of the recurrence depends on the eigenvalues of the characteristic polynomial, hence on the parameters p, q, r in the definition of the recurrence, and on the initial values of the sequence.
- there exits an interpolating function $b(t) := \alpha_1\lambda_1^{\,t} + \alpha_2\lambda_2^{\,t} + \alpha_2\lambda_3^{\,t}, t \in \mathbb{R}$, which is the superposition of at most three spirals, if $\left|\lambda_j\right| \neq 1$, and/or circles, if $\left|\lambda_k\right| = 1$.

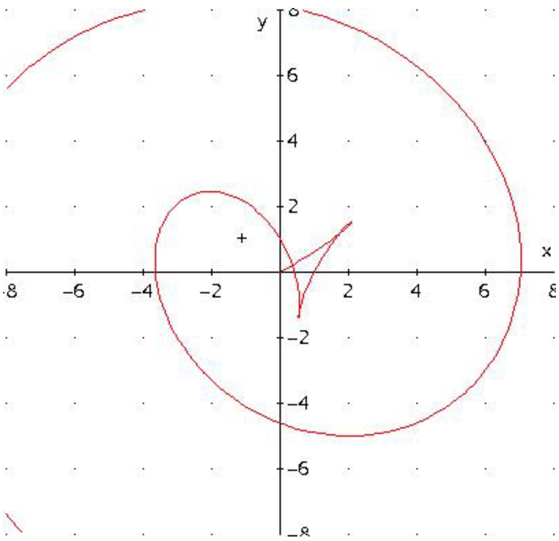
We will give an illustrative example:
Choose three eigenvalues $\lambda_1 = -1, \lambda_2 = -1 + i, \lambda_3 = -0.5 - 0.5i$, then *DERIVE* gives:

```
#17:  EXPAND((z + 1)·(z + 1 - i)·(z + 0.5 + 0.5·i), z)

#18:  z^3 + 2.5·z^2 + 2.5·z + 1 - i·(0.5·z^2  + 0.5·z)
```

So the parameters are $p = -1, q = r = -2.5 + 0.5i$. The next function `spiral1(w,t,v)` computes the parametric equation of the spiral with the parameter vector `w=[p,q,r]`, the variable `t`, and the initialvector `v=[z0,z1,z2]` :

```
#19:  spiral1(w, t, v) ≔
            Prog
             e ≔ EIGENVALUES([[0, 1, 0], [0, 0, 1], w])
             w ≔ VECTOR(SUBST([1; s; s^2], s, e↓k), k, 1, 3)
             z ≔ SOLUTIONS([α, β, γ]·w = [v↓1; v↓2; v↓3],
                      [α, β, γ])·VECTOR(If(k ≠ 0, k^t, 0, 0), k, e)
             [RE(z), IM(z)]` ↓1

#20:  spiral1([-1, -2.5 + 0.5·i, -2.5 + 0.5·i], t, [0, 1, i])
```

We do not print the algebraic result but rather the graphic.



The *Binet*-decomposition is:

$$\begin{pmatrix}0\\1\\i\end{pmatrix}=3\begin{pmatrix}1\\-1\\1\end{pmatrix}-\left(\tfrac{3}{5}+\tfrac{4}{5}i\right)\begin{pmatrix}1\\-1+i\\-2i\end{pmatrix}-\left(\tfrac{12}{5}-\tfrac{4}{5}i\right)\begin{pmatrix}1\\-\frac{1}{2}-\frac{i}{2}\\\frac{i}{2}\end{pmatrix}$$

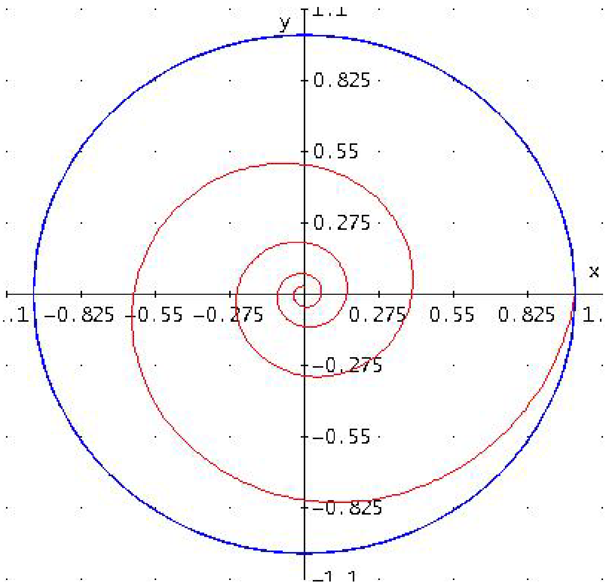
The eigenvalue λ_2 with $|-1+i|=\sqrt{2}>1$ is responsible for $\lim_{t\to\infty} b(t)=\infty$, i.e. that the spiral tends to infinity.

We change now the set of initial values, `[1,-0.5-0.5i,0.5i]` in the first case:

```
#22: spiral1([-1, -2.5 + 0.5·i, -2.5 + 0.5·i], t, [1, -0.5 - 0.5·i, 0.5·i])
#23 : [2^(-t/2)·COS(3·π·t/4),-2^(-t/2)·SIN(3·π·t/4)]
```

and `[1,-1,1]` in the second case:

```
#24: spiral1([-1, -2.5 + 0.5·i, -2.5 + 0.5·i], t, [1, -1, 1])
```




```
#25 : [COS(π·t),SIN(π·t)]
```

In the first case the *Binet*-decomposition is

$$v_0=1\cdot\begin{pmatrix}1\\-\frac{1}{2}-\frac{i}{2}\\\frac{i}{2}\end{pmatrix}$$

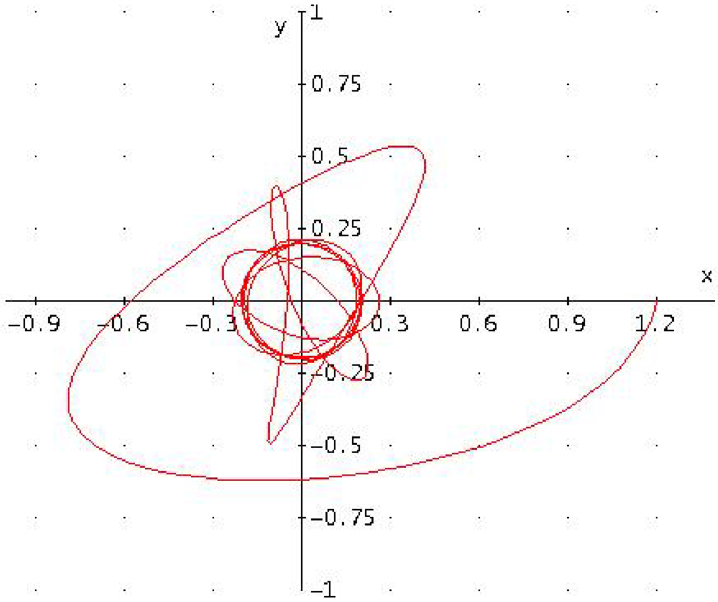
so $b(t):=(-\frac{1}{2}-\frac{1}{2}i)^t$, is the *Binet*-representation, which is the equation of a clockwise winding spiral with limit point 0. In the second case we find

$$v_0=1\cdot\begin{pmatrix}1\\-1\\1\end{pmatrix}\text{ and } b(t)=(-1)^t=e^{-i\pi t},$$

which is the parametric form of the unit circle in $\mathbb{C}$.

A last example in this instance is a combination of the last two cases with

$$v_0=\frac{1}{5}\cdot\begin{pmatrix}1\\-1\\1\end{pmatrix}-1\cdot\begin{pmatrix}1\\-\frac{1}{2}-\frac{i}{2}\\\frac{i}{2}\end{pmatrix}=\begin{pmatrix}\frac{6}{5}\\-\frac{7}{10}-\frac{1}{2}i\\\frac{1}{5}+\frac{1}{2}i\end{pmatrix}.$$



The corresponding curve winds clockwise around a circle with radius $\frac{1}{5}$. If we had added any multiple of the third eigenvector, how ever small in absolute value, we would see instead something like a spiral tending to infinity.

These examples demonstrate the impact of the initial values.

The following *DERIVE*-function is part of the function `spiral1` and computes the coefficients of the *Binet*-decomposition of a initial vector.

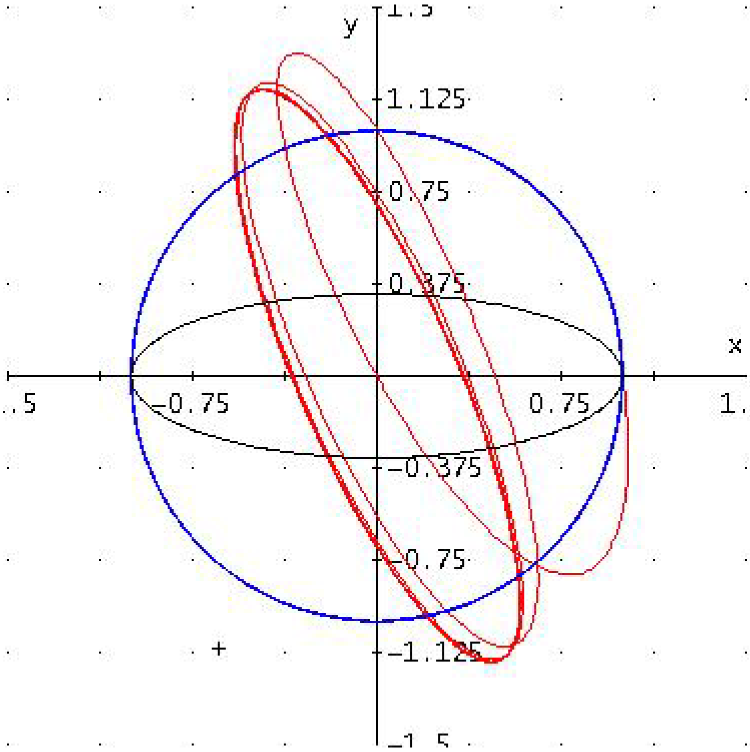
```
#27:  binet_decomp(p, v) ≔
          Prog
              e ≔ EIGENVALUES([[0, 1, 0], [0, 0, 1], p])
              w ≔ VECTOR(SUBST([1; s; s^2], s, e↓k), k, 1, 3)
              [e, SOLUTIONS([α, β, γ]·w = [v↓1; v↓2; v↓3], [α, β, γ])]

#28: binet_decomp([-1, -2.5 + 0.5·i, -2.5 + 0.5·i], [0, i, 1])

#29:         [[-1, -1 + i, -0.5 - 0.5·i], [[3, -0.6 - 0.8·i, -2.4 + 0.8·i]]]
```

(3) Constructing Recurrences to Given Figures

We are in the position to explain the occurrence of the ellipses in Josef's example [1,p. 24], because we have a theory, that can predict the resulting phenomena. Consider the case of three distinct eigenvalues $|\lambda_1|<1, |\lambda_2|=|\lambda_3|=1, \lambda_2\neq 1$ and $\overline{\lambda_2}=\lambda_3$, where the coefficients of the *Binet*-decomposition satisfy the conditions $\alpha_2\cdot\alpha_3\neq 0$ and $|\alpha_2|\neq|\alpha_3|$. Then the curve given by $c(t):=\alpha_2\cdot\lambda_2^t+\alpha_3\cdot\overline{\lambda_2^t}$ is an ellipse. The reason is simply, that $\mathbb{C}$ is a two-dimensional vector space over $\mathbb{R}$, and that a real linear map $f:\mathbb{C}\to\mathbb{C}$ has the form $f(z)=\alpha\cdot z+\beta\cdot\overline{z}$ with $\alpha,\beta\in\mathbb{C}$. The image of a circle is in general an ellipse, and it is evident, that we have $c(t)=f(\lambda_2{}^t)$, with $f(z):=\alpha_2\cdot z+\alpha_3\cdot\overline{z}$. The conditions on α_2,α_3 guarantee, that we obtain an ellipse, which is not a circle. The following example illustrates this case:

Let $\lambda_1=0.5, \lambda_2=-0.8+0.6i$ and $\lambda_3=-0.8-0.6i$ be the roots of the characteristic polynomial. Then the parameters of the recurrence are $p=0.5, q=-0.2, r=-1.1$. We choose three distinct initial vectors and calculate the corresponding *Binet*-decompositions:

$$v_1=\begin{pmatrix}0\\ i\\ 1\end{pmatrix}=\frac{20+32i}{41}e_1-\frac{63-34i}{246}e_2-\frac{57+226i}{246}e_3, v_2=e_2, v_3=\begin{pmatrix}1\\ -0.8-0.2i\\ 0.28+0.32i\end{pmatrix}=\frac{1}{3}e_2+\frac{2}{3}e_3$$



with

$$e_1 = \begin{pmatrix} 1 \\ 0.5 \\ 0.25 \end{pmatrix}, e_2 = \begin{pmatrix} 1 \\ -0.8+0.6i \\ 0.28-0.96i \end{pmatrix}, e_3 = \begin{pmatrix} 1 \\ -0.8-0.6i \\ 0.28-0.96i \end{pmatrix}$$

.

The corresponding curves are red for v_1, blue for v_2, and black for v_3.

The recurrence (1.3) produces a sequence of points $\{z_0, z_1, z_2, ...\}$, and we can draw the corresponding sequence of line segments, each segment connecting two subsequent points z_n and z_{n+1}. If the curve given by $b(t)$ is a circle, then all segments have equal length.

Now two cases can happen:

(1) The sequence is periodic, then the sequence of line segments looks like a regular polygon. This happens, if the corresponding eigenvalue is of the form $\lambda = e^{a\cdot\pi\cdot i}, a \in \mathbb{Q} \setminus \{0\}$.
(2) The sequence is not periodic and the points are a dense subset of the circle. In this case the corresponding eigenvalue is of the form $\lambda = e^{a\cdot\pi\cdot i}, a \in \mathbb{R} \setminus \mathbb{Q}$. Then the envelope of all line segments is a smaller circle with the same centre as the generating circle.

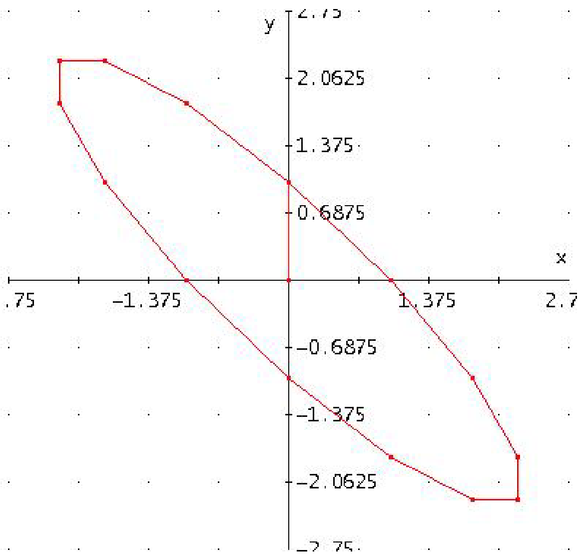
In the case of two conjugate eigenvalues of absolute value 1 we thus have as envelopes of the line segments either a polygon with vertices on an ellipse (1), or an ellipse (2). We will investigate Josef's example from [1,p. 24]. The first observation is, that Rüdiger Baumann's approach is almost equal to our first function `g(p,n,v)`, but he uses the `iterates` function instead of matrix powers. We recognize, that in our notation $p = 0.95, q = -0.1$ and $r = 0$ in the first example of [1,p. 24] and $p = -0.95, q = -0.1$ and $r = 0$ in the second. Now we look at the roots of the characteristic polynomial:

```
#30:  APPROX(SOLUTIONS(λ^3  + 0.1·λ + 0.95 = 0, λ))

#31: [-0.9491533240,0.4745766620+0.8807207416·i,0.4745766620-0.8807207416·i]

#32: APPROX(ABS(#32 SUB 2))

#33:  1.000445916

#34: APPROX(SOLUTIONS(λ^3  + 0.1·λ - 0.95 = 0, λ))

#35: [0.9491533240,-0.4745766620+0.8807207416·i,-0.4745766620-0.8807207416·i]
```

The real eigenvalue has absolute value smaller than 1, so its contribution to the *Binet*-representation is negligible for large values of n. In both cases the other two eigenvalues are conjugate to each other and of absolute value nearly 1.

We finish this section with two examples. The first is a periodic recurrence. Set $\lambda_1 = 0$, $\lambda_2 = e^{i\cdot\pi/7}$, and $\lambda_3 = e^{-i\cdot\pi/7}$. We compute the parameters p,q,r with `f1` and then draw the picture with `polygon1`:

```
#36: f1(x, y, z) ≔ [x·y·z, - (x·y + y·z + z·x), x + y + z]

#37 : f1(0,EXP(pi·i /7),EXP(-pi·i/7)

#38 : [0, -1, 1.801937735]

#39: pointseq1(p, n, v) ≔ VECTOR(spiral1(p, k, v), k, n SUB 1 , n SUB2 )
```

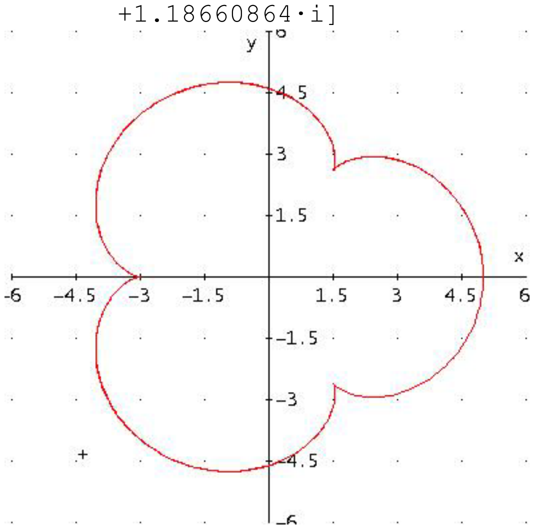


The next example demonstrates that we can even achieve cycloids and periodic polygons with vertices on a cycloid. We use the *Binet*-representation to construct a recurrence with the desired property.

We want to construct a cycloid with three cusps on a circle. A possible parametric representation in complex notation is:
$b(t) = 4\cdot e^{i\cdot\alpha\cdot t} + e^{4i\cdot\alpha\cdot t}, \alpha \in \mathbb{R} \setminus \{0\}$

To construct a recurrence with the above *Binet*-representation, choose eigenvalues $\lambda_1 = 1, \lambda_2 = e^{i\cdot\alpha} = 0.8 + 0.6i, \lambda_3 = e^{4i\cdot\alpha} = \lambda_1^{\ 4}$. We perform the following computations:

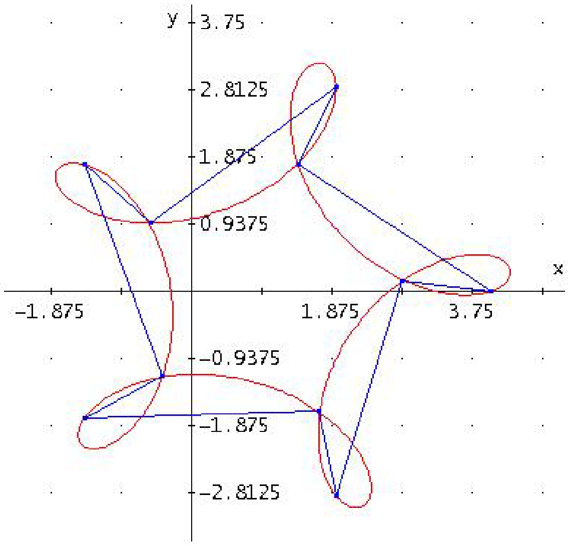
```
#40: l:=0.8+0.6·i

#41: f1(l,a,a^4)

#42: f2(x, n, m) ≔ n·[1, x, x^2 ] + [1, x^m , x^(2m) ]

#43: f2(a, 4, 4)

#44: [0.1512431615+0.9884965887·i,-1.533215641-2.175105228·i,2.38197248+
     +1.18660864·i]
```




```
#45: spiral1(#42,t,#44)
```

Note, that the function `f2(x, n, m)` computes the initial vector $v_0 = 4\cdot e_2 + e_3$. The result looks like the figure to the left.

If we want to construct a recurrence with a periodic polygon with vertices on a cycloid, we need to choose eigenvalues, which are roots of unity, e.g.:
$\lambda_1 = 1, \lambda_2 = e^{i\cdot\pi/5}, \lambda_3 = e^{4i\cdot\pi/5} = \lambda_1^{\ 4}$
We omit the calculations and just present the result in the next figure:



The previous examples suggest the question, which figures are constructible with recurrences. So far, we have been able to construct spirals, circles, ellipses, and cycloids as interpolations of *Binet*-representations. The resulting sequences are periodic or infinite.

We proceed to construct recurrences, where the corresponding *Binet*-representation leads to a parametric curve, which is either a parabola or a hyperbola.

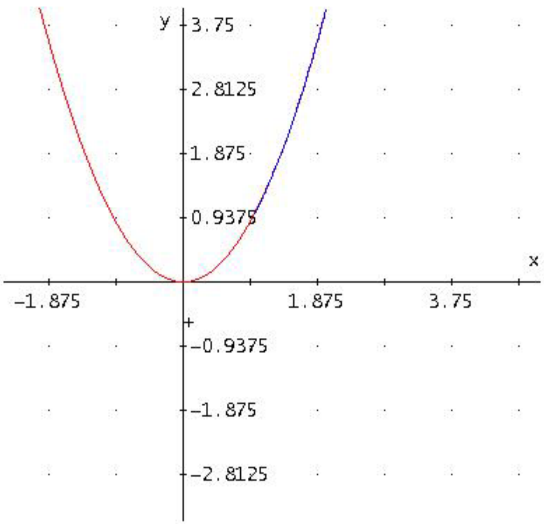
Let us begin with a parabola. To do so, we choose three real eigenvalues $\lambda_2 > 0, \lambda_3 = \lambda_2^2$, and $\lambda_1 = 1$. Now choose the initial vector

(0.13) $$v_0 = e_2 + i \cdot e_3 = \begin{pmatrix} 1 \\ \lambda_2 \\ \lambda_2^2 \end{pmatrix} + i \begin{pmatrix} 1 \\ \lambda_2^2 \\ \lambda_2^4 \end{pmatrix}$$

Then the interpolating function is $b(t) = \lambda_2^t + i\lambda_2^{2t} = \lambda_2^t + i\left(\lambda_2^t\right)^2$. The resulting parametric curve is part of a parabola as the next figure shows.

Take $\lambda_2 = 2, \lambda_3 = 4$. Then $(z-1)(z-2)(z-4) = z^3 - 7z^2 + 14z - 8$, so $p = 8, q = -14, r = 7$. We obtain as initial vector $v_0 = \begin{pmatrix} 1 \\ 2 \\ 4 \end{pmatrix} + i \begin{pmatrix} 1 \\ 4 \\ 16 \end{pmatrix} = \begin{pmatrix} 1+i \\ 2+4i \\ 4+16i \end{pmatrix}$.

```
#45: spiral1([8, -14, 7], t,[1 + i, 2 + 4·i, 4 + 16·i])
```




```
#46 :        [2^t,4^t]
```

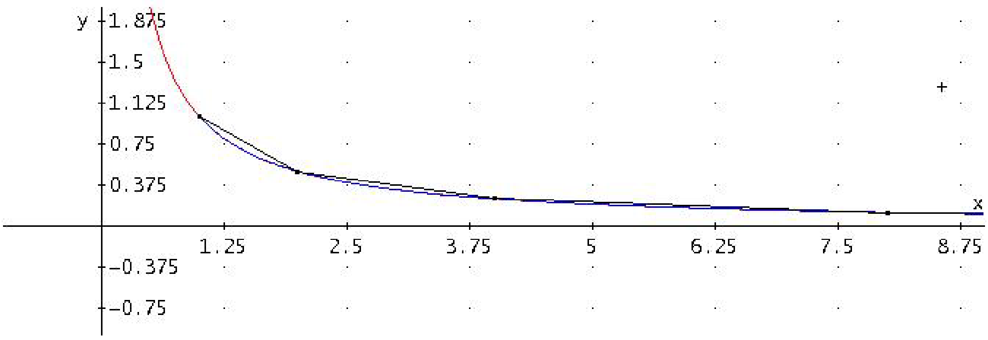
Notice, that the blue part of the parabola is the parametric curve corresponding to the *Binet*-representation.
If we had chosen $\lambda_2 = 0.5, \lambda_3 = 0.25$, then the corresponding parametric curve with a suitable initial vector would be the arc of the parabola connecting $z = 1+i$ with $z = 0$.

To complete our tour through conic sections we will give now the construction of a *Binet*-representation, which generates a parametric curve, which is part of a hyperbola.

For a hyperbola we choose $\lambda_1 = 1, \lambda_2 > 0$, and $\lambda_3 = \dfrac{1}{\lambda_2}$. Then $p = 1$, $q = -r$, and $r = 1 + \lambda_2 + \lambda_3 = 1 + \lambda_2 + \dfrac{1}{\lambda_2}$. The next figure is with $\lambda_1 = 1, \lambda_2 = 2$, and $\lambda_3 = 0.5$, so

$p=1, q=-3.5, r=3,5$. Since the parametric curve $b(t)=2^t+i\cdot 2^{-t}$ is part of the hyperbola given by $y=\dfrac{1}{x}$, we take $v_0=\begin{pmatrix}1\\2\\4\end{pmatrix}+i\cdot\begin{pmatrix}1\\0.5\\0.25\end{pmatrix}$ as initial vector.

```
#47: spiral1([1, -3.5, 3.5], t, [1, 2, 4] + i·[1, 0.5, 0.25])

#48: [2^t, 0.5^t]

#49: pointseq1([1, -3.5, 3.5], [0, 6], [1, 2, 4] + i·[1, 0.5, 0.25])

#50: We omit the list of points, which is the result of #49:
```



So far, we have been able to construct special conics, and the question is: Is it possible to generate any conic section as the interpolation of a *Binet*-representation of a suitable recurrence. The answer is affirmative, and the next two simple propositions are the key to the proof of the general result.

Let $f:\mathbb{C}\to\mathbb{C}$ be a function. We can apply f to the components of the initial vector v_0 to obtain the modified initial vector

$$v_0'=\begin{pmatrix}f(z_0)\\f(z_1)\\f(z_2)\end{pmatrix}. \qquad (0.14)$$

If we use this modified initial vector in (0.3) we obtain a modified recurrence $\{z_n'\}_{n\in\mathbb{N}}$ as result. In some particular cases the impact of f on the recurrence is the same as on the initial vector, i.e.: $z_n'=f(z_n)$ for all $n\in\mathbb{N}$.

Proposition 3.1

Let $\{z_n'\}_{n\in\mathbb{N}}$ be a recurrence with initial vector (0.14), then

(1) $f(z)=a\cdot z, a\in\mathbb{C}$, implies $z_n'=f(z_n)$ for all $n\in\mathbb{N}$

(2) $p+q+r=1$ and $f(z)=a\cdot z+b$ with $a,b\in\mathbb{C}$ *implies $z_n'=f(z_n)$ for all $n\in\mathbb{N}$*

(3) $p,q,r\in\mathbb{R}$ and $f(z)=a\cdot z+b\cdot\overline{z}$ with $a,b\in\mathbb{C}$ *implies $z_n'=f(z_n)$ for all $n\in\mathbb{N}$*

(4) $p,q,r\in\mathbb{R}, p+q+r=1$ and $f(z)=a\cdot z+b\cdot\overline{z}+c$ with $a,b,c\in\mathbb{C}$ *implies $z_n'=f(z_n)$ for all $n\in\mathbb{N}$*

Proof: (1) is an evident consequence from the definition. To prove (2) observe

(0.15)

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ p & q & q \end{pmatrix}\begin{pmatrix} a\cdot z_n + b \\ a\cdot z_{n+1} + b \\ a\cdot z_{n+2} + b \end{pmatrix} = \begin{pmatrix} a\cdot z_{n+1} + b \\ a\cdot z_{n+2} + b \\ p\cdot(a\cdot z_n + b) + q\cdot(a\cdot z_{n+1} + b) + r\cdot(a\cdot z_{n+2} + b) \end{pmatrix} =$$

$$= \begin{pmatrix} a\cdot z_{n+1} + b \\ a\cdot z_{n+2} + b \\ a\cdot(p\cdot z_n + q\cdot z_{n+1} + r\cdot z_{n+2}) + (p+q+r)\cdot b) \end{pmatrix} = \begin{pmatrix} a\cdot z_{n+1} + b \\ a\cdot z_{n+2} + b \\ a\cdot(p\cdot z_n + q\cdot z_{n+1} + r\cdot z_{n+2}) + b) \end{pmatrix} = \begin{pmatrix} f(z_{n+1}) \\ f(z_{n+2}) \\ f(z_{n+3}) \end{pmatrix}$$

The proof of (3) is by calculation

(0.16)
$$\begin{aligned} & p\cdot(a\cdot z_n + b\cdot \overline{z}_n) + q\cdot(a\cdot z_{n+1} + b\cdot \overline{z}_{n+1}) + r\cdot(a\cdot z_{n+2} + b\cdot \overline{z}_{n+2}) = \\ & = a\cdot(p\cdot z_n + q\cdot z_{n+1} + r\cdot z_{n+2}) + b\cdot\overline{(p\cdot z_n + q\cdot z_{n+1} + r\cdot z_{n+2})} = \\ & = a\cdot z_{n+3} + b\cdot \overline{z}_{n+3} = f(z_{n+3}) \end{aligned}$$

because conjugation of a complex number and multiplication with a real number commute.

The proof of (4) is a combination of (0.15) and (0.16). This completes the proof. □

We changed only the initial vector, but the eigenvalues are kept, so the next proposition is a simple consequence of the last:

Proposition 3.2

Under the conditions of proposition 3.1 the Binet-representation of the recurrence with initial vector v_0' *is* $b'(t) = f(b(t))$.

Proof: We have already proved, that $b'(t) = f(b(t))$ holds for all $n \in \mathbb{N}$. The coefficients in the *Binet*-decomposition $v_0' = \alpha_1' e_1' + \alpha_2' e_2' + \alpha_2' e_2', \alpha_k' \in \mathbb{C}^3$ are unique, thus the proof is finished. □

We summarize our results:

- *Given an arbitrary conic section* Γ *in* $\mathbb{C}$, *we can construct a suitable recurrence and build a suitable initial vector* v_0 *so, that the resulting sequence* $\{z_n\}_{n\in\mathbb{N}}$ *is a discrete subset of* Γ.
- *In the case of a circle or an ellipse we con construct recurrences with suitable initial vectors and* $\{z_n\}_{n\in\mathbb{N}} \subset \Gamma$, *which are either periodic with a given period length* $m \in \mathbb{N}$, *or dense in* Γ.

Proof: We gave explicit constructions for circles, a parabola, and a hyperbola. Since all circles and all parabolas are similar, we can use proposition 3.1 (2) to map the special circle at any other circle or parabola by a similarity map $z \to a \cdot z + b$ with $a, b \in \mathbb{C}$. By proposition 3.1 (4) a circle and a hyperbola can be mapped at any arbitrary ellipse or hyperbola by an affine map $z \to a \cdot z + b \cdot \overline{z} + c$ with $a, b, c \in \mathbb{C}$. This proves the first statement.

To prove the second, we just need to start with a an eigenvalue of the form $\lambda = e^{i \cdot \alpha \cdot \pi}$ and $\alpha \in \mathbb{Q}$, as in the example above, for a periodic orbit, and $\alpha \in \mathbb{R} \setminus \mathbb{Q}$ for a dense orbit. □

This is the complete answer to Josef's question.

(4) The Case of Multiple Eigenvalues

Until now we have considered only examples with three distinct eigenvalues. It is even possible to give a *Binet*-representation in the case of multiple eigenvalues. In the following propositions M is always the matrix defined in (1.4).

Proposition 4.1.1

Let λ_1 be a simple and λ_2 be a double eigenvalue of the matrix M.

Set $e_1 = \begin{pmatrix} 1 \\ \lambda_1 \\ \lambda_1^2 \end{pmatrix}, e_2 = \begin{pmatrix} 1 \\ \lambda_2 \\ \lambda_2^2 \end{pmatrix}$, and $v = \begin{pmatrix} 0 \\ 1 \\ 2\lambda_2 \end{pmatrix}$. *Then the following statements hold:*

(1) $\langle e_1, e_2, v \rangle$ is a basis of $\mathbb{C}^3$ iff $\lambda_1 \neq \lambda_2$.

(2) $(M - \lambda_2)v = e_2$, or equivalently: $Mv = e_2 + \lambda_2 v$

(3) $M^n v = n \cdot \lambda_2^{n-1} \cdot e_2 + \lambda_2^n \cdot v$ for $n \geq 0$

Proof: (1) An easy calculation gives $\det(e_1, e_2, v) = (\lambda_2 - \lambda_1)^2 \neq 0 \Leftrightarrow \lambda_2 \neq \lambda_1$.

(2) We expand $(z - \lambda_1)(z - \lambda_2)^2 = z^3 - (\lambda_1 + 2\lambda_2) \cdot z^2 + (2\lambda_1\lambda_2 + \lambda_2^2) \cdot z - \lambda_1\lambda_2^2$ and see, that the parameters in the matrix M are $p = \lambda_1\lambda_2^2, q = -(2\lambda_1\lambda_2 + \lambda_2^2), r = \lambda_1 + 2\lambda_2$. We proceed:

$$(M - \lambda_2)v = \begin{pmatrix} -\lambda_2 & 1 & 0 \\ 0 & -\lambda_2 & 1 \\ p & q & r - \lambda_2 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 2\lambda_2 \end{pmatrix} = \begin{pmatrix} 1 \\ -\lambda_2 + 2\lambda_2 \\ -(2\lambda_1\lambda_2 + \lambda_2^2) + (\lambda_1 + \lambda_2) \cdot 2\lambda_2 \end{pmatrix} = \begin{pmatrix} 1 \\ \lambda_2 \\ \lambda_2^2 \end{pmatrix} = e_2$$

(3) As a consequence of (2) we have: $M^2 v = M(e_2 + \lambda_2 v) = \lambda_2 e_2 + \lambda_2 M v = 2\lambda_2 e_2 + \lambda_2^2 v$, and by induction $M^n v = n \cdot \lambda_2^{n-1} \cdot e_2 + \lambda_2^n \cdot v$.□

We notice, that this proposition is true, even if one eigenvalue is zero. The next one settles the case of a triple eigenvalue.

Proposition 4.1.2

Let $\lambda \neq 0$ be a triple eigenvalue of M and $e = \begin{pmatrix} 1 \\ \lambda \\ \lambda^2 \end{pmatrix}, v_1 = \begin{pmatrix} 0 \\ 1 \\ 2\lambda \end{pmatrix}$, and $v_2 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$. *Then:*

(1) $\langle e, v_1, v_2 \rangle$ *is a basis of* $\mathbb{C}^3$.

(2) $(M-\lambda)v_1 = e$ *or equivalently:* $Mv_1 = \lambda \cdot v_1 + e$

(3) $(M-\lambda)v_2 = v_1$ *or equivalently:* $Mv_2 = \lambda \cdot v_2 + v_1$

Proof: (1) We calculate $\det(e, v_1, v_2) = 1 \neq 0$.
(2) Since λ is a triple eigenvalue, we find for the parameters in M: $p = \lambda^3, q = -3\lambda^2, r = 3\lambda$.

So $$(M-\lambda)v = \begin{pmatrix} -\lambda & 1 & 0 \\ 0 & -\lambda & 1 \\ \lambda^3 & -3\lambda^2 & 2\lambda \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 2\lambda \end{pmatrix} = \begin{pmatrix} 1 \\ -\lambda + 2\lambda \\ -3\lambda^2 + 4\lambda^2 \end{pmatrix} = \begin{pmatrix} 1 \\ \lambda \\ \lambda^2 \end{pmatrix} = e$$ follows.

(3) A similar calculation proves (3). □

Now we have all pieces together to state

Proposition 4.1.3

Let the notations be as above and let $w := \alpha_1 \cdot e_1 + \alpha_2 \cdot e_2 + \alpha_3 \cdot v, \alpha_k \in \mathbb{C}$ for $k = 1,2,3$.
Then in the following formulae are valid:

In the case of a double eigenvalue λ_2*:*

(1) $M^n w = \alpha_1 \lambda_1^n \cdot e_1 + \left(\alpha_2 \cdot \lambda_2^n + \alpha_3 \cdot n \cdot \lambda_2^{n-1}\right) \cdot e_2 + \alpha_3 \cdot \lambda_2^n \cdot v$ for $n \in \mathbb{N}_0$

(2) If $\lambda_2 \neq 0$, *then* $M^n w = \alpha_1 \lambda_1^n \cdot e_1 + \lambda_2^n \left(\left(\alpha_2 + \alpha_3 \cdot \frac{n}{\lambda_2} \right) \cdot e_2 + \alpha_3 \cdot v \right)$ for $n \in \mathbb{N}_0$

(3) If $\lambda_2 = 0$, *then* $M^n w = \begin{cases} \alpha_1 \cdot e_1 + \alpha_2 \cdot e_2 + \alpha_3 \cdot v & \text{for } n = 0 \\ \alpha_1 \lambda_1 \cdot e_1 + \alpha_3 \cdot v & \text{for } n = 1 \\ \alpha_1 \lambda_1^n \cdot e_1 & \text{for } n \geq 2 \end{cases}$

In the case of a triple eigenvalue $\lambda \neq 0$*:*

(4) $M^n v_1 = \lambda^n \cdot v_1 + n \cdot \lambda^{n-1} \cdot e$ for $n \geq 0$

(5) $M^n v_2 = \lambda^n \left(\frac{n(n-1)}{2} \cdot e + n \cdot v_1 + v_2 \right)$ for $n \geq 0$

(6) $M^n w = \lambda^n \left((n\alpha_1 + \frac{n(n-1)}{2} \alpha_2 + \alpha_3) \cdot e + (\alpha_1 + n\alpha_2) \cdot v_1 + \alpha_2 \cdot v_2 \right)$ for $n \geq 0$

Proof: (1) is a consequence of proposition 4.1.1 part (3). (2) is equivalent to (1) if $\lambda_2 \neq 0$.
(3) If $\lambda_2 = 0$, then

$$Me_2 = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & \lambda_1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, Mv = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & \lambda_1 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ \lambda_1 \end{pmatrix}, \text{ and } M^2 v = \begin{pmatrix} 1 \\ \lambda_1 \\ \lambda_1^2 \end{pmatrix} = e_1.$$

Calculating Mw and $M^2 w$ proves (3).
(4) From 4.1.2 part (2) we have $M^2 v_1 = \lambda^2 + \lambda(\lambda \cdot e + \lambda \cdot v_1) = 2\lambda^2 \cdot e + \lambda^2 \cdot v_1$ and (4) follows by induction. (5) follows from proposition 4.1.2 part (3) and the previous statement.
(6) is the result of the linearity of the powers of M. □

We choose the initial vector $w = \begin{pmatrix} z_0 \\ z_1 \\ z_2 \end{pmatrix}$ and pass in the previous proposition to the components of the resulting vector $M^n w$. This leads us to

Proposition 4.2 (*Binet*-representation for multiple eigenvalues)

Let λ_2 be a double eigenvalue and let $\alpha_1, \alpha_2, \alpha_3$ be the coefficients of the Binet-*decomposition of the initial vector .*

(1) If $\lambda_2 \neq 0$, then $z_n = \alpha_1 \lambda_1^n + \alpha_2 \lambda_2^n \left(1 + \frac{n}{\lambda_2}\right)$ for $n \geq 0$

(2) If $\lambda_2 = 0$, then $z_n = \alpha_1 \lambda_1^n$ for $n \geq 2$

Let λ be a triple eigenvalue, $\lambda \neq 0$. Then

(3) $z_n = \alpha_1 \cdot \lambda^n + n \cdot \alpha_2 \cdot \lambda^{n-1} + \frac{n(n-1)}{2} \cdot \alpha_3 \cdot \lambda^{n-2}$ for $n \geq 0$

Remarks: (1) We note, that only (1) and (3) admit a continuous interpolation in the sense, that we get z_0, z_1, z_2 for $t = 0, t = 1, t = 2$ if we replace the discrete variable n by the continuous variable $t \in [0, \infty)$.
(2) If we had chosen $\lambda = 0$ as a triple eigenvalue, then we had $z_n = 0$ for all $n \geq 3$, whatever the initial values had been.
(3) In the first case, convergence or divergence of the sequence $\{z_n\}_{n\in\mathbb{N}}$ can depend on the eigenvalues and on the initial vector. In the other cases (2) and (3) the convergence depends only on the eigenvalues λ_1 resp. λ.
(4) It is no problem, to write a *DERIVE*-function, which calculates the appropriate *Binet*-representations for all different cases. We present the *DERIVE*-code and two examples:

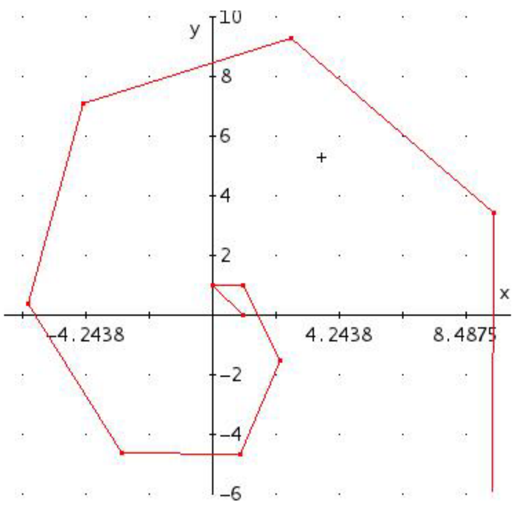
```
aux1(p, t, v) ≔
  Prog
    e ≔ EIGENVALUES([[0, 1, 0], [0, 0, 1], p])
    w ≔ VECTOR(SUBST([1; r; r^2], r, e↓k), k, 1, 3)
    (SOLUTIONS([α, β, γ]·w = [v↓1; v↓2; v↓3], [α, β, γ])·VECTOR(IF(k ≠ 0, k^t, 0, 0), k, e))↓1


aux2a(p, n, v) ≔
  Prog
    w ≔ [1, 1, 0; p↓1, p↓2, 1; p↓1^2, p↓2^2, 2·p↓2]
    (SOLUTIONS(w·[a, b, c] = v, [a, b, c]))↓1·[IF(p↓1 = 0, 0, p↓1^n), p↓2^n, n·p↓2^(n - 1)]

aux2b(p, n, v) ≔
  Prog
    w ≔ [1, 1, 0; p↓1, 0, 1; p↓1^2, 0, 0]
    (SOLUTIONS(w·[a, b, c] = v, [a, b, c]))↓1·[p↓1^n, 0, 0]

aux2(p, n, v) ≔
  If p↓2 = 0
     aux2b(p, n, v)
     aux2a(p, n, v)

aux3(p, n, v) ≔
  Prog
    w ≔ [1, 0, 0; p↓1, 1, 0; p↓1^2, 2·p↓1, 1]
    (SOLUTIONS(w·[a, b, c] = v, [a, b, c]))↓1·[p↓1^n, n·p↓1^(n - 1), n·(n - 1)/2·p↓1^(n - 2)]
```

```
binet(p, n, v) ≔
  Prog
    s ≔ SOLUTIONS(x^3 = p·[1, x, x^2], x)
    w ≔ IF(s↓1 + 2·s↓2 = p↓3, s, REVERSE(s))
    l ≔ DIMENSION(s)
    If l = 3
       aux1(p, n, v)
       If l = 2
          aux2(w, n, v)
          aux3(s, n, v)

comptoreal(z) ≔ [RE(z), IM(z)]

pointseq(p, n, m, v) ≔ VECTOR(comptoreal(binet(p, k, v)), k, n, m)
```

First example:

We take the single eigenvalue $\lambda_1 = 0.6 + 0.8i$ and the double eigenvalue $\lambda_1 = 0.6 - 0.8i$. This gives $p = 0.6 - 0.8i, q = -1.72 + 0.96i, r = 1.8 - 0.8i$. We omit the output of the resulting list of points but print only the command and the graphical representation of the result:

```
pointseq([0.6 - 0.8·i, -1.72 + 0.96·i, 1.8 - 0.8·i], 0, 10, [1, i, 1 + i])
```

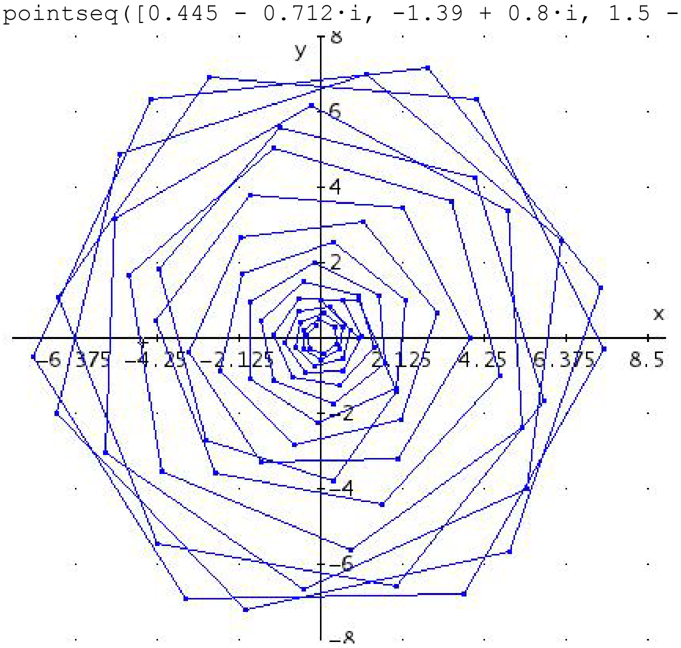


We notice, that the sequence of points is divergent. The reason is, that the factor $\left(1 + \frac{n}{\lambda_2}\right)$ in $z_n = \alpha_1 \lambda_1^n + \alpha_2 \lambda_2^n \left(1 + \frac{n}{\lambda_2}\right)$ is unbounded, if n tends to infinity, though $\left|\lambda_1^n\right| = \left|\lambda_2^n\right| = 1$ for all n.

Second example:

This example works with $\lambda_1 = 0.5 + 0.8i$ and $\lambda_1 = 0.5 - 0.8i$, so the absolute values are just below 1:

```
pointseq([0.445 - 0.712·i, -1.39 + 0.8·i, 1.5 - 0.8·i], 0, 100, [1, i, 1 + i])
```



We can see, that the values are at first increasing, but since $\lim_{n \to \infty} \alpha_1 \lambda_1^n + \alpha_2 \lambda_2^n \left(1 + \frac{n}{\lambda_2}\right) = 0$, the sequence of points is convergent to the origin.

Finally we look at limits of quotient sequences $\left\{\frac{z_{n+1}}{z_n}\right\}_{n \in \mathbb{N}}$, because the *Binet*-representations allow us to give exact conditions on the convergence.

As before, we must distinguish three cases:

(i) *The characteristic equation of the recurrence has three distinct roots. Then, roughly speaking, the eigenvalue with largest absolute value, which is present in the Binet-representation, wins: Let* $|\lambda_1| > |\lambda_2| \geq |\lambda_3|$ *, then by proposition 2.2*

$\frac{z_{n+1}}{z_n} = \frac{\alpha_1\lambda_1^{n+1} + \alpha_2\lambda_2^{n+1} + \alpha_3\lambda_3^{n+1}}{\alpha_1\lambda_1^{n} + \alpha_2\lambda_2^{n} + \alpha_3\lambda_3^{n}}$. *Since* $\left|\frac{z_2}{z_1}\right| < 1$ and $\left|\frac{z_3}{z_1}\right| < 1$, *the limit is*

$$\lim_{n\to\infty}\frac{z_{n+1}}{z_n} = \lim_{n\to\infty}\frac{\lambda_1^{n+1}\left(\alpha_1 + \alpha_2(\lambda_2/\lambda_1)^{n+1} + \alpha_3(\lambda_3/\lambda_1)^{n+1}\right)}{\lambda_1^{n}\left(\alpha_1 + \alpha_2(\lambda_2^{n}/\lambda_1) + \alpha_3(\lambda_3/\lambda_1)^{n}\right)} = \lambda_1$$

(ii) *The characteristic equation has two distinct roots. Then again the eigenvalue with largest absolute value, which is present in the Binet-representation, wins. If the eigenvalues are different in absolute value, then let* λ_M *be the eigenvalue with larger absolute value. We find by proposition 4.2 and similar considerations as above, that* $\lim_{n\to\infty}\frac{z_{n+1}}{z_n} = \lim_{n\to\infty}\frac{\alpha_1\lambda_1^{n+1} + \alpha_2\lambda_2^{n+1}\left(1+\frac{n+1}{\lambda_2}\right)}{\alpha_1\lambda_1^{n} + \alpha_2\lambda_2^{n}\left(1+\frac{n}{\lambda_2}\right)} = \lambda_M$.

(iii) *In the case of a triple eigenvalue we get by proposition 4.2, that*

$$\lim_{n\to\infty}\frac{z_{n+1}}{z_n} = \lim_{n\to\infty}\frac{\alpha_1\cdot\lambda^{n+1} + n\cdot\alpha_2\cdot\lambda^{n} + \frac{n(n+1)}{2}\cdot\alpha_3\cdot\lambda^{n-1}}{\alpha_1\cdot\lambda^{n} + n\cdot\alpha_2\cdot\lambda^{n-1} + \frac{n(n-1)}{2}\cdot\alpha_3\cdot\lambda^{n-2}} = \lambda\cdot\lim_{n\to\infty}\frac{\alpha_1\cdot\lambda^{2} + n\cdot\alpha_2\cdot\lambda + \frac{n(n+1)}{2}\cdot\alpha_3}{\alpha_1\cdot\lambda^{2} + n\cdot\alpha_2\cdot\lambda + \frac{n(n-1)}{2}\cdot\alpha_3} = \lambda$$

Comments: In the case of two or three distinct eigenvalues a limit must not exist, while in the case of a triple eigenvalue a limit always exists, if $\lambda \neq 0$. If $\lambda = 0$, then $z_n = 0$ for $n \geq 3$ and the quotients are not defined. Note, that the presence of a particular eigenvalue in the *Binet*-representation depends on the initial values.

We finish with some number theoretic considerations: The *Binet*-decomposition of the initial vector and the *Binet*-representation together show, that the following proposition holds:

Proposition 4.3

Let $F(p,q,r,z_0,z_1,z_2)$ *be an intermediary number field* $\mathbb{Q} \subseteq F(p,q,r,z_0,z_1,z_2) \subset \mathbb{C}$, *which contains the coefficients* p,q,r *of the recurrence and the initial values* z_0,z_1,z_2. *Then the following statements hold:*

(1) $\{z_n\}_{n\in\mathbb{N}} \subset F(p,q,r,z_0,z_1,z_2)$

(2) If one of the limits $\lim_{x\to\infty} z_n$ *or* $\lim_{x\to\infty}\frac{z_{n+1}}{z_n}$ *exists, then this limit belongs to the unique field extension* $\widetilde{F}$ *of* $F(p,q,r,z_0,z_1,z_2)$ *of at most degree three, which is generated by the roots of the characteristic equation of the recurrence.*

Proof: (1) The definition of the recurrence uses only the four rational operations, so all the z_n stay inside $F(p,q,r,z_0,z_1,z_2)$. (2) The eigenvalues are roots of the characteristic polynomial, so a standard argument from *Galois*-theory says, that they belong to an extension field of at most degree three. The coefficients in the *Binet*-decomposition are solutions of a system of linear equations in this extension field, so they must belong to this field too. The limits, if they exist, depend only on the coefficients of the *Binet*-decomposition and on the roots of the characteristic equation. So they belong to the extension field. □

Let us consider Carl's and Josef's example once again: The initial values are pairs of rational numbers, or in complex notation, numbers with rational real and imaginary parts. The same is true for the eigenvalues (2.7). As a consequence, the limit $z_\infty = 0.4 + 0.4i$ has rational real and imaginary part.

(5) A Theorem on Coefficients

We only mention, that everything presented here can easily and in a straightforward manner be generalized to higher dimensional recurrences

$$z_{n+m} = \sum_{k=1}^{m} a_k z_{n+k} \quad \text{with initial values } z_0, z_1, \ldots, z_{m-1} \in \mathbb{C}.$$

Thus we state the next propositions in full generality. The convergence of the recurrences depends generally on the eigenvalues, they depend on the coefficients $a_1, a_2, \ldots, a_m$ of parameters and on the initial vector $v_0 = (z_0, z_1, \ldots, z_m)$. In a very particular case the convergence is guaranteed by a condition on the coefficients of the recursion. To state the proposition precisely, we introduce the following notation: $\delta(a_k) := \begin{cases} k & \text{if } a_k \neq 0 \\ 0 & \text{if } a_k = 0 \end{cases}$.

Proposition 5.1

Let $p(z) := \sum_{k=0}^{n-1} a_k z^k$ be a polynomial with nonnegative coefficients $a_k \geq 0$ for $k = 1, .., n-1$, and $\sum_{k=0}^{n-1} a_k =: s$. Then the following statements are true:

(1) If $s = 1$ and z is a solution of the equation $z^n = p(z)$, then $|z| \leq 1$. In particular $z_1 = 1$ is a solution.

(2) There exists a positive solution ξ of the equation $z^n = p(z)$, and for all other solutions z we have $|z| \leq \xi$.

(3) If $s < 1$, then there exists a positive solution $\xi < 1$ of the equation $z^n = p(z)$, and for all other solutions z we have $|z| \leq \xi < 1$.

(4) If all coefficients are nonnegative, $s = 1$, and $m := \gcd(\delta(a_0), \delta(a_1), \ldots, \delta(a_{n-1}))$, then the equation $z^n = p(z)$ has exactly m solutions $z_j, j = 1, \ldots, m,$ with $|z_j| = 1$. All other solutions satisfy $|z| < 1$.

Proof: (1) We assume, that there is a solution z with $|z| > 1$. We obtain the following chain of equalities/inequalities: $|z^n| = \left|\sum_{k=0}^{n-1} a_k z^k\right| \leq \sum_{k=0}^{n-1} |a_k| \cdot |z^k| \leq s \cdot |z^{n-1}| = |z^{n-1}| \Leftrightarrow |z| \leq 1$. This is a contradiction to $|z| > 1$ and the first part is proved. Clearly $z = 1$ is a solution because we have $s = 1$.

(2) We first show, that exactly one positive solution exists. Consider $g(x) := \dfrac{x^n}{\sum_{k=0}^{n-1} a_k x^k}$. After possibly cancelling equal powers of x we have: $g(0)=0$. On the other hand $\lim_{x\to\infty} g(x) = \infty$, so, by continuity, there must be a real number $\xi > 0$ with $g(\xi)=0$. Now let z be another solution, then division by ξ^n gives $\left(\dfrac{z}{\xi}\right)^n = \sum_{k=0}^{n-1} \dfrac{a_k}{\xi^{n-k}} \left(\dfrac{z}{\xi}\right)^k$. We substitute $\theta := \dfrac{z}{\xi}$ and obtain: $\theta^n = \sum_{k=0}^{n-1} \dfrac{a_k}{\xi^{n-k}} \theta^k$. $\theta = 1$ is a solution of this last equation, so by proposition 1.1: $\sum_{k=0}^{n-1} \dfrac{a_k}{\xi^{n-k}} = 1$. Since $\xi > 0$, we find by (1), that $\left|\dfrac{z}{\xi}\right| \le 1$ or equivalently: $|z| \le \xi$. This also proves the uniqueness of ξ.

(3) We only need to prove, that $\xi < 1$ if $s < 1$. Assume, that $\xi \ge 1$. Then $\dfrac{1}{\xi^{n-p}} \le \dfrac{1}{\xi}$, and as a consequence $1 = \sum_{k=0}^{n-1} \dfrac{a_k}{\xi^{n-k}} \le \sum_{k=0}^{n-1} \dfrac{a_k}{\xi} \le \dfrac{s}{\xi}$ and equivalently $\xi \le s < 1$. This ia a contradiction, so we have proved $\xi < 1$.

(4) If the gcd equals 1, then the statement is a theorem of Ostrovsky [2, p. 3], which can be proved along the lines of (1) and (2). The condition on the the gcd has the following interpretation: Let $m := \gcd(\delta(a_0), \delta(a_1), \ldots, \delta(a_{n-1}))$, then $z^n - p(z) = q(z^m)$ is a polynomial of z^m and $q(z) := z^v - \sum_{k=0}^{\upsilon-1} b_k \cdot z^k$. Obviously the coefficients satisfy $b_k = a_{m\cdot k}$ and $\sum_{k=0}^{\upsilon-1} b_k = 1$. Now we have $\gcd(\delta(b_0), \ldots, \delta(a_{\upsilon-1}), \upsilon) = 1$, so by Ostrovsky's theorem $z = 1$ is the unique solution of $q(z) = 0$ with $|z| = 1$. Thus the numbers $z_j := e^{2\pi i \cdot j/m}, j \in \{0, 1, \ldots, m-1\}$ are the only solutions of $z^n = p(z)$ with $|z| = 1$ and the proposition is completely proved.□

The following corollaries are immediate consequences of proposition 5.1.

Corollary 5.2

Let $z_{n+m} = \sum_{k=1}^{m} a_k z_{n+k}$ *be a recurrence with initial values* $z_0, z_1, \ldots, z_{m-1} \in \mathbb{C}$, *nonnegative coefficients* a_k, $\sum_{k=0}^{n-1} a_k = 1$, *and* $\gcd(\delta(a_0), \delta(a_1), \ldots, \delta(a_{n-1})) = 1$. *Then the sequence* $\{z_n\}_{n\in\mathbb{N}}$ *converges.*

Proof: The conditions stated above guarantee, that there is exactly one simple eigenvalue $\lambda_1 = 1$ and all other eigenvalues satisfy $|\lambda_k| < 1, k \in \{2, 3, \ldots, n\}$. So even in the case of multiple eigenvalues the corresponding *Binet*-representation is convergent, and the convergence is independent of the initial values, because the powers of the eigenvalues tend to zero as the exponents tend to infinity, for $k \in \{2, 3, \ldots, n\}$, and to 1 for $\lambda_1 = 1$.□

Corollary 5.3

Let $z_{n+m} = \sum_{k=1}^{m} a_k z_{n+k}$ *be a recurrence with initial values* $z_0, z_1, \ldots, z_{m-1} \in \mathbb{C}$*, nonnegative coefficients* a_k*, and* $\sum_{k=0}^{n-1} a_k = s < 1$*. Then the sequence* $\{z_n\}_{n\in\mathbb{N}}$ *converges to zero.*

Proof: By proposition 5.1. (3) all eigenvalues satisfy $|\lambda_k| < 1$, and consequently all powers in the corresponding *Binet*-representation converge to zero whatever the initial values are. □

Remarks: (1) This corollary was proved for $n = 3$ with different means as proposition 1.2 . Josef's example from [1, p. 5] with $p = a_0 = q = a_1 = 0.5$ and $r = a_2 = 0$ satisfies the conditions of corollary 5.2 : $p + q + r = 1$ and $\gcd(0,1,3) = 1$, so the sequence is convergent with every initial triangle.
(3) The condition on the gcd in corollary 5.2 cannot be omitted, because this condition on the gcd is necessary to guarantee convergence, as the following example will show. Let $n = 3$, $a_1 = 1$, and $a_2 = a_3 = 0$. This is the case of the triple eigenvalue $\lambda = 1$. We have the *Binet*-representation:

$$z_n = \alpha_1 \cdot 1^n + n \cdot \alpha_2 \cdot 1^{n-1} + \frac{n(n-1)}{2} \cdot \alpha_3 \cdot 1^{n-2} = \alpha_1 + n \cdot \alpha_2 + \frac{n(n-1)}{2} \cdot \alpha_3$$

So $\lim_{n\to\infty} z_n = \infty$ if α_2 and α_3 do not vanish both. It is possible to construct examples for $n = 4$ which are bounded or even periodic but not convergent.

(6) Conclusions

A small shift from points in the plane to complex numbers allows us to use the machinery of linear algebra. Especially the use of eigenvalues of the matrix, which describes the recurrence, allows us to understand some geometrically surprising phenomena occurring with computer experiments, such as points on a circle or ellipse. The corresponding *Binet*-representation allows to build something like a construction kit for recurrences with a prescribed behaviour. The *Binet*-representation gives us complete information about convergence or divergence of the sequence of points defined by recurrences. The convergence/divergence is governed by the eigenvalues, resp. the roots of the characteristic equation and all we have to look for, is $\lim_{n\to\infty} R(n) \cdot \lambda^n$, where $R(n)$ is a rational function of n . Finally a CAS, like *DERIVE*, allows us to visualize the theory of recurrences.

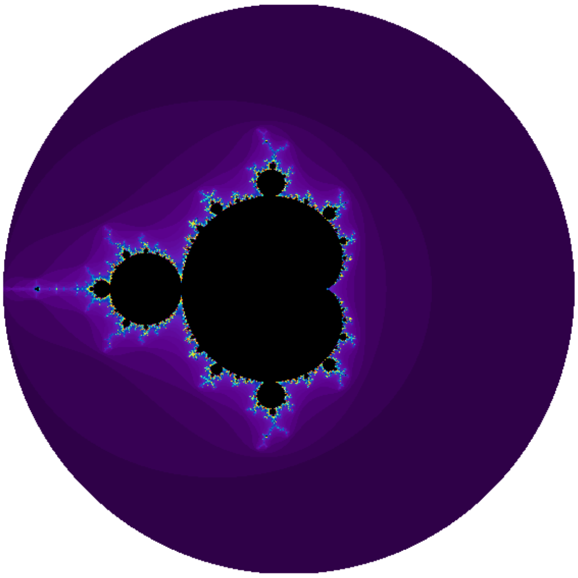
(7) References

[1] Josef Böhm & Carl Lewis Leinbach, *Using Rational Arithmetic to Develop a Proof*, DNL # 80, 2011

[2] Victor V. Prasolov, *Polynomials*, Springer, Algorithms and Computation in Mathematics, Volume 11, 1999

Our long time DUG Member Robert Setif sent a mail including a picture of the famous Mandelbrot Set – the Apple Man:

Bonjour.
Can Derive program a function that creates an image such as the attached file?
Thank you and best regards



Robert

It is mere chance that I had also an exchange of mails with Rob Gough concerning the famous Mandelbrot set. He sent a DERIVE file and complained about boring calculation times.

Dear Josef,

However, of greater importance to me is my ability to program in Derive (for my own pleasure). With a lot of difficulty (because the Help functions don't work properly) I managed to create a program to produce the Henon Attractor (Chaos Theory). So my next task was to produce the Mandelbrot set. This followed on relatively easily from the previous work, but I could not produce a graphical presentation of this. I know how to produce the data but cannot display it! I produced a 2D iteration based on x:=x^2 –y^2+a, y:=2*x*y+b that produced a 3D vector [a,b,c] where c is a measure of the stability of the iteration. My intention was to use the third axis to represent the colour of the 2D [a,b] plot. Can I do this? If it would help I could enclose a pdf file of the work I have done.

Best wishes

Rob

The Mandelbrot Set

Rob Gough

The Mandelbrot Set defines a region where a certain recursive function remains stable and does not produce values heading for infinity. An outline is shown in 'Chaos Under Control' by D Peak and M Frame, 1994. The shape is mainly confined to a region with -2<a<+0.5 on the horizontal axis (a-axis)and -1<b<+1 on the vertical axis (b-axis). The aim is to concentrate on any part of this 2D plot within the limits [A1, A2, B1, B2] where A1<a<A2 and B1>b>B2 with any desired pixel detail (□). To create a linear sequence of coordinates suitable for the Derive VECTOR function it is necessary to scan the desired region in the fashion of a CRT display from top left [A1,B1] to bottom right [A2,B2].

#1: $$\left[K := 1 + \frac{A2 - A1}{\delta},\ L := 1 + \frac{B1 - B2}{\delta},\ J := K \cdot L\right]$$

#2: $$[k :\in \text{Integer},\ l :\in \text{Integer},\ j :\in \text{Integer}]$$

#3: $$[0 \le k \le K,\ 0 \le l \le L,\ 0 \le j \le J - 1]$$

#4: $$\left[l := \text{FLOOR}\left(\frac{j}{K}\right),\ k := j - l \cdot K\right]$$

#5: $$\left[a := A1 + \delta \cdot \left(j - \text{FLOOR}\left(\frac{j}{K}\right) \cdot K\right),\ b := B1 - \delta \cdot \text{FLOOR}\left(\frac{j}{K}\right)\right]$$

#6: $$[a := A1 + \delta \cdot k,\ b := B1 - \delta \cdot l]$$

The 2-dimensional [a,b] plot can be coloured with reference to the stability criterion of the recursion. This is based on five numbers [C1, C2, C3, C4, C5] where C1<C2<C3<C4<C5 with C1 the least stable and C5 the most stable. For example, if the recursion is still within certain limits after C5 iterations then the parameters [a,b] is considered stable and the third dimension records '5' and the coordinates [a,b,5] are returned.

The recursion is based on the dummy variables [x,y]:

x:= x^2 -y^2 +a

y:= 2*x*y + b

and the stability criterion is based on x^2 + y^2 <4.

```
#7:   [x :=, y :=]

      XY_AUX(A1, A2, B1, B2, δ, n, x, y) :=
        If n = 0
#8:        [x, y]
           XY_AUX(A1, A2, B1, B2, δ, n - 1, x^2 - y^2 + a, 2·x·y + b)

#9:   XY(A1, A2, B1, B2, δ, n) := XY_AUX(A1, A2, B1, B2, δ, n, 0, 0)
```

#10: $$\text{DMAX}(A1, A2, B1, B2, \delta, n) := \text{XY}(A1, A2, B1, B2, \delta, n)^2$$

```
#11:  levs0 := [2, 5, 10, 20, 50]

      M(A1, A2, B1, B2, δ, C1, C2, C3, C4, C5, a, b) :=
        Prog
          n := 0
          Loop
            If 0 ≤ n < C1 ∧ DMAX(A1, A2, B1, B2, δ, C1, C2, C3, C4, C5, a, b, n) ≥ 4
               RETURN [a, b, 0]
               If C1 ≤ n < C2 ∧ DMAX(A1, A2, B1, B2, δ, C1, C2, C3, C4, C5, a, b, n) ≥ 4
                  RETURN [a, b, 1]
#12:              If C2 ≤ n < C3 ∧ DMAX(A1, A2, B1, B2, δ, C1, C2, C3, C4, C5, a, b, n) ≥ 4
                     RETURN [a, b, 2]
                     If C3 ≤ n < C4 ∧ DMAX(A1, A2, B1, B2, δ, C1, C2, C3, C4, C5, a, b, n) ≥ 4
                        RETURN [a, b, 3]
                        If C4 ≤ n < C5 ∧ DMAX(A1, A2, B1, B2, δ, C1, C2, C3, C4, C5, a, b, n) ≥ 4
                           RETURN [a, b, 4]
                           If n = C5
                              RETURN [a, b, 5]
            n := n + 1

#13:  MB(A1, A2, B1, B2, δ, C1, C2, C3, C4, C5, j) := M(A1, A2, B1, B2, δ, C1, C2, C3, C4, C5, a, b)

#14:  MBS(A1, A2, B1, B2, δ, C1, C2, C3, C4, C5) := VECTOR(MB(A1, A2, B1, B2, δ, C1, C2, C3, C4, C5, j), j, 0, J - 1)
```

Lastly, these 3-dimensional coordinates [a,b,c] are returned as a vector sequence of length J.

Here is a more detailed example:

```
#25:  [A1 := -1.3, B1 := 0.4, A2 := -0.7, B2 := -0.4, δ := 0.02]

#26:  [C1 := 5, C2 := 10, C3 := 20, C4 := 50, C5 := 100, n := 50]

#27:  MBS(-1.3, -0.7, 0.4, -0.4, 0.02, 5, 10, 20, 50, 100)

#28:  [[-1.3, 0.4, 1], [-1.28, 0.4, 1], [-1.26, 0.4, 2], [-1.24, 0.4, 2], [-1.2

        1], [-1.12, 0.4, 1], [-1.1, 0.4, 1], [-1.08, 0.4, 1], [-1.06, 0.4, 1],
```

#28 consists of 1271 triples.

My first answer was as follows:

I introduce variable `set1` for the list of points (which is not printed here).

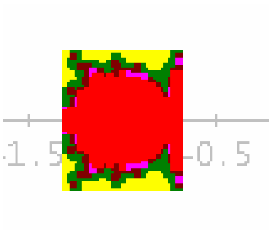
```
#26:  set1 := MBS(-1.3, -0.7, 0.4, -0.4, 0.02, 5, 10, 20, 50, 100)

#27:  set1 := [[-1.3, 0.4, 1], [-1.28, 0.4, 1], [-1.26, 0.4, 2], [-1.24, 0.4

        [-1.14, 0.4, 1], [-1.12, 0.4, 1], [-1.1, 0.4, 1], [-1.08, 0.4, 1],
```

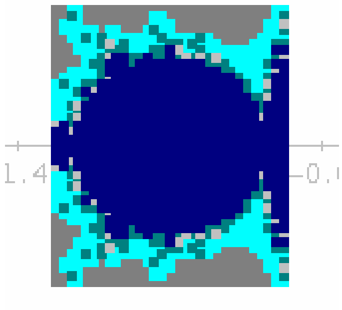
Calculation of #26 needs a very long time (7000 seconds!!). I believe that the reason is the recursive nature of # 8. I am quite sure that this should (and could) be improved.

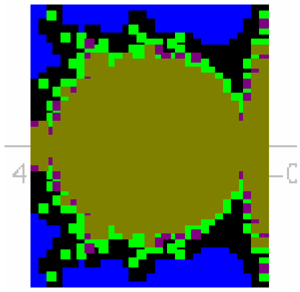
This is the trick: The SELECT-command separates the points according to the depth of iteration and extracts the first two columns (x,y) making the points ready for plot. The resulting matrices are plotted in different colours. Make sure that the points are plotted discrete and you may change the point size.

```
#28:  set11 := VECTOR((SELECT(v  = k_, v, set1))↓↓[1, 2], k_, 0, 5)
                            3
```

Expression #28 can be plotted in the 2D Plot Window. It is not necessary approximating #17 but take care that in the Plot Window Options > Approximating Before Plotting is activated. Each Plot results in another combination of colours.



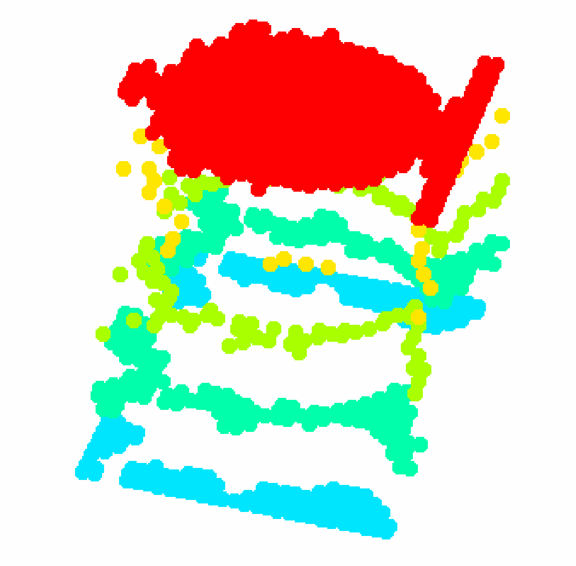


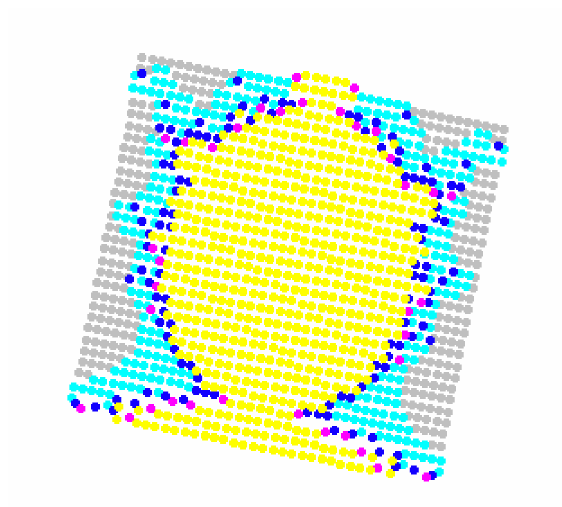


Now, the 3D-presentation: I apply SELECT again, but the result does not lead to a satisfying plot (#29), because all points are connected.

```
      set111 := VECTOR(SELECT(v  = k_, v, set1), k_, 0, 5)
#29:                           3

      set13d := VECTOR(VECTOR([v], v, set111 ), k, DIM(set111))
#30:                                        k
```

The command in #30 produces single points collected according to their levels (0 – 5) in matrices (lists) which are ready for plotting in 3D.





Point Size Large (Left) and the Top View with Point Size Medium (right)

As the readers might assume I was not really satisfied with my first answers. Not because of the resulting plots but because of the enormous calculation times. I tried to do better which was not really nasty for me because I have been interested and fascinated by all kinds of fractals (see earlier Newsletters).

I was sure that I had to overcome the recursive procedure by writing a program including all necessary parameters in the parameter list of the function (program) to be developed.

I define some "levels of escape". `pos()` relates the number of the "escaping iteration" to the position of the escape level in the list. Given `levs1` and iteration #45 escapes, then `pos(levs1,45)` = 3 and `pos(levs5,45)` = 4 but `pos(levs2,45)` = 1, etc.

```
#1:   levs := [2, 5, 10, 20, 50]

#2:   levs1 := [5, 10, 20, 50, 100]

#3:   levs2 := [50, 100, 130, 170, 200]

#4:   levs3 := [100, 130, 170, 200, 300]

#5:   levs4 := [5, 10, 20, 50, 75, 100, 150]

#6:   levs5 := [5, 10, 20, 40, 50, 60, 70, 80, 90, 100]

#7:   pos(l_, n) := POSITION(true, VECTOR(m - n ≥ 0, m, l_))
```

`crit()` is the core part of my procedure. It returns the initial point [a,b] together with its "escaping number" – depending on the level list l_ – which is later associated with a plot colour.

```
      crit(a, b, l_, z0, z1, n, n_) :=
        Prog
          z0 := a + î·b
          n := MAX(levs)
          n_ := 1
          Loop
#7:         If n_ > n
               RETURN [a, b, DIM(l_)]
            z1 := z0^2 + a + b·î
            If ABS(z1) > 2
               RETURN [a, b, pos(l_, n_)]
            z0 := z1
            n_ :+ 1
```

Expression #8 scans the region for a1 ≤ x ≤ a2 and b1 ≤ y ≤ b2 in δ-steps.

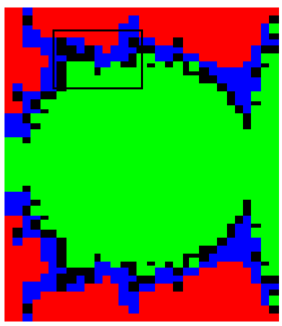
```
#8:   MS(a1, b1, a2, b2, δ, l_, z0, z1, n, n_, x_, y_) :=

APPEND(VECTOR(VECTOR(crit(a, b, l_, z0, z1, n, n_), a, a1, a2, δ), b, b2, b1, δ))
```

Let's see if it works and if there is really a success in speeding up the calculation:

```
#11:  set := MS(-1.3, 0.4, -0.7, -0.4, 0.02, [5, 10, 20, 50, 100])
```

needs only 43 seconds!!

I don't print the set `set.` In fact I removed it for saving memory!



```
#12:  VECTOR((SELECT(v  = k_, v, set))↓↓[1, 2], k_, 5)
                      3
```

The nice thing is zooming in. So I zoom into the black rectangle top right.

To make it easier I invented a function for plotting.

```
#13:  plot(s_, l_) := VECTOR((SELECT(v↓3 = k_, v, s_))↓↓[1, 2], k_, DIM(l_) + 1)

#14:  plot(set, levs5)
```

This gives the same picture as above. Then I deleted the point list `set` in order to save memory.

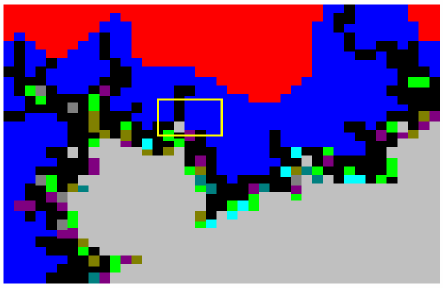
```
#15:  set :=
```

What is following was all done using this procedure. But at a certain moment I considered that formatting the out put and performing the output needs also calculation time and memory. In fact I do not need the list of the points, I only want them to be plotted. So I wrote a very short program combining calculation and plotting as well:

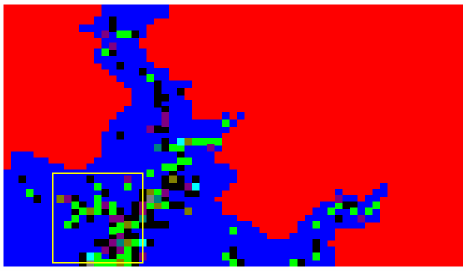
```
      Mprog(a1, b1, a2, b2, δ, l_, z0, z1, n_, n, x_, y_, set) :=
        Prog
#16:      set := APPEND(VECTOR(VECTOR(crit(a, b, l_, z0, z1, n_, n), a, a1, a2, δ), b, b2, b1, δ))
          VECTOR((SELECT(v↓3 = k_, v, set))↓↓[1, 2], k_, DIM(l_) + 1)
```

Using `Mprog` I started the journey into the interior of the boundary region:

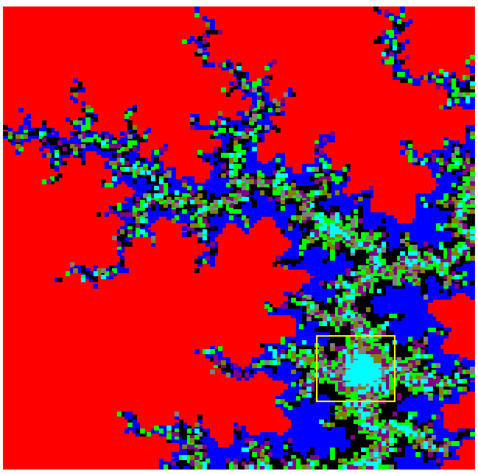
```
#17:  Mprog(-1.2, 0.35, -1, 0.2, 0.005, levs5)

needs 44 sec for calculating and plotting
```

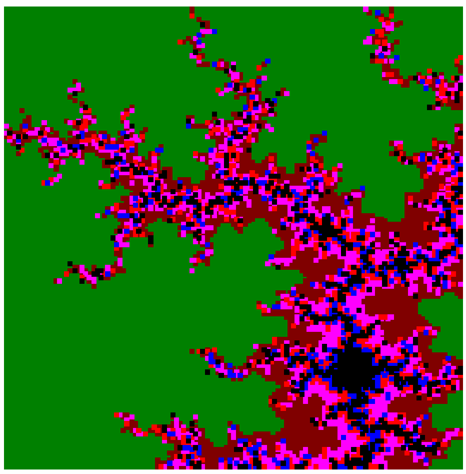


```
#18:  Mprog(-1.13, 0.3, -1.1, 0.28, 0.0005, levs5)
```



```
#19:  Mprog(-1.127, 0.287, -1.121, 0.28, 0.00005, levs5)
```

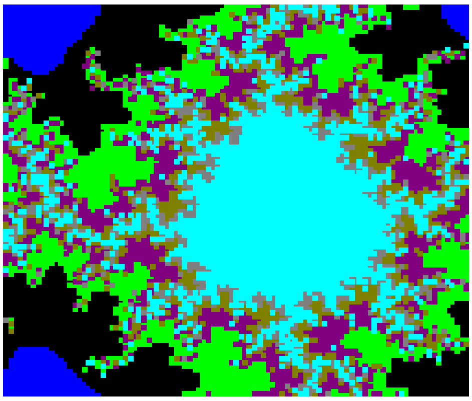


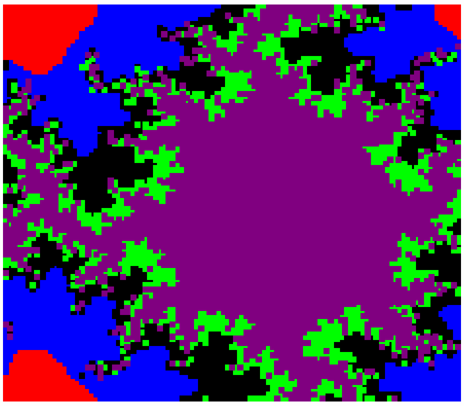


Same detail plotted using different colour palettes.

Performing the last zoom

#20: Mprog(-1.123, 0.282, -1.122, 0.281, 0.00001, levs5)

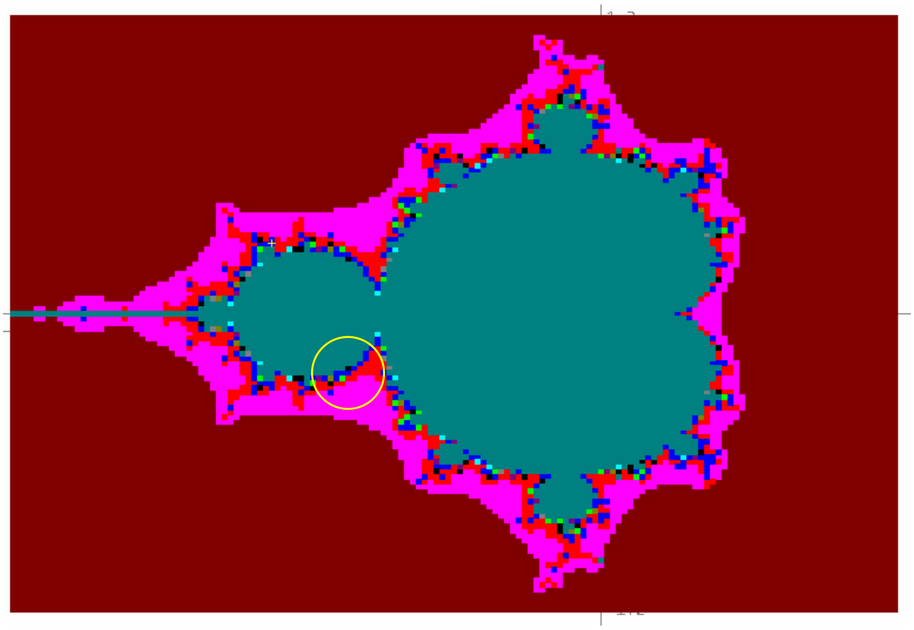




And tried this function for plotting the whole "Appel Man"…

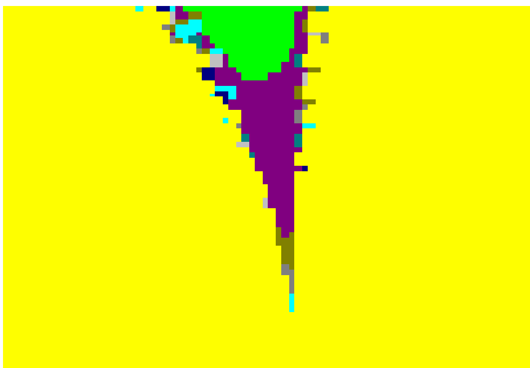
… followed by a short visit of the famous "Sea Horse Valley"

#21: Mprog(-2, 1.2, 1, -1.2, 0.02, levs5)

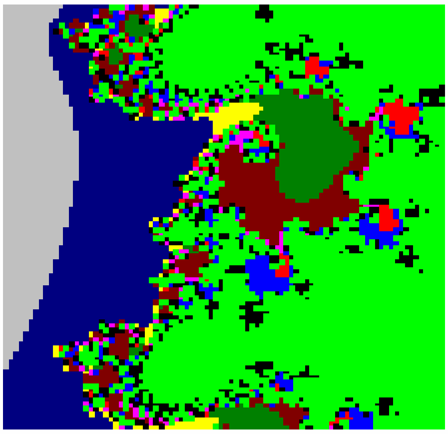


#22: Mprog(-0.8578, 0.1692, -0.6623, 0.0225, 0.005, levs5)

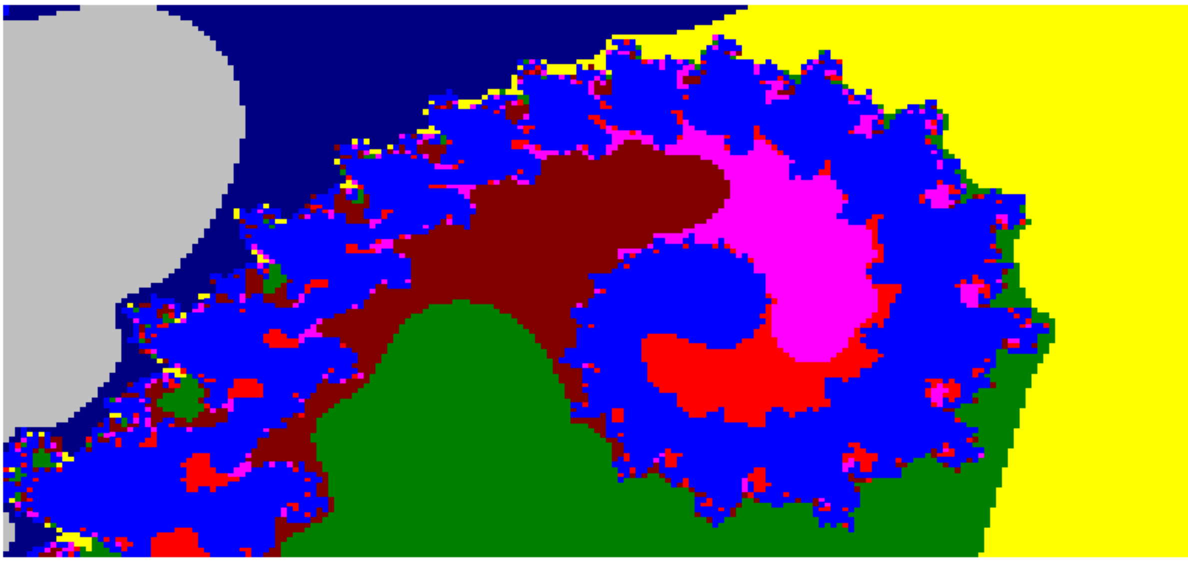
#23: Mprog(-0.8578, 0.1692, -0.6623, 0.0225, 0.0025, levs5)



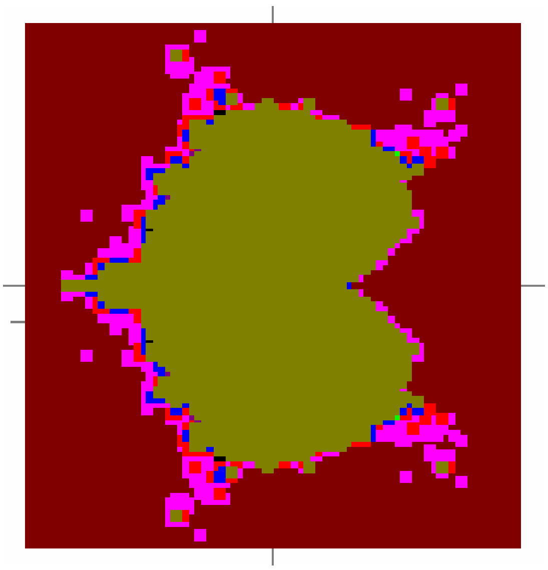
#24: Mprog(-0.75075, 0.10816, -0.742, 0.09898, 0.0001, [5, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 150])

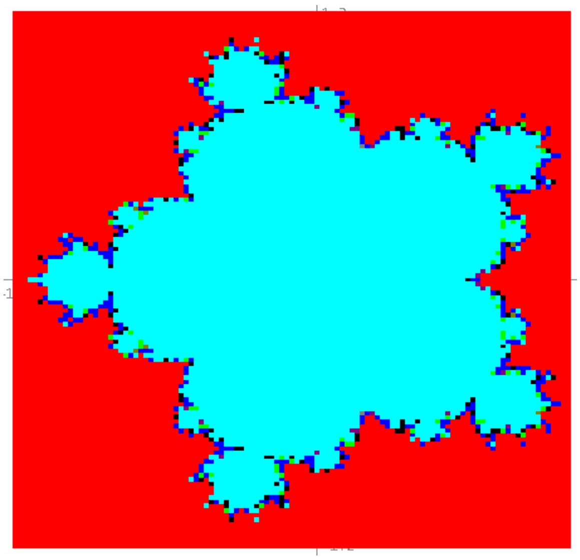


#24: Mprog(-0.747, 0.1057, -0.745, 0.1046, 0.00001, levs5)



Did you ever try changing the iteration z1:=z0^2 + a + b·*i* to z1:=z0^4 + a + b·*i* or z1:=z0^6 + a + b·*i* or z1:=z0^4 + z0^2 + a + b·*i* or …





See more spectacular Mandelbrot sets on page 50.

Comments on Klaus Körner's Contribution in DNL#83

David Halprin, Australia

Josef
Hello from downunder.

Even though I haven't written to you for a long time, my interest in DUG had never waned.

Your last DNL #83 grabbed my extra-special attention with the correspondence to and fro with Klaus Körner. My interest in the aerofoil profile started about 35 years ago when I came across a book by Richard von Mises in the university library. At the time there were no personal computers and I wondered how to plot such a curve. With the advent of the IBM PC locally in 1987 I still had trouble working out the right approach.

However, `the penny dropped' when I was working with curves, that were derived/transformed from other curves, such as:- Bertrand, Caustics (Dia-, Cata-), Caylean, Centroid (Centre of Gravity) Lines, Cissoids, Conchoids, Cycloids (Epi-, Hypo-, Prolate-, Curtate-), Evolute & Involute, Glissettes, Reflective Inverse, Isoptics, Orthoptics, Mannheim, Orthotomic, Orthogonal Trajectories, Pedal, Contra-Pedal, Paragraph (Parallel, Telegraph etc.), Radial, Pursuit, Negative Pursuit, Pseudo-Pursuit, and many other pseudo curves u.s.w..

Here was my initial interest in going backwards and forwards from Involute to Evolute and all the other such pairs of curves with their inverses. i.e. Base Curve to Derived Curve, then Derived Curve back to Base Curve.

Hence, the "AHA Moment" had arrived for the Joukowski Aerofoil profile, it being yet another derived/transformed curve, and in that particular case, the base curve is the nominated circle.

$$\text{Let } z_1 = x + iy \;\therefore\; \frac{1}{z_1} = \frac{x - iy}{x^2 + y^2}$$

$$\text{Let } z_2 = u + iv \;\therefore\; \frac{1}{z_2} = \frac{u - iv}{u^2 + v^2}$$

JOUKOWSKI PROFILE DEFINITION

$$x = u\left(1 + \frac{k}{u^2 + v^2}\right),\quad y = v\left(1 - \frac{k}{u^2 + v^2}\right),\quad z_1 = z_2 + \frac{k}{z_2}\quad \text{The Transform}$$

$$\therefore\; z_1^2 - z_1 z_2 + k = 0 \;\text{ giving }\; z_1 = ½.\left(z_2 + \sqrt{z_2^2 - 4k}\right)\quad \text{Inverse Transform}$$

$$(u - h)^2 + (v - h)^2 = h^2 + (h + 1)^2 \;\text{ reduces to }\; u^2 + v^2 = 2h(u + v + 1) + 1$$

$$\text{Let } h^2 + (h + 1)^2 = c^2,\quad u - h = c.\cos\theta,\quad v - h = c.\sin\theta$$

$$z_2 = u + iv = h(1 + i) + c.\text{cis}\theta = \sqrt{2}h.e^{\frac{i\pi}{4}} + c.e^{i\theta}\quad \text{The Circle}$$

$$z_1 = \sqrt{2}h.e^{\frac{i\pi}{4}} + c.e^{i\theta} + \frac{1}{\sqrt{2}h.e^{\frac{i\pi}{4}} + c.e^{i\theta}}\quad \text{The Joukowski Aerofoil Profile}$$

---

cis θ is just the common shorthand for Euler's rule cos θ + i.sin θ = $e^{i\theta}$.

This shows that the circle, which is being transformed, has as its centre (h,h) and that it passes through (-1,0).

In order to revert back to the circle, Z , whence the aerofoil profile arose, is the simple inverse transform as shown above.

If one chooses a base curve, and then performs a mathematical operation upon it, which transforms it into another curve, this latter curve may be referred to as a "derived curve" or a "transformed curve" and the operation is called a "mapping".

Curve A is operated upon to derive curve B by a process of transformation, summarised as a mapping from the Z plane to the W plane. A transforms to B. B is derived from A. A is mapped from the Z plane to the W plane.

There are two divisions of such mappings, which are called conformal and non-conformal. In the former case, the angles between two differentiable arcs are preserved.

In this paper, one finds a simple explanation of the basic Joukowski aerofoil profile, which lacks aerodynamic qualities due to its cusp, and a modification, which gives it theoretical practical possibilities in flight.

Also, one finds how to apply an inverse transformation from an aerofoil, back to a base curve, whence it arose.

$$Z_1 = Z_2 + \frac{k}{Z_2}$$

$$Z_1 + 2 = Z_2 + 2 + \frac{1}{Z_2} = \frac{(Z_2 + 1)^2}{Z_2}$$

$$Z_1 - 2 = Z_2 - 2 + \frac{1}{Z_2} + \frac{(Z_2 - 1)^2}{Z_2}$$

$$\frac{Z_1 - 2}{Z_1 + 2} = \left(\frac{Z_2 - 1}{Z_2 + 1} \right)^2$$

This is now modified, so as to eliminate the cusp, and is called the "Kármán-Trefftz transform".

$$\frac{Z_1 - n}{Z_1 + n} = \left(\frac{Z_2 - 1}{z_2 + 1} \right)^n$$

When n = 2 we have the regular Joukowski aerofoil profile, with the cusp being the angle between the tangents of the upper and lower surfaces (" = 0), the trailing edge angle.
In the modified version, we have:-

$$\alpha = \pi(2 - n) \quad \text{or} \quad n = 2 - \frac{\alpha}{\pi} \quad \text{where} \quad n \le 2$$

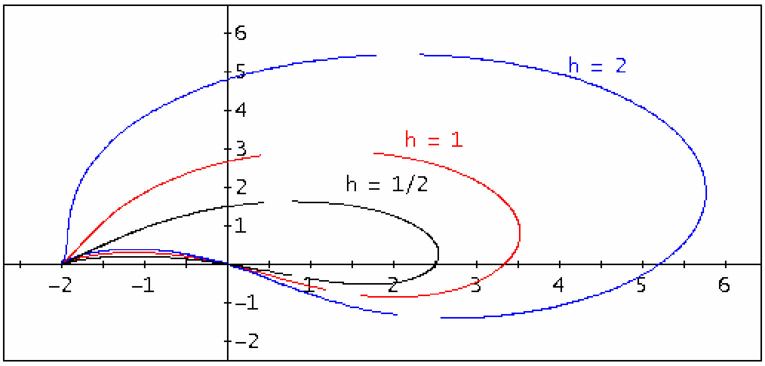
When N = 1.9 we see a good example of a more preferable profile for an actual wing.

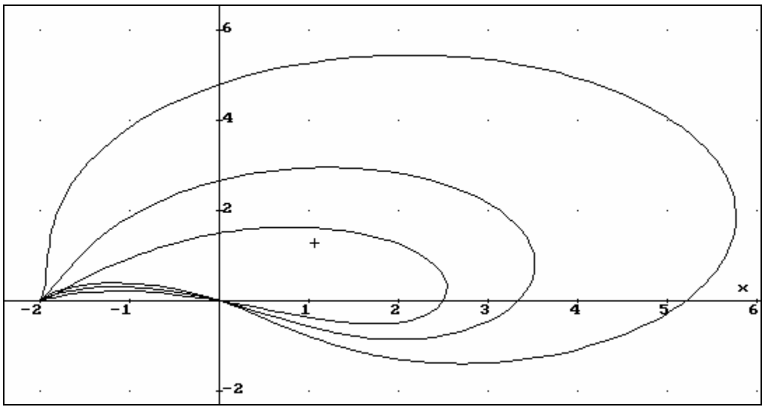
I shall attach a couple of .mth and .prt files from Derive XM, which I generated in 2003. Next week, I hope to implement the Z above in Derive For Windows 6.1 by running [RE(Z),IM(Z)]

N.B Variations on transliteration:-Joukowski = Joukowsky = Zhukovsky

Here are some graphs of the airofoils produced with JOUK.MTH and JOUKOW-1.MTH.

It was very interesting for me (Josef) that plotting with DERIVE XM was much easier than plotting with DERIVE 6.1. See the gap in the 6.1. aerofoils (first graph) and compare with the respective DERIVE XM-graph. Both were plotted with the same interval for the parameter in JOUK.MTH.

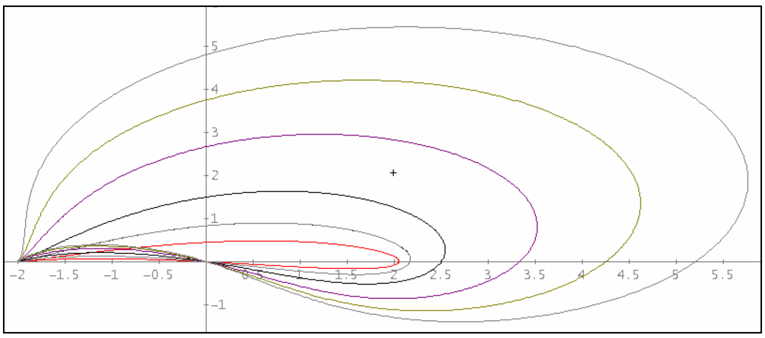




The next plots are results from JOUKOW-1.MTH.

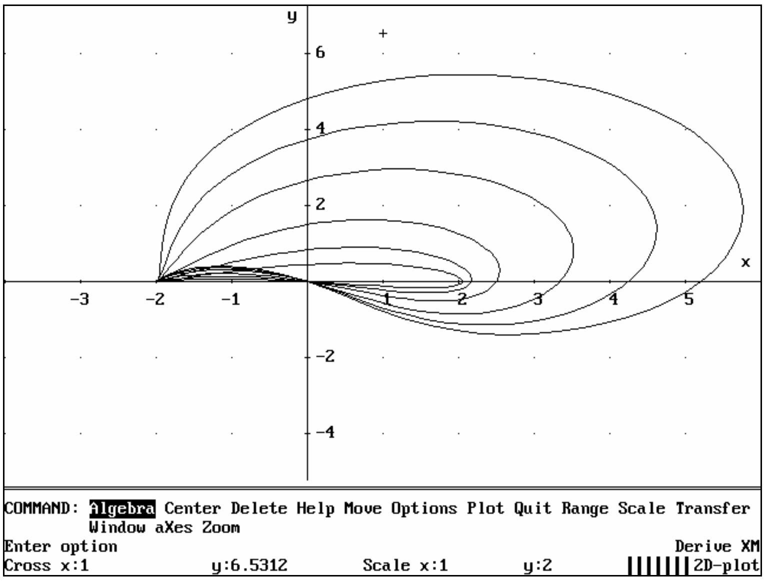
$$\left[\mathrm{RE}\left(\frac{2\cdot\left(\left(\dfrac{h\cdot(1+i)+e^{i\cdot t}\cdot\sqrt{(h+1)^2+h^2}-1}{h\cdot(1+i)+e^{i\cdot t}\cdot\sqrt{(h+1)^2+h^2}+1}\right)^2+1\right)}{1-\left(\dfrac{h\cdot(1+i)+e^{i\cdot t}\cdot\sqrt{(h+1)^2+h^2}-1}{h\cdot(1+i)+e^{i\cdot t}\cdot\sqrt{(h+1)^2+h^2}+1}\right)^2}\right),\ \mathrm{IM}\left(\frac{2\cdot\left(\left(\dfrac{h\cdot(1+i)+e^{i\cdot t}\cdot\sqrt{(h+1)^2+h^2}-1}{h\cdot(1+i)+e^{i\cdot t}\cdot\sqrt{(h+1)^2+h^2}+1}\right)^2+1\right)}{1-\left(\dfrac{h\cdot(1+i)+e^{i\cdot t}\cdot\sqrt{(h+1)^2+h^2}-1}{h\cdot(1+i)+e^{i\cdot t}\cdot\sqrt{(h+1)^2+h^2}+1}\right)^2}\right)\right]$$

h = 1/8, 1/4, 1/2, 1, 3/2 and 2. Plotting with DERIVE 6.1 needs a lot of calculation time (first figure)



Plotting with DERIVE XM is a work of an instant!

I don't know what makes the difference?



P.S.---------------------28 October 2011

I wrote this letter/paper yesterday afternoon, and before I went to bed I discovered an online eBook website that allowed me to purchase the 1959 Reprint of 'The Theory of Flight', so I shall be able to relive the experience after 35 years and see what von Mises' mathematics does.

I shall send you a copy later.

This is the link where David bought the copy of Mises' Theory of flight:

http://isbnbookfinder.com/Theory-of-Flight/p179525/

Regards
David Halprin

These are some links which I found wrt David's paper:

http://math.fullerton.edu/mathews/c2003/JoukowskiTransBib/Links/JoukowskiTransBib_lnk_3.html

http://de.wikipedia.org/wiki/Konforme_Abbildung

http://www.vaxman.de/analog_computing/joukowski/joukowski.html

http://www.luftpiraten.de/glos_j00.html

David sent another paper which I will present in the next DNL.

Many thanks for your persisting interest in cooperating with the DUG, Josef

Some requests concerning the TIs:

From Hannes Scherzer, Austria

I am owner of TI-Nspire™ CAS Version 3.0.1.1753. My question is:

How to shade an area between two curves. The TI-83 PLUS offers the function shade(….).

In DERIVE I can enter: 0 ≤ y ≤ f(x) AND (x ≥ 0 AND x ≤ 4).

Is there a possibility to do this with Nsipre, too?

DNL:

I don't know how to do this. Is there any Nspire user who can help?

From Nils Hahnfeld, Virgin Islands

(1) How can I convince the TI89 that (-1)^n ∗ (-1)^n = 1?
Why does it refuse simplifying this?

(2) Do you know a trick how to simplify (2n)!/n! or (n+3)!/n! ?

DNL:

DERIVE does not have any problems with the simplifications – but the first calculation needs specification of n as an integer.

I believe that the "not-simplifying" is connected with the internal treatment as complex numbers.

#1: $(-1)^n \cdot (-1)^n = (-1)^{2 \cdot n}$

#2: n :∈ Integer

#3: $(-1)^n \cdot (-1)^n = 1$

#4: $\frac{2 \cdot n!}{n!} = 2$

#5: $\frac{(n + 3)!}{n!} = (n + 1) \cdot (n + 2) \cdot (n + 3)$

An interesting mathematical request:

Hi;
My objective is to discover the exact 3D points for creating Platonic solids. Since these are the only geometries that have equal length sides and equal angles between them, I thought it might be easiest to find a formula that enables me to calculate farthest points within a sphere where the lengths and angles are equal. Does such a formula exist?
Jonathan Pike [jonathanpike55@yahoo.com]

Hi;
The Platonic solids nest perfectly one within another. This configuration, starting with the octahedron, is known as Metranon's Cube. I would like to build a mathematically perfect model of the same. Andrew Robbins was kind enough to point me to this page:

http://mathworld.wolfram.com/PlatonicSolid.html

which gives me a lot of pertinent data; however, the data for exactly nesting one solid in another seems lacking by half. It gives the formulas for determining the spherical radii of nested solids vis-a-vis their "dual" solids; however, those duals are matched icosahedron-dodecahedron, octahedron-cube and tetrahedron-tetrahedron (for a star tetrahedron). In the Metranon's Cube as described above, the tetrahedron is between the octahedron and the cube, so how does one figure out the "duals" or the radii of nested solids? This problem compounds itself with the matter of bridging the cube to the icosahedron. How do I resolve these matters?
Jonathan

A TI-Nspire problem (conflict?) with radian and degree

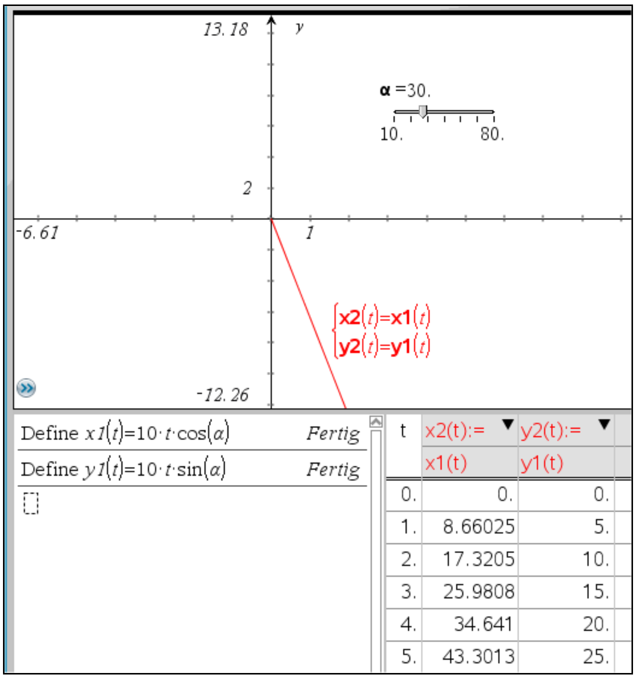
I defined a straight throw in parameter form dependent on the angle. I set the degree mode in the general settings. Surprisingly I obtain a graph showing a negative slope (with an angle of 30°) in the geometry window. The function table is correct – fitting to the degree settings. Working in rad mode the graph appears correctly.

My conjecture is: the graph is based on the radiant although the equation gives correct function values which are given in the table.

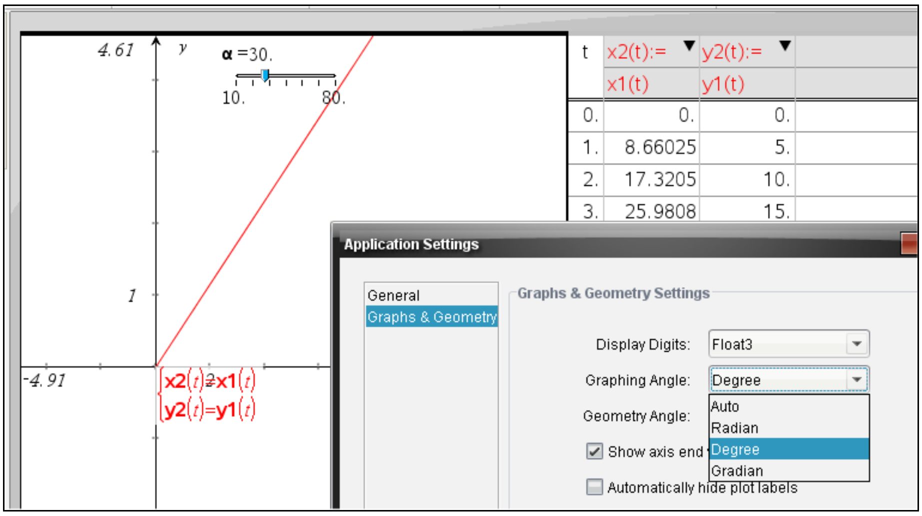
My question is: Is my conjecture right or am I doing something wrong?

Many thanks and best regards

Helmut



The "trick" is: You have to set the Geometry Angle to Degree in the Application settings for the Graphs & Geometry Application,too. It is not sufficient doing this in the General Settings. Don't ask me why one has to do this twice. Josef



Robert Setif's "autistic problem"

```
Dear Josef,
My function naff3 is not correct. I can't master the command append(out,res
sub [ ,...,] nor append(out,[res sub [ ,..., ]. I would obtain 123 456 or
[123],[456] or [[123],(456]] for 123456;

I would like to have 1 234 567 890 for 1234567890  or with the minimum of
[] 12 345 678 901 for 12345678901.

Best regards.
Robert
```

```
naff3(1234567891, 3) = [[[1, 234, 567, 891]]]
```

```
Von: Robert SETIF [robert.setif@gmail.com]
Gesendet: Dienstag, 17. Mai 2011 12:41
Betreff: naffndigs !!!

Dear Josef,
Excuse me with my "autistic" problems.
I would like a function that displays only the fractional part .(or the
integer part and the rest below).
For 3.14... -> 14159 26535 89793 ... or 141 592 653 ...
This is not urgent.
Thank you very much.
Best regards.
Robert
```

Dear Robert,

in earlier times a used to program a lot using string operations.
I attach my version of diffn.mth.

The trick is to insert a blank (= " ") between the groups of characters.

Best regards
Josef

```
      naffndigs(n, digs, res, out) :=
        Prog
          n := STRING(n)
          out := ""
          Loop
#1:         If DIM(n) ≤ digs
               Prog
                 out := APPEND(n, out)
                 RETURN out
            out := APPEND(" ", n↓[DIM(n) - digs + 1, ..., DIM(n)], out)
            n := n↓[1, ..., DIM(n) - digs]

#2:   naffndigs(1234567890, 4) = 12 3456 7890

#3:   naffndigs(1234567890, 3) = 1 234 567 890

#4:   naffndigs(1234567890, 4) = 12 3456 7890

#5:   naffndigs(1234567890, 6) = 1234 567890

#6:   naffndigs(1234567890, 5) = 12345 67890

#7:   naffndigs(12345678901234567890, 3) = 12 345 678 901 234 567 890

#8:   naffndigs(12345678901234567890, 2) = 12 34 56 78 90 12 34 56 78 90

#9:   naffndigs(12345678901234567890, 1) = 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0
```

Rick Nungester wrote in the eDUG-Forum:

I am using Derive 5.06 for Windows, looking at exponentiation with negative base and/or power. "#i" is "i-hat" or SQRT(-1). Here are Derive results:

-0.62^3.4

Simplify: -29791*38440^(1/5)/1250000

Approximate: -0.1968483343

Same base value, but complex representation: (-0.62 + 0*#i)^3.4

Simplify:
-29791*4617605000000^(1/10)/5000000+29791*38440^(1/5)/5000000 +/-
#i*(55935854318121351468726663312657665048706155177731 31*SQR(5)/4768371582031250000 0000000000000000000000000000000000000+508507766528375922442969666478706045897328683433921/190734863281250000000000000000000000000000000000000)^(1/10)

Approximate: -0.06082948062 +/- 0.1872138910*#i

*** Why aren't the two answers the same?

*** Are both answers correct?

I thought complex exponentiation z^u is defined as e^(u*LN(z)), or as an algorithm, find LN(z) (a function with principal branch as part of its definition and undefined at z = 0), then find u*LN(z) (a function), then find e^(u*LN(z)) (a function).

I also thought x^y is real if (x > 0) or (x = 0 and y > 0) or (x < 0 and y is integer), with 0^0 depending on who is doing the interpretation. For example, 0^0 = 1 in Derive but = "undefined" on the HP-32SII calculator.

*** Where else am I confused?

The above is a typing error. It should be: (-0.62)^3.4

I got this reply via email:

…here (-0.62^3.4) approximates to -0.06082948062 - 0.1872138910·î(same version of DERIVE). The absolute value of the above approximates to 0.1968483343, in accordance with your post.

My guess is that some non-standard settings (about branches and cuts in the complex plane) are in effect on your system. Try starting up DERIVE with the default (factory) settings and repeat the calculation.

Dirk

Thanks Dirk, that was it. I had Derive set to not prompt me if I wanted to start with factory settings, and have been accumulating non-factory changes for years. I don't know how I got my posted results, but now don't care. From now on I plan to always start with factory defaults and only deviate from them intentionally, saving those commands as part of my saved .dfw files.

A Mathematical Model for Snail Shells (3)

Piotr Trebisz, Germany

In meinen ersten zwei Beiträgen über die Trebisz-Spiralen vom Typ 1 und 2 bin ich ausführlich auf die Konstruktion von Schneckenhäusern eingegangen die auf planaren bzw. konischen logarithmischen Spiralen basieren. Man könnte also meinen dass damit schon alles über beide Schneckenhaus-Typen gesagt wurde.

Beiden Schneckenhaus-Typen ist gemein, dass wir das lokale mitdrehende Koordinatensystem mit Hilfe der frenetschen Formeln hergeleitet haben. In der Differentialgeometrie ist das die Standardmethode für die Bestimmung eines begleitenden Dreibeins einer Raumkurve.

Ich werde aber in diesem und im nächsten Beitrag zeigen dass es zu jeder planaren und konischen Spirale ein korrespondierendes orthogonales krummliniges Koordinatensystem gibt, das sich auf ganz natürliche Weise aus der Spiralen-Gleichung ergibt und in dem das Selbstdurchdringungsproblem auf einfache Weise analytisch lösbar ist!

In diesem Beitrag werde ich mich auf planare logarithmische Spiralen konzentrieren. Die daraus abgeleiteten Schneckenhäuser nenne ich "Trebisz-Spiralen - Typ 3".

In krummlinigen Koordinatensystemen sind die Koordinatenachsen keine Geraden sondern Raumkurven. Die wohl bekanntesten sind das sphärische und das zylindrische Koordinatensystem.

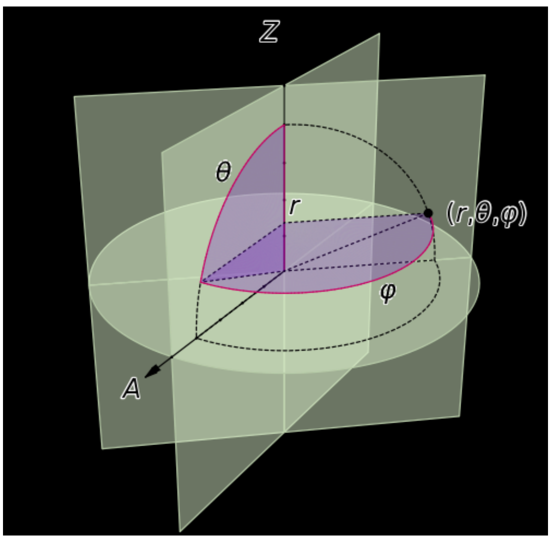
In my first contributions I treated snail shells based on planar and conical logarithmic spirals. Both types have in common that the local co-rotating system of coordinates was derived using the Frenet formulae (which is the standard method in differential geometry).

In this contribution and in the next one as well I will demonstrate that there exists a corresponding curvilinear coordinate system to each planar and conical spiral, which can be easily derived from the spiral equation. Then the interpenetrating problem can be solved analytically.

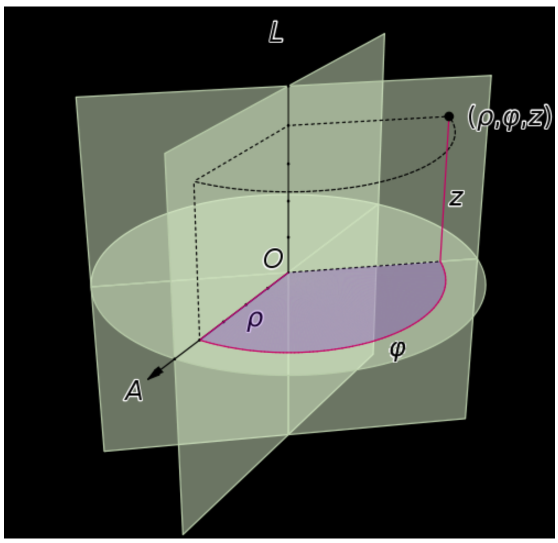
Here I will concentrate on planar logarithmic spirals. The resulting snail shells will be called "Trebisz-Spirals – Type 3".

In a curvilinear coordinate system the coordinate axes are no straight lines but space curves. The best known of these systems are the spherical and the cylindrical coordinate system.

Sphärisches Koordinatensystem:



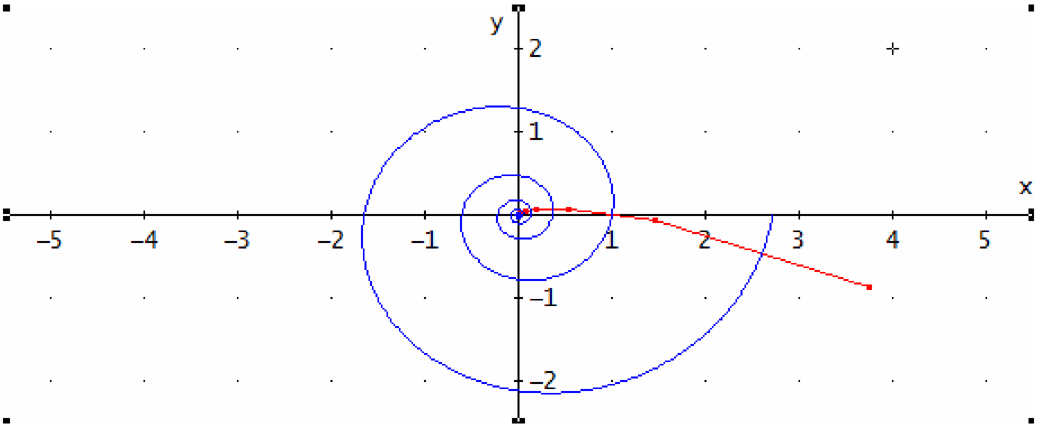
Zylindrisches Koordinatensystem:



Worin liegt nun also die Motivation dafür, ein krummliniges Koordinatensystem zu verwenden? Dazu werfen wir am besten einen Blick auf den Längsschnitt durch ein Schneckenhaus vom Typ 1, das nach der Windungszahl parametrisiert ist. Die Spiralkurve, die als Basis für das Schneckenhaus dient, ist hier in blau dargestellt. Die roten Linien sind Projektionen der sich berührenden Mantelkreise in der YZ-Ebene.

What is the motivation for using a curvilinear coordinate system? Let's have a look on the axial section of a type 1 snail shell, which is parameterized by the winding number. The spiral which serves as base for the shell is presented blue. The red segments are the projections of the osculating circles of the surface into the yz-plane.

Beispiel: b = *e* und 0 ≤ t ≤ *e*

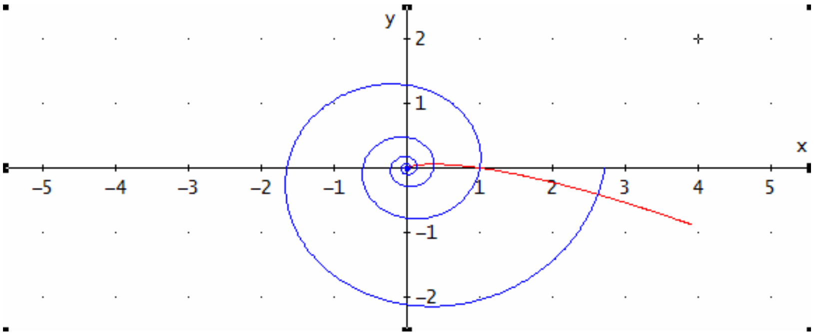


Es zeigt sich deutlich, dass die Mantelkreise einen Kantenzug ergeben der wiederum eine logarithmische Antispirale approximiert! Das erkennt man daran, dass der Winkel und das Längenverhältnis zwischen zwei benachbarten Kanten konstant sind. Die beiden Spiralen schneiden sich in den Schnittpunkten immer unter dem Winkel von 90°. Das ist auch nicht weiter überraschend, denn die Mantelkreise stehen ja definitionsgemäß immer senkrecht auf den Tangenten der Spiralkurve.

Ein Schneckenhaus vom Typ 3 entspricht im Wesentlichen einem Schneckenhaus vom Typ 1. Als Gerüst dient dieselbe planare logarithmische Spirale und der Radius des Mantelkreises wächst auch linear mit dem Radius bzw. mit der Länge der Spirale. Der Unterschied besteht nur darin, dass der Mantelkreis in einem krummlinigen lokalen Koordinatensystem definiert ist und nicht in einem kartesischen begleitenden Dreibein wie beim Schneckenhaus vom Typ 1. Aus dem Kantenzug im Längsschnitt wird daher eine perfekte, stetige Antispirale.

We can see that the segments approximate a logarithmic antispiral. This can be recognized by the fact that the angles and the ratios of consecutive segments is constant. Both spirals intersect with 90°. This is not really surprising because the surface circles are – by definition – always perpendicular to the tangents of the spiral.
A type 3 snail shell basically corresponds with a type 1 shell. The same planar logarithmic spiral serves as scaffold and the radius of the surface increases linearly with the radius and the length of the spiral respectively. The difference is that the surface circle is now defined in a curvilinear local coordinate system and not in a Cartesian three pod (type 1). The segment progression changes to a perfect continuous antispiral.

Beispiel: b = *e* und 0 ≤ t ≤ *e*



Die Verwendung einer solchen Antispirale als lokale Y-Achse bietet aber nicht nur ästhetische, sondern auch ganz handfeste Vorteile. Die Berührungspunkte der Mantelkreise liegen hier alle auf einer stetigen Kurve, dadurch wird das Selbstdurchdringungsproblem analytisch lösbar!

Ausgangspunkt der Herleitung ist die eigentliche Spirale, die als Basis für das Schneckenhaus dient, und ihre Tangentengleichung. Der Parameter *t* steht hierbei für den Radius bzw. die Länge der Spirale, während *b* die Basis des Logarithmus ist.

Using such an „antispiral“ as local y-axis does not offer only aesthetic benefits but also concrete ones. All the osculation points are lying on a continuous curve which makes the interpenetrating problem solvable.

Starting point of the derivation is the spiral which serves as base for the shell and its tangent equation. Parameter t *describes the radius and the length of the spiral, respectively, and* b *is the base of the logarithm.*

```
#3:   SPIRALE(b, t) := [0, t·COS(2·π·LOG(t, b)), t·SIN(2·π·LOG(t, b))]

                                 d
#4:   SPIRALE_TANGENTE(b, t) :=  —— SPIRALE(b, t)
                                 dt
```

Gesucht ist eine logarithmische Antispirale, die die Basisspirale immer senkrecht schneidet.

We are looking for an antispiral which intersects the base spiral always orthogonally.

```
#5:   ANTI_SPIRALE(a, s) := [0, s·COS(2·π·LOG(s, a)), s·SIN(2·π·LOG(s, a))]

                                      d
#6:   ANTI_SPIRALE_TANGENTE(a, s) :=  —— ANTI_SPIRALE(a, s)
                                      ds
```

Es erweist sich hier als sehr hilfreich, dass planare logarithmische Spiralen bei $t = 1$ bzw. $s = 1$ immer durch den Punkt [0, 1, 0] gehen, unabhängig von ihrer Basis.

It proves to be helpful that planar logarithmic spirals for t = *1 and* s = *1 are passing the point [0, 1, 0] independent on their base.*

```
#7:   SPIRALE(b, 1) = ANTI_SPIRALE(a, 1) = [0, 1, 0]
```

Gesucht ist eine Basis *a* für die Antispirale, so dass die beiden Tangenten in den Schnittpunkten immer senkrecht zu einander stehen.

We try to find a base a *for the antispiral such that both tangents in the intersection points are perpendicular.*

#8: $\text{BASIS(b)} := (\text{SOLUTIONS}(\text{SPIRALE\_TANGENTE}(b, 1) \cdot \text{ANTI\_SPIRALE\_TANGENTE}(a, 1) = 0, a, \text{Real}))_1$

#9: $\text{BASIS(b)} = e^{-4\cdot\pi^2/\text{LN}(b)}$

Die Länge der Antispirale hängt linear vom Parameter *s* ab, da ihre Tangente an jedem Punkt eine konstante Länge aufweist.

The length of the antispiral depends linearly on the parameter „s because its tangent is of constant length in every of its points.

#10: $|\text{ANTI\_SPIRALE\_TANGENTE}(\text{BASIS}(b), s)| = \dfrac{\sqrt{(\text{LN}(b)^2 + 4\cdot\pi^2)}}{2\cdot\pi}$

Da die Anti-Spirale also nach ihrer Länge parametrisiert ist, aber multipliziert um einen konstanten Maßstabsfaktor, ist sie als Y-Achse in einem krummlinigen Koordinatensystem gut geeignet.

Als X-Achse benötigen wir einen Vektor, der senkrecht zu der Ebenen steht in der sich die Basisspirale windet. Auf diese Weise stellen wir sicher, dass die lokale X-Achse immer senkrecht auf die Tangenten der Antispirale und der Basisspirale steht.

Die Gleichungen für die Achsen des krummlinigen Koordinatensystems lauten daher:

As the antispiral is parameterized wrt to its length but multiplied by a constant scaling factor, it is well suitable for serving as y-axis in a curvilinear coordinate system.

As x-axis we need a vector perpendicular to the plane which contains the base spiral. So we can make sure that the local x-axis is always perpendicular to the tangents of base spiral and antispiral as well.

The equations for the axes of the curvilinear coordinate system are:

```
#11:  X_ACHSE(b, x) := |ANTI_SPIRALE_TANGENTE(BASIS(b), s)|·[x, 0, 0]

#12:  Y_ACHSE(b, y) := ANTI_SPIRALE(BASIS(b), y)
```

Um zu gewährleisten dass der Maßstab in Richtung der X-Achse genauso groß ist wie in Richtung der Y-Achse, wird die X-Achse mit dem konstanten Maßstabsfaktor der Y-Achse multipliziert.

Zudem ist dieses krummlinige Koordinatensystem orthogonal, das heißt, dass die partiellen Ableitungsvektoren immer senkrecht zu einander stehen.

To make sure that the scaling in direction of the x-axis is equal to the scaling in direction of the y-axis the x-axis is multiplied by the constant scaling factor of the y-axis.

Additionally this curvilinear coordinate system is orthogonal, i.e. the vectors of the partial derivatives are perpendicular.

#13: $\left(\dfrac{d}{dx}(\text{X\_ACHSE}(b, x) + \text{Y\_ACHSE}(b, y))\right) \cdot \dfrac{d}{dy}(\text{X\_ACHSE}(b, x) + \text{Y\_ACHSE}(b, y)) = 0$

Der Mantelkreis wird im krummlinigen Koordinatensystem definiert. Sein Radius soll linear mit dem Radius bzw. mit der Länge der Basis-Spirale wachsen, *m* ist dabei der Linearfaktor.

The surface circle is defined in the curvilinear coordinate system. Its radius shall grow linear with the radius and the length of the base spiral, respectively, with m *as linear factor.*

```
#14:  KREIS(b, m, φ) := X_ACHSE(b, m·COS(φ)) + Y_ACHSE(b, 1 + m·SIN(φ))
```

Damit haben wir die Gleichung des Mantelkreises für $t = 1$ dessen Zentrum im Punkt [0, 1, 0] liegt. Das lässt sich zeigen indem man den Radius auf 0 reduziert.

The result is the equation of the surface circle for t = 1 with its centre in point [0, 1, 0]. This can be shown by reducing the radius to 0.

```
#15:  KREIS(b, 0, φ) = [0, 1, 0]
```

Um daraus die Mantelfläche des Schneckenhauses zu konstruieren, muss man den Mantelkreis um den Winkel der Basisspirale rotieren und mit ihrem Radius skalieren.

For constructing the surface of the snail shell one has to rotate the surface circle by the angle of the base spiral and scale it by its radius.

```
#16:  MANTEL(b, m, t, φ) := t·ROTATE_X(2·π·LOG(t, b)) • KREIS(b, m, φ)

#17:  MANTEL(b, 0, t, φ) = SPIRALE(b, t)
```

Die Berührungspunkte der Mantelkreise liegen nach jeder Windung auf der Antispiralen. Dieser Umstand ermöglicht es uns, das Selbstdurchdringungsproblem analytisch zu lösen!

Wir wissen, dass sich die Zentren der Mantelkreise nach jeder Windung in den Schnittpunkten der Basisspirale mit der Antispirale befinden.

Es müssen also zuerst die Schnittpunkte der Spiralen berechnet werden.

Sowohl die Basisspirale als auch die Antispirale sind nach ihrem Radius parametrisiert.

The osculation points of the surface circles are located on the antispirals after every turn. This fact makes possible solving the interpenetration problem analytically!

We know that the centres of the surface circles after every turn are located in the intersection points of the base spiral and the antispiral.

So we have to calculate the intersection points of the spirals first.

The base spiral and the antispiral are both parameterized wrt to their radius.

```
#18:  |SPIRALE(b, t)| = t ∧ |ANTI_SPIRALE(BASIS(b), s)| = s
```

Wir können deshalb für beide Spiralen denselben Parameter t verwenden, was uns die Berechnung der Schnittpunkte sehr vereinfacht.

So we can use the same parameter t *for both spirals which makes calculation of the intersection points much easier.*

```
         SOLUTIONS((SPIRALE(b, t))  = (ANTI_SPIRALE(BASIS(b), t))  ∧
#19:                              2                               2

                    SOLVE((SPIRALE(b, t))  = (ANTI_SPIRALE(BASIS(b), t)) , t, Real), t, Real)
                                         3                              3
```

$$\#20: \left[0, 1, b^{-4\cdot\pi^2/(LN(b)^2 + 4\cdot\pi^2)}, b^{4\cdot\pi^2/(LN(b)^2 + 4\cdot\pi^2)} \right]$$

Die Basisspirale und die Antispirale haben aber nicht nur vier, sondern unendlich viele Schnittpunkte. Anhand der vier speziellen Lösungen, die uns DERIVE liefert, können wir eine allgemeine Lösung für die Schnittpunkte angeben. k ist hier eine beliebige ganze Zahl.

The base spiral and the antispiral don't have only four intersection points but infinitely many. Inspecting the four special solutions delivered by DERIVE we can indicate a general solution for the intersection points. k *is an arbitrary integer.*

```
#21:                        2          2       2
                     4·π ·k/(LN(b)  + 4·π )
      SCHNITT(b, k) := b
```

Damit können wir den Linearfaktor *m* so bestimmen, dass sich die Mantelfläche nach jeder Windung exakt berührt ohne sich selbst zu durchdringen.

Hence we can define the linear factor m *in such a way that the surface is osculating itself after each turn without interpenetrating.*

```
      M_YZ(b) := (SOLUTIONS(SCHNITT(b, k)·(1 + m) = SCHNITT(b, k + 1)·(1 - m), m, Real))
#22:                                                                                   1

                    2         2      2
                4·π /(LN(b)  + 4·π )
               b                      - 1
#23:  M_YZ(b) = ─────────────────────────
                    2         2      2
                4·π /(LN(b)  + 4·π )
               b                      + 1
```

Aus der Differenz zweier aufeinander folgender Schnittpunkte können wir außerdem direkt ablesen, wie groß eine Windung beim Schneckenhaus vom Typ 3 ist, also die Umdrehungszahl nach der sich die Mantelfläche selbst berührt.

The difference of two consecutive intersection points enables reading off how large one turn of a type 3 snail shell is, i.e. the winding number after which the surface is osculating itself.

```
#24:  WINDUNG(b) := LOG(SCHNITT(b, k + 1), b) - LOG(SCHNITT(b, k), b)

                             2
                        4·π
#25:  WINDUNG(b) = ─────────────
                          2      2
                    LN(b)  + 4·π
```

Und damit sind wir fertig! Man muss nur das passende *m* in die Mantelfläche einsetzen und erhält so ein Schneckenhaus vom Typ 3, das nach dem Radius bzw. nach der Länge parametrisiert ist.

Now it is done, we are ready! We have to substitute the right m *into the surface expression in order to obtain a type 3 snail shell which is parameterized wrt to its length.*

```
#26:  SchneckenHausYZ_R(b, t, φ) := MANTEL(b, M_YZ(b), t, φ)
```

Will man das Schneckenhaus nicht nach dem Radius, sondern nach der Zahl der Umdrehungen θ parametrisieren, dann ersetzt man t durch b^θ. Der Radius wächst dann nicht mehr linear zu t sondern zu b^θ.

If one does not want parameterize the snail shell wrt the radius but wrt to the number of turns θ *then one has to replace* t *by* b^θ*. Then the radius will grow linear to* b^θ*.*

```
                       θ      θ
#28:  |SPIRALE(b, b )| = b

                                                         θ
#29:  SchneckenHausYZ_U(b, θ, φ) := MANTEL(b, M_YZ(b), b , φ)
```

Da wir wissen welcher Umdrehungszahl eine Windung entspricht, können wir auch ein Schneckenhaus definieren das nach der Anzahl der Windungen parametrisiert ist.

As we know which rotation number corresponds with one turn we can define a snail shell parameterized by the number of turns.

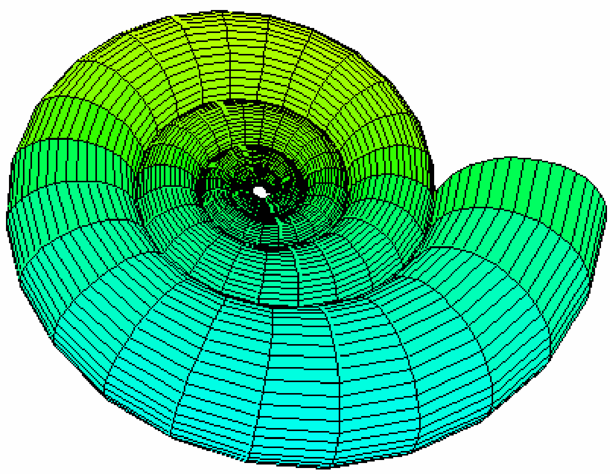
```
#31:   SchneckenHausYZ_W(b, θ, φ) := SchneckenHausYZ_U(b, θ·WINDUNG(b), φ)
```

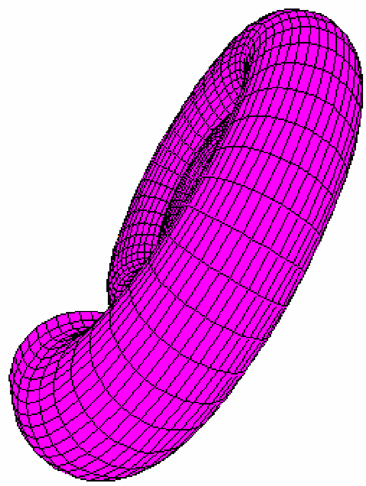
Und damit die ganze Rechnerei nicht umsonst war werden wir belohnt mit einigen schönen Schneckenhäusern.

Finally we are rewarded for all our calculations with some pretty snail shells.

<u>Basis 2:</u>

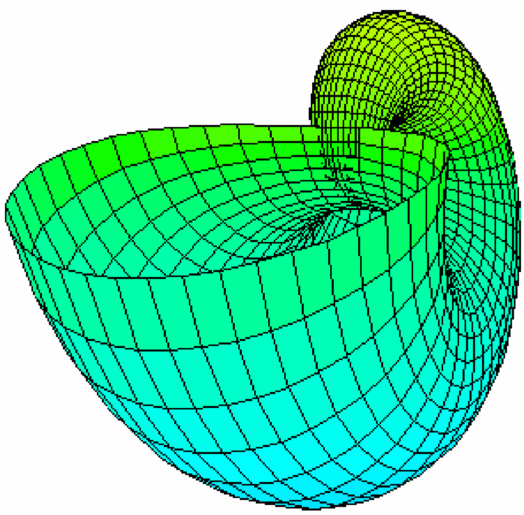
#40: MANTEL(2, M_YZ(2), 2^t, s) $(-\pi \le s\ \pi, -4 \le t \le 1)$





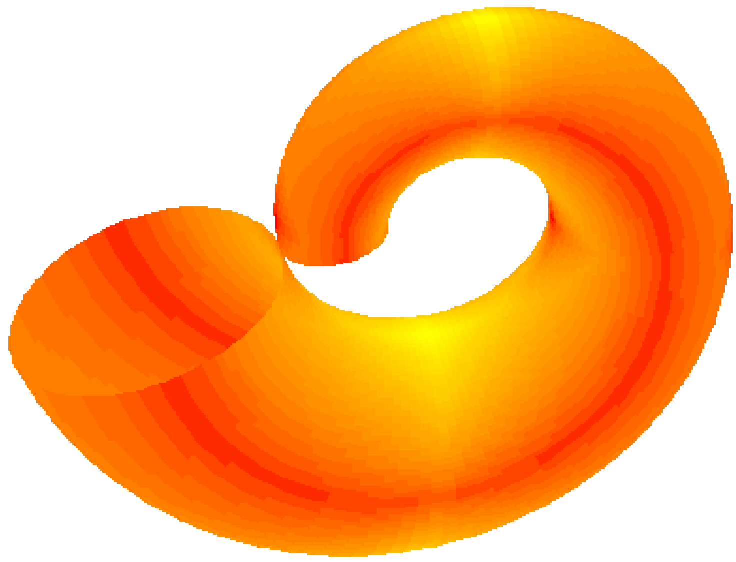
<u>Base 6:</u>

#41: MANTEL(6, M_YZ(6), 6^t, s) $(-\pi \le s\ \pi, -2 \le t \le 1)$

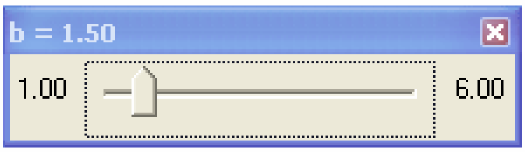


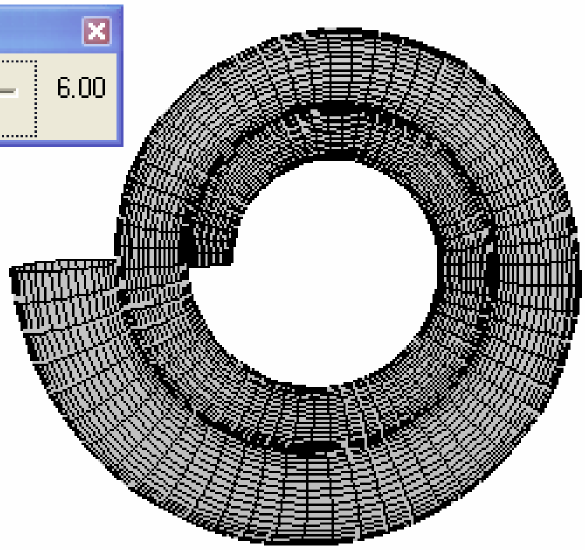
Base e with exact one turn:

#42: MANTEL(e, M_YZ(e), e^t, s)   ($-\pi \le s\ \pi$, $0 \le t \le 1$)



You can study the influence of *b* by introducing a slider bar for *b*, Josef.





This is another spiral.

Do you recognize which one? Josef

Solving Triangles has been always an issue for creating functions and programs using various software products. One can involve the students preparing the tools or provide the tools for a meaningful use. We had "TRIGO" in earlier DNLS (#23) and I presented a respective program in my NSpire-Programming book.

Erik van Lantschoot's tool is a tricky program working with lists and Boolean expressions for treating the different cases. It should be no problem to transfer this program to TI-92 and V 200 (and DERIVE?). Josef

TRIGO

Erik van Lantschoot, Germany

```
trigo(3,4,5,0,0,0)
        s= {3.0000,4.0000,5.0000}
        w= {36.8699,53.1301,90.0000}
                                        Fertig
trigo(0,0,0,30,60,90)
        s= {t,1.7321·t,2.0000·t}
        w= {30.0000,60.0000,90.0000}
                                        Fertig
trigo(2,3,0,10,0,0)
        s= {2.0000,3.0000,4.8854}
        w= {10.0000,15.0981,154.9019}
        The second solution is:
        s= {2.0000,3.0000,1.0235}
        w= {10.0000,164.9019,5.0981}
                                        Fertig
trigo(0,3,2,0,10,0)
        s= {4.9494,3.0000,2.0000}
        w= {163.3522,10.0000,6.6478}
                                        Fertig
trigo(0,3,2,0,0,50)
        Not a valid triangle!
                                        Fertig
trigo(0,150,0,45,0,105)
        s= {212.1320,150.0000,289.7777}
        w= {45.0000,30.0000,105.0000}
```

The German version is among the downloadable files in DNL84.zip.

```
trigo
Define trigo(s1,s2,s3,a1,a2,a3)=
Prgm
setMode({2,2,5,2,1,18})
© Program for solving triangles. You need exactly three
© data (sides or angles). Uses lists for the sides s:={s1,s2,s3}
© and similar for the angles w:={a1,a2,a3}.
© The program call is trigo(s1,s2,s3,)
© The sides and angles which are requested are indicated by 0.
© w[i] is the angle opposite to side  s[i].
© The four possible cases of the solution are indicated as TYPE (number of sides).
© ks (no side) and kw (no angle) are boolean list variables
© which indicate where there is no side  (true) or no angle.
© The variables starting with "id" are lists of the positions of data.
© e.g. s={0,23,44} then ks={true,false,false} and ids={2,3}
Local s,w,ids,idw,idkw,idks,idsw,idskw,idkskw,k,ks,kw,m,ss,r,typ
s:={s1,s2,s3}:w:={a1,a2,a3}
ks:=s={0,0,0}: kw:=w={0,0,0}
ids:=wstw(not ks): typ:=dim(ids): idw:=wstw(not kw): idks:=wstw(ks)
If dim(ids)+dim(idw)≠3 Then: Disp "Too many or too few data":Stop: EndIf
© STARTING HERE TYPE 0
If typ=0 Then: If sum(w)≠180 Then: Disp "Sum of angles ≠ 180°":Stop:EndIf
 s[1]:=t:  For k,2,3: s[k]:=sin(w[k])/sin(w[1])·t: EndFor
© STARTING HERE TYPE 1
ElseIf typ=1 Then: idkw:=wstw(kw): w[idkw[1]]:=180-sum(w):kw:=w={0,0,0}
idsw:=wstw(not ks and not kw): m:=s[idsw[1]]/sin(w[idsw[1]]):idks:=wstw(ks)
For k,1,2: s[idks[k]]:=m·sin(w[idks[k]]): EndFor
© STARTING HERE TYPE 2
ElseIf typ=2 Then
If dotP(s,w)=0 Then
© STARTING HERE TYPE 2.1
s[idw[1]]:=√(s[ids[1]]^2+s[ids[2]]^2-2·s[ids[1]]·s[ids[2]]·cos(w[idw[1]])) :Goto t3
Else
© STARTING HERE 2.2
idskw:=wstw(not ks and kw): idsw:=wstw(not ks and not kw)
m:=s[idsw[1]]/sin(w[idsw[1]]): w[idskw[1]]:=sin⁻¹(s[idskw[1]]/m)
idkskw:=wstw(ks and kw): w[idkskw[1]]:=180-sum(w)
If s[idskw[1]]/m>1 Then
Disp "Not a valid triangle!"
Goto aus
EndIf
s[idkskw[1]]:=m·sin(w[idkskw[1]])
Disp "s= ",s: Disp "w= ",w
© Is there another solution?
If w[idsw[1]]+180-w[idskw[1]]>180 Then: Goto aus:
Else: Disp "The second solution is:"
w[idskw[1]]:=180-w[idskw[1]]:w[idkskw[1]]:=0: w[idkskw[1]]:=180-sum(w)
s[idkskw[1]]:=m·sin(w[idkskw[1]])
EndIf:EndIf
© STARTING HERE TYPE 3
ElseIf typ=3 Then: If max(s)≥sum(s)-max(s) Then
Disp "Triangle inequality violated!":Stop: EndIf
Lbl t3 © coming from Typ 2.1
ss:=sum(s)/2: r:=√((ss-s[1])·(ss-s[2])·(ss-s[3])/ss)
For k,1,3: w[k]:=2·tan⁻¹(r/(ss-s[k])): EndFor
EndIf: Disp "s= ",s: Disp "w= ",w: Lbl aus
EndPrgm
```

There is an auxiliary function **wstw** hidden in the program:

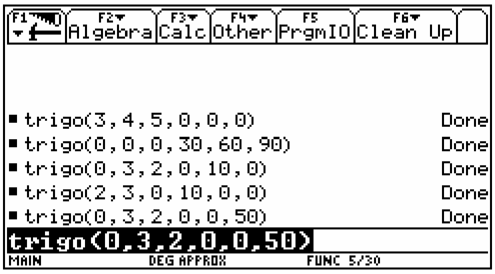
```
Define wstw(g)=
Func
:© wstw=where is what? g=list of boolean expression, e.g. where do I find
:© a side and no angle? then: g:={not ks and kw}. (ks = keine Seite = no side)
:© The indices of the positions of "true"in g are collected in list lst.
:Local h,lst: lst:={}
:For h,1,3: If g[h]=true Then: lst:=augment(lst,{h}): EndIf: EndFor
:Return lst
:EndFunc
```

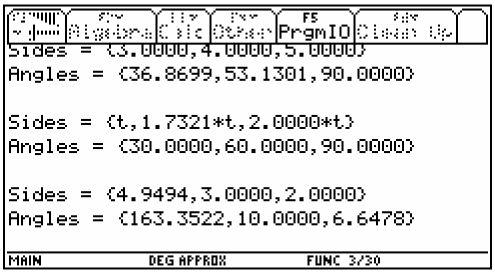
I wrote above that it should be no problem to transfer TRIGO to the other TI-calculators.

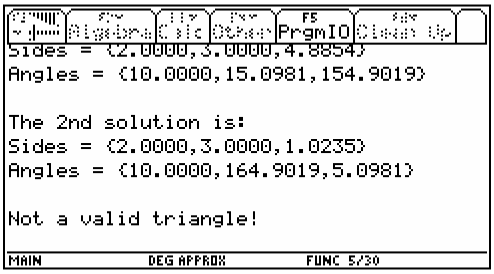
I tried for the Voyage 200. The most important difference is that assigning values needs a → on the Voyage (TI-92) instead of := with Nspire.

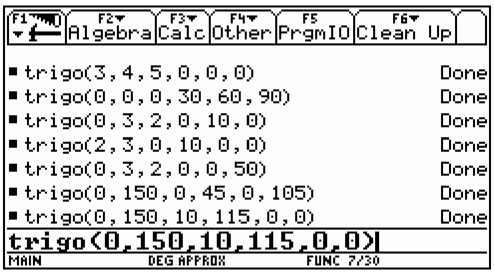
See some screen shots from the V200 treating the same problems as before with TI-Nspire CAS.

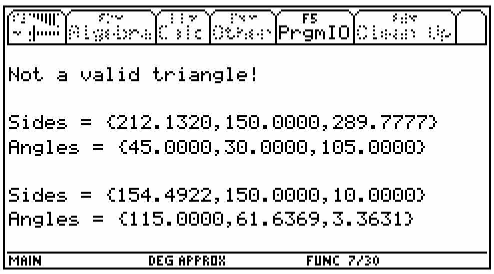
Josef

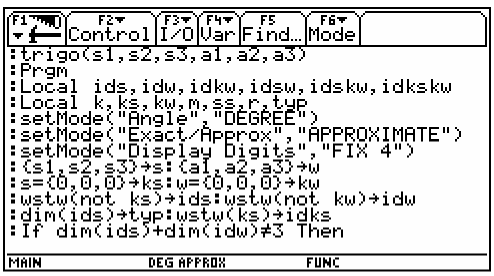






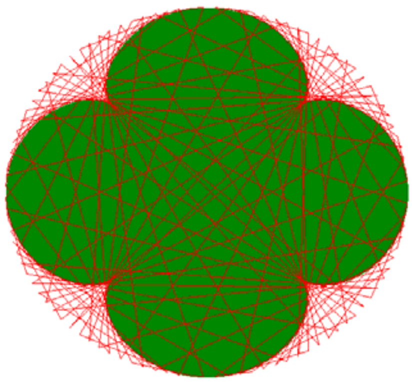


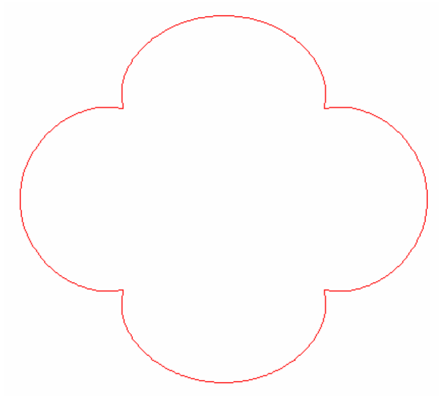




Roland Schröder's gift for the DUG Members:

Roland sent a DERIVE made Four-Leaf Clover as a Lucky Charm 2012 for all DUG Members.





$$\text{VECTOR}\left(\left[1, \frac{\text{MOD}(5^n, 103)\cdot 2\cdot\pi}{103}\right], n, 0, 103\right)$$

???

I wrote back:

I did an extra work looking for the envelope of the family of lines. It fits exactly.
Thanks for the challenge!

(I don't present the formula here and leave it for our readers. It will be presented in DNL#85, Josef)

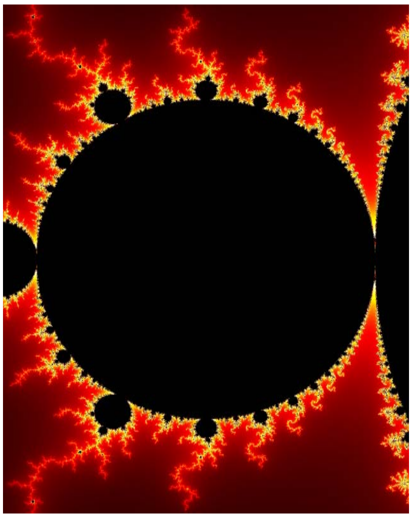
Roland answered (in German and I translate)
Dear Josef,
I find it really striking that your set of points describes the four-leave clove. As I had never to do geometry I don't have any clue why this is the case and how to find such a formula.

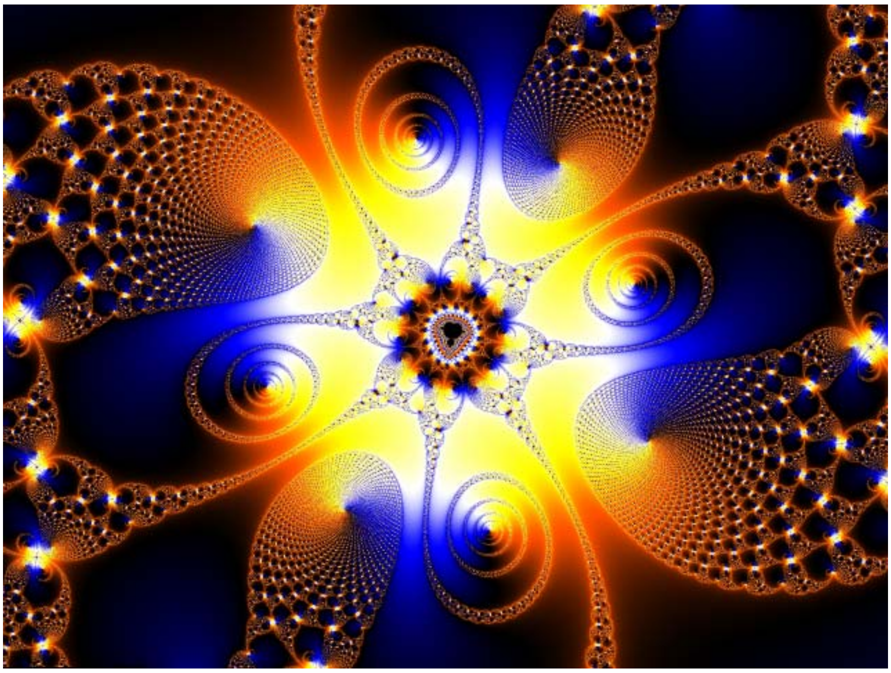
I found my formula during a longer "expedition" which started with the question: What will happen when representing sequences of numbers on a circle instead on a number ray? As you might guess this results in a modular reduction of the sequences. And this leads us to in number theory.

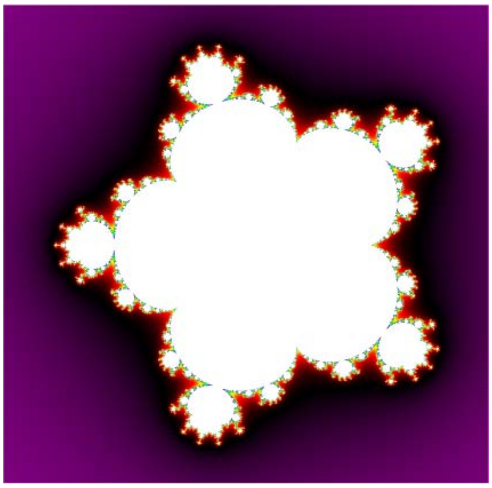
You can find the sequence of powers of two and three presented on a circle in one of my papers of DERIVE applications "Light in the coffee cup" . Some of these papers did you publish in earlier DNLs.

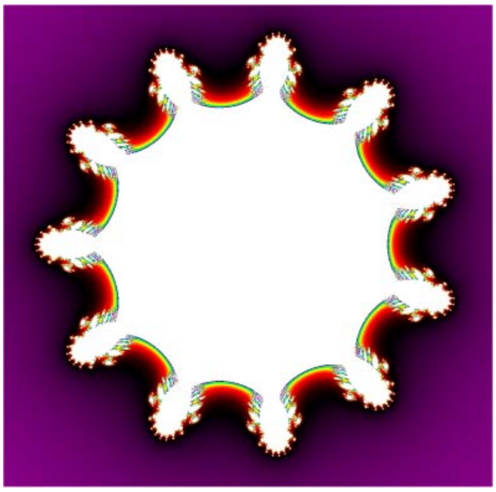
The reason for that lies in the depth of number theory. There is the concept of the "primitive root": the constant base of my sequence of powers must be primitive root of the prime number which indicates the number of points on the circle (the module). You can find a website which gives the respective primitive roots. You will notice that square numbers cannot be primitive roots (which can easily be proved). So 4 cannot be primitive root of the number of points on the circle. It follows that a clover leaf with 4 -1 = 3 leaves is not easy to create.

Best regards
Roland









Download the freeware fractalizer from www.fractalizer.de, Josef