THE DERIVE - NEWSLETTER #90

ISSN 1990-7079

THE BULLETIN OF THE

DERIVE

USER GROUP

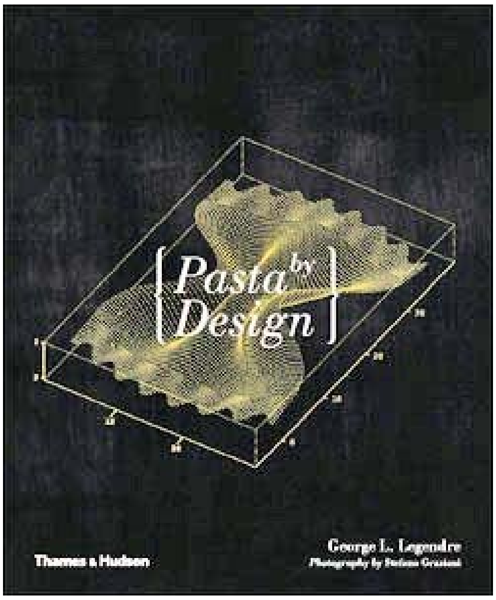
+ CAS-TI

Contents:

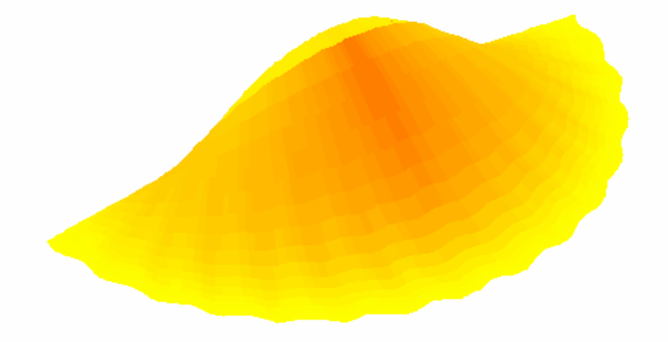
June 2013

Do you like Pasta? Do you like Mathematics?

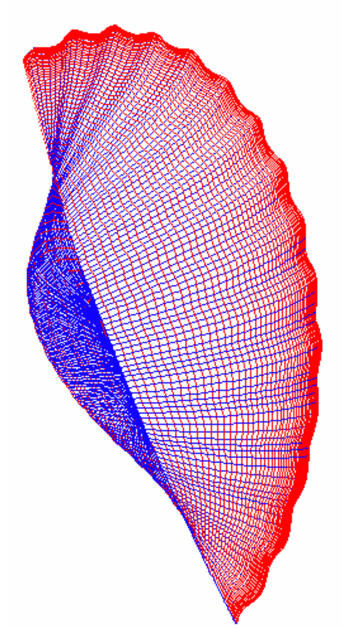
Then you possibly will like

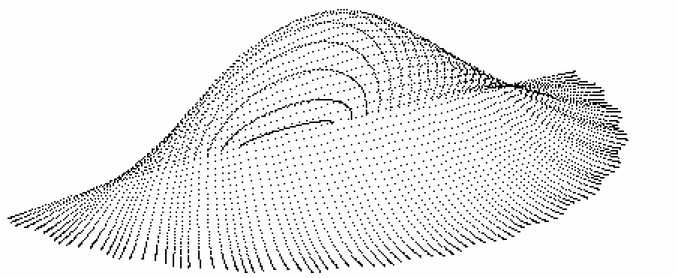


Do you like Agnolotti?



Pasta by Design, George L. Legendre, Thames & Hudson, ISBN 9780500515808
Deutsche Ausgabe: Pasta und Design, Springer Spektrum, ISBN 978-3-8274-2978-0





Or do you prefer Gnocchi?

Buon Appetito!

Liebe DUG-Mitglieder,

In Österreich haben die Sommerferien begonnen und da will ich gleich allen jenen, die schon oder demnächst Ferien haben, einen schönen Sommer und erholsame Tage und Wochen wünschen.

Mit diesem DNL will ich vor allem „alte Schulden" einlösen. Wie so oft haben auch kurze Anfragen und Beiträge ihre Eigendynamik entwickelt und zu weiteren Untersuchungen angespornt. Aus „*Building Towers*" habe ich die Aufforderung zu eigenen Experimenten ebenso gerne aufgegriffen wie mich „*Fresnel Integrals ...*" an frühere DNL-Beiträge zur Cornu Spirale und zu den „natürlichen Gleichungen" erinnert haben. Da entwickelte sich gleich ein lebhafter Briefwechsel mit David Halprin (*Two good Turns*). Davids interessante Laufbahn als Mathematiker verdient einen eigenen Beitrag im nächsten DNL.

Sie finden eine kurze Anmerkung von Erik van Lantschoot zu seinem *Brüsseler Tor in Dendermonde* (DNL#89). Kürzlich hat er eine weitere Ausarbeitung basierend auf seinen intensiven Nachforschungen in historischen Quellen geschickt.

Christine Kova besuchte vor einigen Jahren einen *DERIVE* & GeoGebra Workshop "Fadengrafik". Später schickte sie Bilder mit Mustern, die ihre Schülerinnen nicht nur am PC sondern auch mit Holz, Nägeln und Faden erzeugt haben. Kürzlich erhielt ich wieder Bilder, nun wagten sich die Schülerinnen bereits in den 3D-Raum. Ihre Skulpturen gegossen in Acrywürfel werden sogar in einem Wiener Museum ausgestellt. Es ist schön, dass ein Mathematik-Workshop so nachhaltige Wirkung zeigen kann.

Herbert Nieder verdanke ich den Hinweis auf das „Pasta-Buch". Obwohl ich kein besonderer Nudelfan bin, habe ich doch Appetit bekommen, die Agnolotti und Gnocchi zu modellieren. Im Internet können Sie noch einige Beispiele finden.

Dear DUG Members,

Summer Holidays have started in Austria. I'd like to wish all of you already being in holidays or waiting for them a wonderful summer and relaxing days and weeks.

With this DNL I will settle "old debts". How so often short requests and mails develop their own dynamics and inspire for further investigations. I took seriously Roland's invitation in "*Building Towers*" to undertake own experiments and I was reminded on earlier DNL-contributions on the Cornu Spiral and on "Natural Equations" reading "*Fresnel Integrals ...*". An animated communication with David Halprin followed (*Two good Turns*). David's rich career as mathematician deserves an own contribution in the next DNL.

You can find a short note from Erik van Lantschoot concerning his *Brussels Gate in Dendermonde* (DNL#89). Just recently he sent another paper based on his intense investigations of historical resources.

Christine Kova attended a DERIVE & GeoGebra workshop "Stitching Patterns" several years ago. Later she sent pictures with patterns produced by her students not only by means of mathematics but with wood, nails and threads. Three weeks ago I received some new pictures. Her students transferred the graphics into 3D-space. Their sculptures were cast in acryl cubes. Some of their works of (mathematical) art are exhibited in a Viennese museum. It is great that a mathematics-workshop can achieve such a sustained effect.

We owe the note on the "Pasta-book" to Herbert Nieder from Hamburg. Although I am not a "noodles fan" at all, I got some appetite modelling Agnolotti and Gnocchi. You can find some more examples in the web.

Viele Grüße, Best regards

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: September 2013

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
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
Cesar Multiplication, G. Schödl, AUT
Henon & Co; Find your very own Strange Attractor, J. Böhm, AUT
Rational Hooks, J. Lechner, AUT
Simulation of Dynamic Systems with various Tools, J. Böhm, AUT
Bélidor, E. v. Lantschoot

and others

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

There are many resources dealing with Wythoff's Game. Roland presents this game together with other ones which show a fascinating connection to the Golden Section.

Building Towers

Roland Schröder, Germany

Which numbers give 1 as sum and -1 as product?

```
#1:   SOLVE([a + b = 1, a·b = -1], [a, b])
```

$$\#2: \quad \left[a = \frac{\sqrt{5}}{2} + \frac{1}{2} \wedge b = \frac{1}{2} - \frac{\sqrt{5}}{2},\ a = \frac{1}{2} - \frac{\sqrt{5}}{2} \wedge b = \frac{\sqrt{5}}{2} + \frac{1}{2}\right]$$

The greater solution

$$\#3: \quad \phi := \frac{\sqrt{5}}{2} + \frac{1}{2}$$

has raised to fame not only in mathematics and got – like other famous numbers like π and e – its own symbol Φ. We meet the irrational number Φ in many mathematical contexts but also in nature, architecture, arts and music. There are numerous resources in books and on the web under the key-word "*Golden Section*". If you are lucky you can find a relation between games played with counters and the number Φ. One of them is Wythoff's Game named after its inventor Willem Abraham Wythoff (1865 – 1939). These are the rules:

Two piles (= towers) of different number of counters are offered two players.
The players have to remove counters alternatively according to the following instructions:
- A player is allowed to remove the <u>same number</u> of counters from <u>both piles</u>,
- A player is allowed to remove <u>as many counters as he likes</u> from <u>one pile</u>.
- <u>Winner</u> is the player who removes the <u>last counter</u>.

Wythoff has analysed his game completely by himself. The winner has to know special pairs of numbers which describe the number of counters forming the two piles which make sure that he will win the game. Take the pair (1, 2) as an example. If a player is able to leave two piles with 1 and 2 counters then he will win. Wythoff [1] did not only give the pairs of numbers which should be known by the later winner but he also presented the respective formula:

$$(\text{FLOOR}(n\cdot\Phi), \text{FLOOR}(n\cdot\Phi^2)) \text{ with } n \in \text{N}$$

Wythoff said that he had "drawn the formula out of the hat". It was proofed by Harold Scott McDonald Coxeter [2] (1907 – 2003).

First of all we would like to build the DERIVE-function „`win_move(x,y)`" which delivers to a given pair (x,y) of pile heights the respective pair according to Wythoffs's formula which should be achieved by the later winner to come closer to the final win. There exists to every component x a matching component y such that $[x,y]$ can be presented in one of the two forms:

$$[\text{FLOOR}(n\cdot\Phi), \text{FLOOR}(n\cdot\Phi^2)] \text{ or } [\text{FLOOR}(n\cdot\Phi^2), \text{FLOOR}(n\cdot\Phi)].$$

---

[1] W. A. Wythoff, „*A Modification of the Game of Nim*", Nieuw. Archief voor Wiskunde (2), volume 7, pages 199 – 202, 1907

[2] H.S.M. Coxeter, *The Golden Section, Phyllotaxis and Wythoff's Game*, Scripta Mathematica, Vol. XIX, page 142, New York, 1953

Exchanging the components takes into consideration that the order of the piles (towers) is not important. The fact that to each component x a component y is existing results from Coxeter's proof which will not be repeated in this article. Instead of this we will show an example supported by *DERIVE*:

```
#4:   upper(m) := VECTOR([FLOOR(n·φ), FLOOR(n·φ^2)], n, 1, m)

#5:   lower(m) := VECTOR([FLOOR(n·φ^2), FLOOR(n·φ)], n, 1, m/φ)

#6:   v(m) := APPEND(lower(m), upper(m))
```

Simplifying `v(10)` results in:

$$\#7:\quad v(10) = \begin{bmatrix} 2 & 1 \\ 5 & 3 \\ 7 & 4 \\ 10 & 6 \\ 13 & 8 \\ 15 & 9 \\ 1 & 2 \\ 3 & 5 \\ 4 & 7 \\ 6 & 10 \\ 8 & 13 \\ 9 & 15 \\ 11 & 18 \\ 12 & 20 \\ 14 & 23 \\ 16 & 26 \end{bmatrix}$$

As you can see the left component runs through all natural numbers from 1 to FLOOR(10·Φ). It is important for the tactical analyse of the game that none of these pairs can be reached from another one performing only one move. The opponent is forced to leave this set of pairs. In case of a pair [x,y] fulfilling the condition $y > \Phi\cdot x$ we can reduce y to generate one of the pairs from v(m) – with $m > x$.

The respective (still incomplete) *DERIVE*-function is:

$$\text{VECTOR}(\text{IF}(y > \phi\cdot x \wedge (v(m))_{n,1} = x,\ [x,\ (v(m))_{n,2}],\ \text{else ???}),\ n,\ 1,\ \phi\cdot m)$$

In case of $y < x/\Phi$ then we can reduce x to reach a pair from v(m):

$$\text{IF}\left(y < \frac{\phi}{x} \wedge (v(m))_{n,2} = y,\ [(v(m))_{n,1},\ y],\ \text{else ???}\right)$$

The third case $x/\Phi < y < \Phi \cdot x$ is still remaining. In this case we reduce $[x,y]$ in both piles in such a way that the difference $x - y$ equals the difference of a pair from $v(m)$. So we have to substitute for `else ???` the construct:

$$\mathrm{IF}\left(\frac{x}{\phi} < y < \phi \cdot x \wedge (v(m))_{n,1} - (v(m))_{n,2} = x - y,\ \left[(v(m))_{n,1},\ (v(m))_{n,2}\right],\ 0\right)$$

The complete function reads now:

$$\#8:\quad \mathrm{win}(x, y, m) := \mathrm{VECTOR}\left(\mathrm{IF}\left(y > \phi \cdot x \wedge (v(m))_{n,1} = x,\ \left[x,\ (v(m))_{n,2}\right],\ \mathrm{IF}\left(y < \frac{x}{\phi} \wedge (v(m))_{n,2} = y,\ \left[(v(m))_{n,1},\ y\right],\ \mathrm{IF}\left(\frac{x}{\phi} < y < \phi \cdot x \wedge (v(m))_{n,1} - (v(m))_{n,2} = x - y,\ \left[(v(m))_{n,1},\ (v(m))_{n,2}\right],\ 0\right)\right)\right),\ n,\ 1,\ \phi \cdot m\right)$$

#9: win(13, 5, 10)

#10: [0, 0, 0, 0, 0, 0, 0, [3, 5], 0, 0, 0, 0, 0, 0, 0, 0]

We remove the zeros and make m dependent on x and y. Function `win_move(x,y)` is now ready:

$$\#11:\quad \mathrm{win\_move}(x, y) := (\mathrm{SELECT}(w \neq 0,\ w,\ \mathrm{win}(x,\ y,\ \mathrm{MAX}(x,\ y))))_1$$

#12: win_move(13, 5) = [3, 5]

#13: win_move(30, 20) = [26, 16]

#13 needs 5.3 seconds calculation time.

If our oppponent leaves piles of heights x and y we run `win_move(x,y)` in order to learn how to reduce the number of counters to gain the victory. In case of (13,5) we have to remove 10 counters from the higher pile and in case of (30,20) our strategy must be removing four counters from each pile.

Two special cases are missing: in case $x = y$ it is not necessary to find a *DERIVE*-answer because it is obvious that one will win with the next move. Case `win_move(x,y) = [x,y]` is only possible if the opponent starts the game and possibly knows the strategy. If so then we will loose – we can try our luck and make any move.

#14: win_move(16, 26) = [16, 26]

Agreeing upon $y > x$ function `win(x,y,m)` (expression #8) can be simplified.

Comment of the editor: I remember DNL#45 from 2002 when Richard Schorn wrote an article about *Wythoff's Nim*. He closed his ACDC-contribution as follows:

The generation of the two sequences relies on some of the mentioned properties: Start with 1 as the *x*-value of the first safe pair. Add this to its position number to obtain 2 as the *y*-value. The *x*-value of the next pair is the smallest positive integer not previously used, it is 3. The *y*-value is 5, the sum of 3 and the position number. The next *x*-value is 4, the smallest integer not yet used.

My program for DERIVE is based on this algorithm in Martin Gardner's book *Penrose Tiles to Trapdoor Ciphers.* (Richard Schorn)

```
     w(z, W := [[1, 1, 2]], x := 3, i := 2, M := {1, 2}) :=
       Prog
         Loop
           If i - 1 = z
              RETURN W
           W := APPEND(W, [[i, x, i + x]])
#1:        M := M ∪ {x, i + x}
           i := i + 1
           Loop
             If MEMBER?(x, M)
                x := x + 1
                exit
```

$$w(10) = \begin{bmatrix} 1 & 1 & 2 \\ 2 & 3 & 5 \\ 3 & 4 & 7 \\ 4 & 6 & 10 \\ 5 & 8 & 13 \\ 6 & 9 & 15 \\ 7 & 11 & 18 \\ 8 & 12 & 20 \\ 9 & 14 & 23 \\ 10 & 16 & 26 \end{bmatrix}$$

See the TI-Nspire-version from 2013 which can neither use the `UNION` of sets nor the useful `MEMBER?`-function:

wythoff(20)

$$\begin{bmatrix} 1 & 1 & 2 \\ 2 & 3 & 5 \\ 3 & 4 & 7 \\ 4 & 6 & 10 \\ 5 & 8 & 13 \\ 6 & 9 & 15 \\ 7 & 11 & 18 \\ 8 & 12 & 20 \\ 9 & 14 & 23 \\ 10 & 16 & 26 \\ 11 & 17 & 28 \\ 12 & 19 & 31 \\ 13 & 21 & 34 \\ 14 & 22 & 36 \\ 15 & 24 & 39 \\ 16 & 25 & 41 \\ 17 & 27 & 44 \\ 18 & 29 & 47 \\ 19 & 30 & 49 \\ 20 & 32 & 52 \end{bmatrix}$$

Fertig

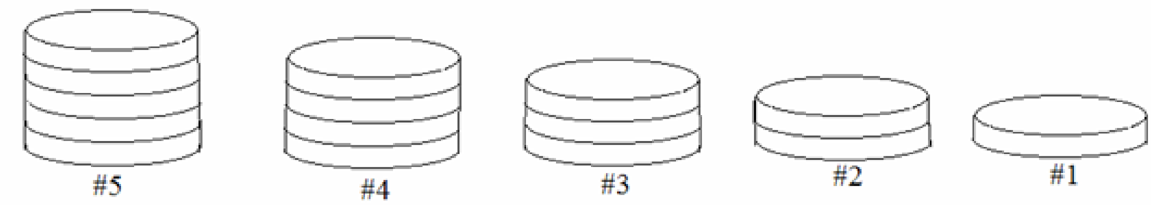
1/99

wythoff 9/16

```
Define wythoff(z)=
Prgm
Local i,li,lx,ly,lm
li:=newMat(z,1):lx:=newMat(z,1):ly:=newMat(z,1)
li[1,1]:=1:lx[1,1]:=1:ly[1,1]:=2:x:=3
lm:={1,2}
For i,2,z
  li[i,1]:=i:lx[i,1]:=x:ly[i,1]:=i+x
  lm:=augment(lm,{x,i+x})
   SortA lm
   For k,1,dim(lm)−1
   If lm[k+1]−lm[k]>1 Then
     x:=lm[k]+1
     Exit
   EndIf
   EndFor
EndFor
Disp augment(augment(li,lx),ly)
EndPrgm
```

Now let's proceed with Roland Schröder's paper!

The author of this paper has – based on a small *DERIVE* program – discovered two more games with towers of counters which both are leading to the sequence <FLOOR($\Phi\cdot n$)$_{n\in\mathbb{N}}$>. Both games are played by only one player who intends to discover mathematical patterns in his results. Both games are starting with a sequence of piles of counters according to the figure given below.



The sequence of "*Countertowers*" or "Cointowers"can be continued to the left as far as one likes. The positions of the towers are numbered from right to left. When the game starts each tower consists of as many counters as its number says. For the first game we can assume an infinite number of piles.

Every move consists of taking the last pile (first from right) and distributing its counters piece for piece on the next piles. We can describe the constellation at the beginning of the game as:

... 7, 6, 5, 4, 3, 2, 1. Then we have:

... 7, 6, 5, 4, 3, 3 after the first move and

... 7, 6, 6, 5, 4 after the second move, etc.

The next moves are left for the reader. (He/she might use the *DERIVE* program which will be presented at the end of the article.) Inspecting the piles to be distributed in the next move and counting its counters we find the sequence <1, 3, 4, 6, ...>. We can conjecture that here again the sequence of the first components of Wythoff's game appears.

The second game deals with a finite number of towers. As the moves follow the same rules as given in the first game there will come a moment when no more piles are at the left end of the row to deposit counters. In this case we proceed by adding one-counter piles in left direction. See an example starting with 4 towers:

4, 3, 2, 1
4, 3, 3
1, 5, 4
1, 1, 2, 6
1, 1, 1, 2, 2, 3
1, 1, 2, 3, 3
1, 2, 3, 4

Having reached the reverse order all piles are moving to the left without changing the number sequence. It needed 6 moves to reach <1, 2, 3, 4> from <4, 3, 2, 1>. One can try starting with 5, 6, 7, ... towers and count how many moves are necessary to reach the reverse constellation of the starting towers. Doing this with paper and pencil (or with the *DERIVE*-program) you will get the following table:

Number of towers at the beginning	1	2	3	4	5	6
Number of moves to reach the mirror position	1	3	4	6	8	9

This sequence in the 2nd row is again very suspicious to form the sequence of the *x*-values from Wythoff's game!

We will simulate the second game using *DERIVE*. This simulation will give the results for the infinite form, too. Let's start with an initial list (vector) representing the towers with heights from 1 to *n*.

```
#1:   init(n) := REVERSE(VECTOR(k, k, n))
```

`h(z)` removes the last element of any sequence of towers (the utmost right tower) and `i(z)` transforms this pile of counters in a list of single counters.

```
#2:   h(z) := REVERSE(REST(REVERSE(z)))

#3:   i(z) := VECTOR(1, m, FIRST(REVERSE(z)))
```

(We use the list-oriented commands like `REVERSE`, `FIRST` and `REST` because *DERIVE* is programmed in LISP which is based on working internally with lists.)

We rename the longer list as `y` and the shorter one as `x`:

```
#4:   [a := DIMENSION(h(z)), b := DIMENSION(i(z))]

#5:   [y := IF(a > b, h(z), i(z)), x := IF(a ≤ b, h(z), i(z))]
```

The order of towers `h(z)` and `i(z)` are compared with respect to the number of towers. The greater number is *y* and the smalles one is *x*. *y* must be divided in two lists: in `f(y)` which consists of the same number of elements as *x* and `g(y)` containing the remaining elements.

$$\#6:\quad f(y) := \mathrm{VECTOR}(y_n,\ n,\ |a - b| + 1,\ \mathrm{DIMENSION}(y))$$

$$\#7:\quad g(y) := \mathrm{VECTOR}(y_n,\ n,\ 1,\ |a - b|)$$

Now we can add `f(y)` and `x` and append `g(y)` in order to get the next sequence of towers.

```
#8:   new_towers(z) := APPEND(g(y), f(y) + x)
```

For better understanding the procedure we will demonstrate the process starting with a sequence of six piles:

```
z := [1, 1, 1, 2, 2, 8]

h(z) = [1, 1, 1, 2, 2]

i(z) = [1, 1, 1, 1, 1, 1, 1, 1]

[a, b] = [5, 8]

y = [1, 1, 1, 1, 1, 1, 1, 1]

x = [1, 1, 1, 2, 2]

f(y) = [1, 1, 1, 1, 1]

g(y) = [1, 1, 1]

f(y) + x = [2, 2, 2, 3, 3]

APPEND(g(y), f(y) + x) = [1, 1, 1, 2, 2, 2, 3, 3]
```

All steps are collected in one iterative procedure, starting with `init(n)`:

```
#9:   towers(n) := ITERATES(new_towers(z), z, init(n))
```

My first game starts with 5 towers:

```
towers(5) = [[5, 4, 3, 2, 1], [5, 4, 3, 3], [6, 5, 4], [1, 1, 7, 6], [1, 1, 1, 2, 2, 8], [1, 1,
  1, 2, 2, 2, 3, 3], [1, 1, 1, 2, 3, 3, 4], [1, 1, 2, 3, 4, 4], [1, 2, 3, 4, 5], [1, 2, 3, 4,
  5]]
```

You can find our *z* from above as 5th element and its follower. It needs 8 moves to come from [5,4,3,2,1] to [1,2,3,4,5].

Question #1: Can we confirm our conjecture from above about the number of elements until reaching the reverse constellation? *DERIVE*'s iteration stops when two iterations remain the same.

```
DIM(towers(10)) = 18

#18:   moves(n) := APPEND([1], VECTOR(DIM(towers(k)) - 2, k, 2, n))

moves(30) = [1, 3, 4, 6, 8, 9, 11, 12, 14, 16, 17, 19, 21, 22, 24, 25, 27, 29, 30, 32,
  33, 35, 37, 38, 40, 42, 43, 45, 46, 48]
```

$$\text{VECTOR}\left(\text{FLOOR}\left(\frac{k\cdot(1+\sqrt{5})}{2}\right),\ k,\ 1,\ 30\right)$$

```
[1, 3, 4, 6, 8, 9, 11, 12, 14, 16, 17, 19, 21, 22, 24, 25, 27, 29, 30, 32, 33, 35, 37,
  38, 40, 42, 43, 45, 46, 48]
```

Please compare the sequences!

Question #2: How can we derive the results of the infinite game from these results?

From the above `towers(5)`-result we can read off the last numbers of the intermediate steps: [1,3,4,6,8]. Then we would need a 6th and an 7th pile of counters to proceed. The sequence needed is the increasing partial sequence of the last elements of the `towers(n)`-results. A short program can help:

```
game1(n, g, l := []) :=
  Prog
    g := VECTOR(FIRST(REVERSE(v)), v, towers(n))
    Loop
      l := APPEND(l, [FIRST(g)])
      If FIRST(g) > g↓2
         RETURN l
      g := REST(g)

game1(20) = [1, 3, 4, 6, 8, 9, 11, 12, 14, 16, 17, 19, 21, 22, 24, 25, 27, 29, 30, 32]
```

Again we have the first components of Wythoff's game.

The interested reader might have found out by himself that there might "sleep" some other exciting discoverings. It could be worthwhile to change the start conditions. What we showed here are only results which affirm the conjectures, there are no proofs. The proofs together with other rules can be found in *Roland Schröder, "Der Goldene Schnitt in Türmen aus Spielsteinen", in Mathematische Semesterberichte, Band 60, Heft 1, Springer 2013.*

http://scienceindex.com/stories/2251230/Der_Goldene_Schnitt_in_Trmen_aus_Spielsteinen.html

Another Game – Another Pattern?

Roland's last paragraph made me considering about "*other exciting discoveries*". What to discover? So I had the idea to start with towers consisting of odd numbers of counters: [..., 7, 5, 3, 1] and playing the game from above:

```
init2(n) := REVERSE(VECTOR(2·k - 1, k, n))

init2(10) = [19, 17, 15, 13, 11, 9, 7, 5, 3, 1]
```

```
towers2(n) := ITERATES(new_towers(z), z, init2(n))

towers2(5)

[[9, 7, 5, 3, 1], [9, 7, 5, 4], [1, 10, 8, 6], [1, 1, 1, 2, 11, 9], [1, 1,
  1, 1, 2, 2, 2, 3, 12], [1, 1, 1, 1, 2, 2, 2, 2, 3, 3, 3, 4], [1, 1, 1,
  1, 2, 2, 2, 3, 4, 4, 4], [1, 1, 1, 1, 2, 2, 3, 4, 5, 5], [1, 1, 1, 1, 3,
  3, 4, 5, 6], [1, 1, 2, 2, 4, 4, 5, 6], [1, 2, 3, 3, 5, 5, 6], [2, 3, 4,
  4, 6, 6], [1, 3, 4, 5, 5, 7], [1, 1, 2, 4, 5, 6, 6], [2, 2, 3, 5, 6, 7],
  [1, 1, 3, 3, 4, 6, 7], [1, 2, 2, 4, 4, 5, 7], [1, 2, 3, 3, 5, 5, 6]]
```

We cannot find the reverse order of the initial status. But we notice a cycle of length 7 starting with towers 1, 2, 3, 4, 5, 5, 6.

We can define a function to calculate the length of the period:

```
per_length(n) := DIM(towers2(n)) - POSITION(FIRST(REVERSE(towers2(n))),
   towers2(n))

per_length(5) = 7
```

How long are the periods for games from 1 to 30 starting piles with odd numbers of counters:

```
VECTOR([k, per_length(k)], k, 1, 30)'

145 sec
```

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	3	4	6	7	1	10	11	13	14	16	17	18	20	21	23	24	25	27	28	30	31	33	34	35	37	38	40	41	42

5 piles lead to a period of length 7. I cannot detect any generating rule or pattern, can you?

Next question: How many moves are necessary until having finished the first cycle?

```
moves2(n) := APPEND([[1, 1]], VECTOR([k, DIM(towers2(k)) - 2], k, 2, n))`
```

$$\text{moves2(20)} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 \\ 1 & 5 & 8 & 12 & 16 & 22 & 23 & 27 & 31 & 34 & 146 & 42 & 209 & 147 & 199 & 348 & 86 & 468 & 279 & 521 \end{bmatrix}$$

Another function lists the length of the pre-periods:

```
pre_per2(n) := VECTOR([k, POSITION(FIRST(REVERSE(towers2(k))), towers2(k))
  - 1], k, 1, n)`
```

$$\text{pre\_per2(20)} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 \\ 0 & 3 & 5 & 7 & 10 & 22 & 14 & 17 & 19 & 21 & 131 & 26 & 192 & 128 & 179 & 326 & 63 & 444 & 253 & 494 \end{bmatrix}$$

I will check for the game starting with 6 piles form 1, 3, …, 11:

```
towers2(6) = [[11, 9, 7, 5, 3, 1], [11, 9, 7, 5, 4], [12, 10, 8, 6], [1,
  1, 1, 13, 11, 9], [1, 1, 1, 1, 2, 2, 2, 14, 12], [1, 1, 1, 1, 2, 2, 2,
  2, 3, 3, 3, 15], [1, 1, 1, 1, 2, 2, 2, 2, 3, 3, 3, 3, 4, 4, 4], [1, 1,
  1, 1, 2, 2, 2, 2, 3, 3, 4, 4, 5, 5], [1, 1, 1, 1, 2, 2, 2, 2, 4, 4, 5,
  5, 6], [1, 1, 1, 1, 2, 2, 3, 3, 5, 5, 6, 6], [1, 1, 1, 1, 2, 3, 4, 4, 6,
  6, 7], [1, 1, 1, 2, 3, 4, 5, 5, 7, 7], [1, 1, 2, 3, 4, 5, 6, 6, 8], [2,
  2, 3, 4, 5, 6, 7, 7], [3, 3, 4, 5, 6, 7, 8], [1, 1, 4, 4, 5, 6, 7, 8],
  [1, 2, 2, 5, 5, 6, 7, 8], [1, 2, 3, 3, 6, 6, 7, 8], [1, 2, 3, 4, 4, 7,
  7, 8], [1, 2, 3, 4, 5, 5, 8, 8], [1, 2, 3, 4, 5, 6, 6, 9], [1, 1, 2, 3,
  4, 5, 6, 7, 7], [1, 2, 3, 4, 5, 6, 7, 8], [1, 2, 3, 4, 5, 6, 7, 8]]
```

This is a strange result, indeed. The period has length 1 (compare with the list of the period lengths) and the pre-period consists of 22 elements.

You are friendly invited to offer other ideas and games – and conjectures – and solutions – and proofs, Roland and Josef.

It was during Christmas holidays when my wife Noor and I made walk through winter wonder world. Suddenly the mobile phone interrupted the silence. At first I didn't know who was calling. It was a colleague with whom I collaborated many years ago developing a curriculum for a special type of vocational schools. He was teacher at a Secondary school for forestry in Styria.

After exchanging some personal small talk about ourselves (family, retirement, health, ...) he tried to explain his problem. I said, "Franz, send an email explaining your request and add one or two examples. I believe to understand you, but written text is quite better". Coming home from our walk I turned on the notebook and found his mail:

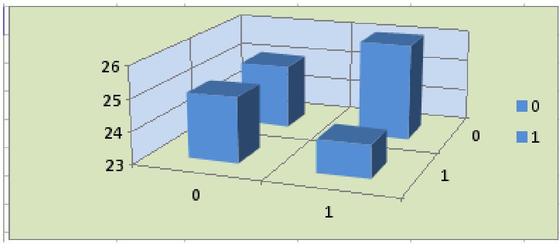
Franz Jauk's Statistics Problem

Dear Josef,
... I have a small problem – in earlier times we produced our own programs ...

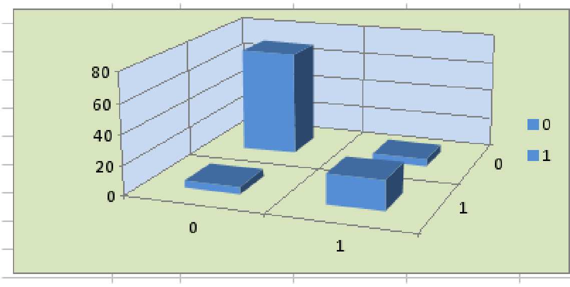
The problem deals with finding a correlation coefficient. Given are a few pairs of values and their frequencies, e.g. (5,6) 25 times, (4,3) 17 times, etc. I don't want to enter 25 times (5,6) and then 17 times (4,3), etc. I would like to enter 5,6,25 followed by 4,3,17, etc. That is it.

Slope and intercept of the regression line are not so important. It would be great if I could represent the regression line but this is not a „must" for me.

Doing this in Excel I have to enter all pairs into the table – and this is boring. At the other hand the Excel 3D-representation looks fine: the value pairs in the *xy*-plane and the frequencies in *z*-direction.



		MM2 >>>			
		0	1		
MM	0	25	26	Korrelation=	-0,02
1	1	25	24		
		Absolute Häufigkeiten			



		MM2 >>>			
		0	1		
MM	0	70	5	correlation=	0,73
1	1	5	20		
		absolute frequencies			

Please excuse that I am bothering you in your well-deserved retirement.

Best regards

Franz

Dear Franz,

... Now to your problem:

Nice to do it with other technologies (btw, it is not too boring working with Excel by copying down the respective cells). What you are expecting can easily be performed with TI-Nspire. It can be done without any additional programming.

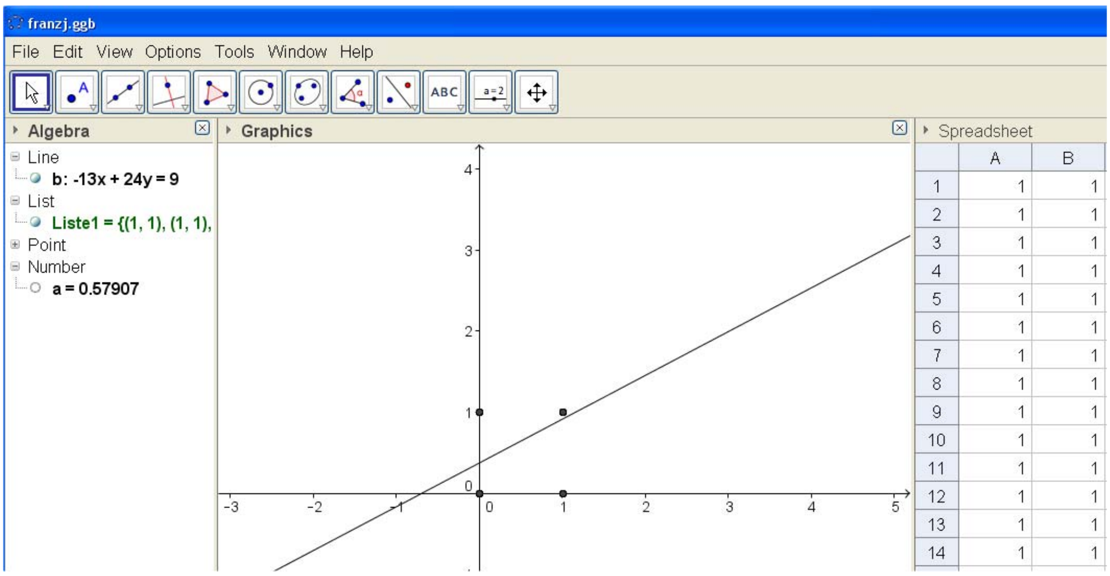
I am quite sure that you can do it with the older TI-handhelds (TI-92, V 200 and TI-84), too. But at the moment – I am on holidays – I don't have these tools at my disposal.

The screen shot below shows the result using one of your data sets. The regression line could easily be plotted together with the scatter diagram. What cannot be done is a 3D-representation.

	A	B	C	D	E	F	G
◆					=TwoVar(a[],b[],c[]): CopyV		=LinRegMx(a[],b[],c[]
1	1	1	20	Title	Statistiken mit zwei Varia...	Title	Lineare Regression ...
2	0	0	70	$\bar{x}$	0.25	RegEqn...	m*x+b
3	0	1	5	Σx	25.	m	0.733333
4	1	0	5	Σx²	25.	b	0.066667
5				sx := sn-...	0.435194	r²	0.537778
6				σx := σn...	0.433013	r	0.733333
7				n	100.	Resid	{0.2,⁻0.0666666666...
8				$\bar{y}$	0.25		
9				Σy	25.		
10				Σy²	25.		
11				sy := sn-...	0.435194		
12				σy := σn...	0.433013		
13				Σxy	20.		
14				r	0.733333		

r is the correlation coefficient and r^2 the coefficient of determination .

You can also work with GeoGebra. But here you have to enter the data pairs (copying down the cells like with Excel). I did not find a way to work with triples (pairs together with their frequencies).



The data set consists of 25 times (0,0), 20 times (1,1), 15 times (0,1) and 5 times (1,0). List **Liste1** was created by a mouse click into the two spreadsheet columns. The correlation coefficient **a = 0.57907** was calculated by **CorrelationCoefficient(Liste1)** and the regression line **b = -13x+24y=8** by **FitLine(Liste1)** (in German: **Korrelationskoeffizient(Liste1)** and **Trendlinie(Liste1)**)

It is not possible to have a 3D-representation of the 4 data piles.

Let us turn to DERIVE. Two short programs produce the statistics and a 3D-plot as requested:

```
reg_korr(md, mdext, d_, i, j, l1, l2, r1, r2) :=
  Prog
    d_ := DIM(md)
    mdext := []
    i := 1
    Loop
      If i > d_ exit
      j := 1
      Loop
        If j > md↓i↓3 exit
        mdext := APPEND(mdext, [[md↓i↓1, md↓i↓2]])
        j :+ 1
      i :+ 1
    l1 := FIT([x, a·x + b], mdext)
    l2 := FIT([x, a·x + b], SWAP_ELEMENTS(mdext`, 1, 2)`)
    [r1 := ∂(l1, x), r2 := ∂(l2, x)]
    ["Reg1: ", l1; "Reg2: ", l2; "r^2: ", r1·r2; "r: ", √(r1·r2)]
```

```
plot3d(md, mdext, pl, i, j) :=
  Prog
    mdext := []
    i := 1
    Loop
      If i > 4 exit
      pl := []
      j := 1
      Loop
        If j > md↓i↓3 exit
        pl := APPEND(pl, [[[md↓i↓1, md↓i↓2, j]]])
        j :+ 1
      mdext := APPEND(mdext, [pl])
      i :+ 1
    mdext
```

See how they are working:

(1) I enter the data as a matrix (named mm):

$$mm := \begin{bmatrix} 1 & 1 & 55 \\ 0 & 0 & 25 \\ 0 & 1 & 15 \\ 1 & 0 & 5 \end{bmatrix}$$

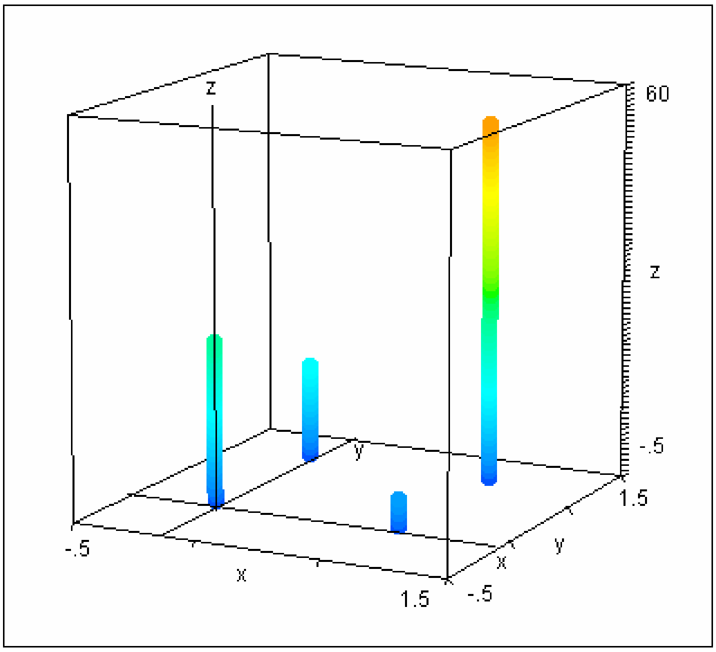
(2) `reg_korr(matrix)` returns all interesting statistics:

$$\begin{bmatrix} \text{Reg1:} & 0.5416666666\cdot x + 0.375 \\ \text{Reg2:} & 0.619047619\cdot x + 0.1666666666 \\ \text{r^2:} & 0.3353174603 \\ \text{r:} & 0.5790660241 \end{bmatrix}$$

You are invited to compare with the GeoGebra results from above.

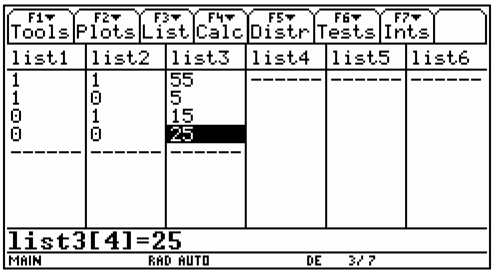
(3) Apply plot3d(matrix) and receive a nice 3D-representation of the pillars.

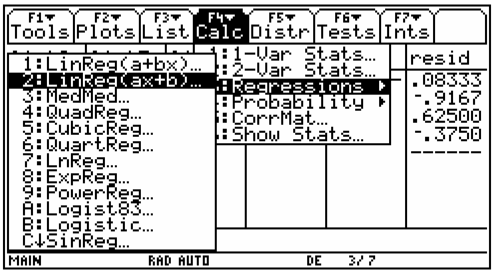
plot3d(mm)

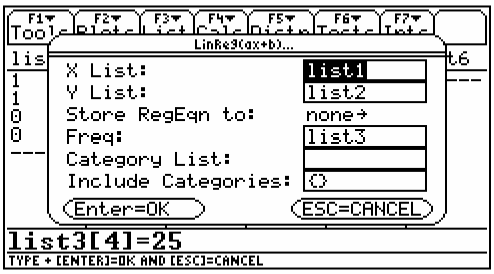


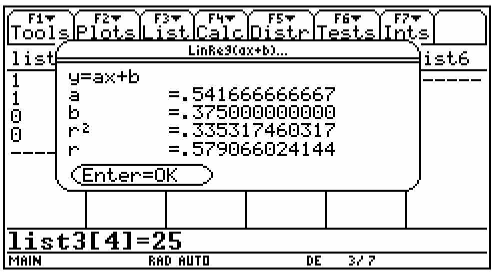
Back home some days later I could accomplish my proposals working with my good old Voyage 200:

The Stats/List Editor is the appropriate tool on the Voyage 200:









Dear Franz, now it is your choice which tool to use. It was very interesting for me to compare the several possibilities treating your question. Now I have a question: Why do you need this tool in order to find the statistics for these special four data pairs?

Josef

Dear Josef,

many thanks for your efforts in your retirement. Your advice was very helpful for me.

… You are right, I owe you an explanation for what I am needing all the procedures. Take any population and collect the presence of two certain properties (in German: Merkmale MM).

I have been interested in the influence of the existence of the properties on the correlation. I describe an example: 100 people form the population. There are some people with both properties (= (1,1)), with none (= (0,0)) or with only one of them ((1,0) or (0,1)). I wanted to find out the influence of the frequencies of the pairs on the correlation. 55 persons have both properties, 25 have none, 5 have the first one only and 15 the second one only.

You know that I was teacher at a special vocational school for forestry. But I have changed the "faculty" in my retirement. I am very interested in "Gender Studies" and I need sometimes statistical evidence to underline special conjectures about our social life. To make it short: I'd like to know "*What is keeping our world together*".

Best regards and many thanks again,

Franz

Piotr Trebisz's FRESNEL INTEGRALS

In the *DERIVE* package is a Utility file called "`FresnelIntegrals.mth`" containing definitions for *Fresnel*-Integrals. Recently I experienced that the definitions presented there are quite problematic. After 15 minutes calculation time I found new formulae for the Fresnel Integrals which seem to perform much better. As they might be useful for other DERIVE Users, too, I wouldn't like to deny them for you.

After treating the "logarithmic spirals" in great detail earlier I wanted to work on other interesting spirals. And I found the *Clothoids* or *Cornu Spirals* a very interesting object. A clothoid is defined as a curve with its curvature proportional to its arc length. So it is in a certain sense the equivalent to the logarithmic spiral whose curvature is inverse proportional to its length. Considering its definition we can easily find the tangent of the clothoid:

$$\left[\cos\frac{t^2}{2a^2}, \sin\frac{t^2}{2a^2}\right]$$

with a an arbitrary positive proportionality constant and t the arc length of the curve. Its curvature is given by

$$\kappa = \frac{t}{a^2}.$$

The curve is the defined integral of the tangent from 0 to length s. The resulting integral is a so called *Fresnel integral* which unfortunately cannot be calculated analytically. The utility file mentioned above contains two functions "FRESNEL_COS" and "FRESNEL_SIN" which evaluate these integrals numerically. Unfortunately the numerical procedure turns out to be unstable for greater values. There are also power series "FRESNEL_COS_SERIES" and "FRESNEL_SIN_SERIES" available. Both power series are formally correct but they have some not so nice properties like

a) they are not defined optimally,

b) they converge very slow for greater values and

c) they are very susceptible for rounding errors when calculating greater values. So it is necessary evaluating them in exact mode only which causes extra long calculation times very soon.

What to do? Although the tools given in the utility files are not very useful in my opinion, we can use *DERIVE* to calculate the integrals requested using another *DERIVE* feature. Luckily the Fresnel integrals can also be calculated applying the complex error function `ERF(z)`. `ERF` can be calculated in two ways: as power series and as continued fraction. The first one converges very fast for small values and the second one converges fast for large values. And … it is implemented as a standard function in *DERIVE*. This makes possible calculating Fresnel integrals and clothoids in real time with *DERIVE* without any extra programming efforts.

I am sending a short file containing functions for Fresnel integrals and for clothoids. Maybe that one could produce an updated `FresnelIntegrals.mth` file.

With best regards
Piotr Trebisz

Comment of the editor: I asked Piotr for possible applications of the Fresnel integrals (except the clothoid) and he answered:

Unfortunately I cannot offer additional examples at the moment. It was the treatment of the clothoid which led me to these integrals. I know that they are used in physics for calculation of wave diffraction, they are used in quantum mechanics and I found them in Richard Feyman's path integrals. And as you wrote in your mail clothoids are used in road construction for forming smooth connections between segments of different curvatures – and in particle accelerators.

The Fresnel functions are the integrals of $\cos(a\cdot t^2)$ and $\sin(a\cdot t^2)$. a is an arbitrary real number.

Let L the arc length of the clothoid and A its parameter (positive real number). Its curvature is proportional to $\frac{L}{A^2}$. Its asymptotic points are

$$\left(\frac{A\cdot\sqrt{\pi}}{2},\frac{A\cdot\sqrt{\pi}}{2}\right) \text{ and } \left(-\frac{A\cdot\sqrt{\pi}}{2},-\frac{A\cdot\sqrt{\pi}}{2}\right).$$

You can plot the clothoid expressions #5 or #8 of the following *DERIVE* file after introducing a slider bar for a in real time.

With my best regards

Piotr Trebisz

What follows is Piotr's DERIVE file. As I was very inspired by his mail and mth-file I did some research in my books and in the web. The list of my findings is given in the references at the end of this constribution, Josef

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

#2: --

#3: [a :∈ Real (0, ∞), s :∈ Real]

#4: --

#5: $$\mathrm{KLOTHOIDE}(a,\ s) := \left[\frac{\sqrt{\pi}\cdot a}{4}\cdot\left((1 + i)\cdot\mathrm{ERF}\left(\frac{s\cdot(1 - i)}{2\cdot a}\right) + (1 - i)\cdot\mathrm{ERF}\left(\frac{s\cdot(1 + i)}{2\cdot a}\right)\right),\ \frac{\sqrt{\pi}\cdot a}{4}\cdot\left((1 + i)\cdot\mathrm{ERF}\left(\frac{s\cdot(1 + i)}{2\cdot a}\right) + (1 - i)\cdot\mathrm{ERF}\left(\frac{s\cdot(1 - i)}{2\cdot a}\right)\right)\right]$$

#6: KLOTHOIDE_KOMPLEX(a, s) := KLOTHOIDE(a, s) • [1, −i]

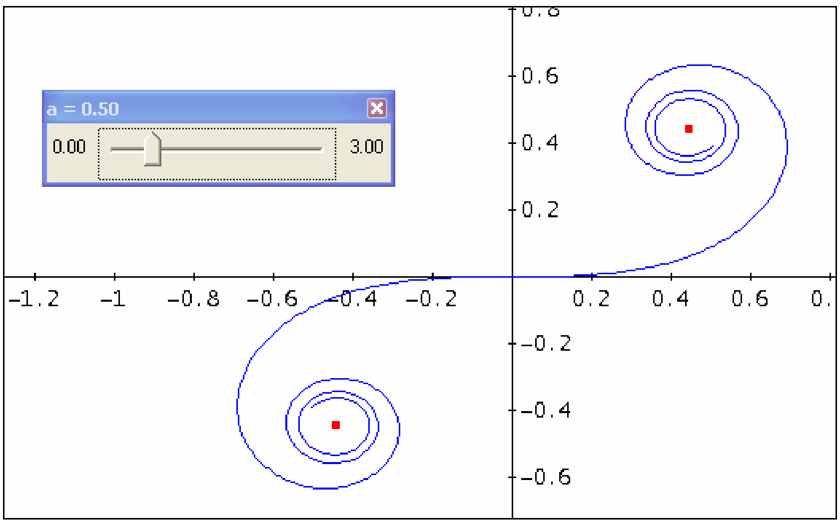
#7: $$\mathrm{KLOTHOIDE\_KOMPLEX}(a,\ s) := \left(\frac{\sqrt{\pi}\cdot a}{2} + \frac{\sqrt{\pi}\cdot i\cdot a}{2}\right)\cdot\mathrm{ERF}\left(\frac{s\cdot(1 - i)}{2\cdot a}\right)$$

#8: [RE(KLOTHOIDE_KOMPLEX(a, s)), IM(KLOTHOIDE_KOMPLEX(a, s))]

#9: $$\lim_{s\to\infty} [\mathrm{RE}(\mathrm{KLOTHOIDE\_KOMPLEX}(a,\ s)),\ \mathrm{IM}(\mathrm{KLOTHOIDE\_KOMPLEX}(a,\ s))] = \left[\frac{\sqrt{\pi}\cdot a}{2},\ \frac{\sqrt{\pi}\cdot a}{2}\right]$$

#10: $$\lim_{s\to-\infty} [\mathrm{RE}(\mathrm{KLOTHOIDE\_KOMPLEX}(a,\ s)),\ \mathrm{IM}(\mathrm{KLOTHOIDE\_KOMPLEX}(a,\ s))] = \left[-\frac{\sqrt{\pi}\cdot a}{2},\ -\frac{\sqrt{\pi}\cdot a}{2}\right]$$

#11: $$\begin{bmatrix} \frac{a\cdot\sqrt{\pi}}{2} & \frac{a\cdot\sqrt{\pi}}{2} \\ -\frac{a\cdot\sqrt{\pi}}{2} & -\frac{a\cdot\sqrt{\pi}}{2} \end{bmatrix}$$



#12: $$\frac{d}{ds}\ \mathrm{KLOTHOIDE}(a,\ s) = \left[\mathrm{COS}\left(\frac{s^2}{2\cdot a^2}\right),\ \mathrm{SIN}\left(\frac{s^2}{2\cdot a^2}\right)\right]$$

See now Piotr’s improved Fresnel integral functions together with their application defining the spiral in another way:

#14: $\text{FRESNEL\_C}(z) := \frac{1}{4}\cdot\left((1 + i)\cdot\text{ERF}\left(\frac{\sqrt{\pi}\cdot z\cdot(1 - i)}{2}\right) + (1 - i)\cdot\text{ERF}\left(\frac{\sqrt{\pi}\cdot z\cdot(1 + i)}{2}\right)\right)$

#15: $\text{FRESNEL\_S}(z) := \frac{1}{4}\cdot\left((1 + i)\cdot\text{ERF}\left(\frac{\sqrt{\pi}\cdot z\cdot(1 + i)}{2}\right) + (1 - i)\cdot\text{ERF}\left(\frac{\sqrt{\pi}\cdot z\cdot(1 - i)}{2}\right)\right)$

#16: $\text{FRESNEL}(z) := \text{FRESNEL\_C}(z) + i\cdot\text{FRESNEL\_S}(z)$

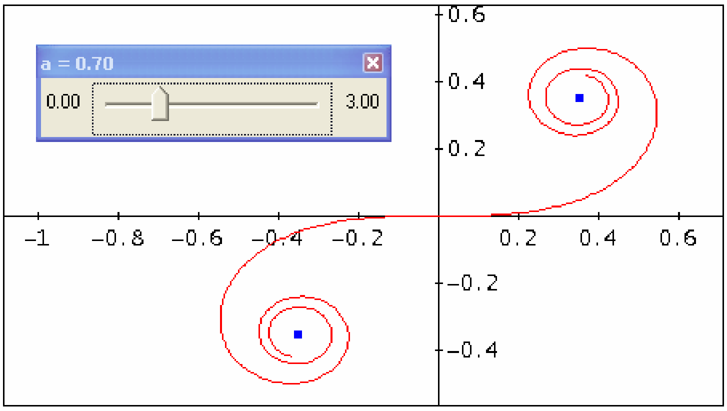
#17: $\text{FRESNEL}(z) := \left(\frac{1}{2} + \frac{i}{2}\right)\cdot\text{ERF}\left(\frac{\sqrt{\pi}\cdot z\cdot(1 - i)}{2}\right)$

#18: $a\cdot[\text{RE}(\text{FRESNEL}(z)),\ \text{IM}(\text{FRESNEL}(z))]$

#19: $\lim_{z\to\infty} a\cdot[\text{RE}(\text{FRESNEL}(z)),\ \text{IM}(\text{FRESNEL}(z))] = \left[\frac{a}{2}, \frac{a}{2}\right]$

#20: $\lim_{z\to-\infty} a\cdot[\text{RE}(\text{FRESNEL}(z)),\ \text{IM}(\text{FRESNEL}(z))] = \left[-\frac{a}{2}, -\frac{a}{2}\right]$

#21: $\begin{bmatrix} \frac{a}{2} & \frac{a}{2} \\ -\frac{a}{2} & -\frac{a}{2} \end{bmatrix}$



Original DERIVE Utility:

#24: $\text{FRESNEL\_SIN}(2) = \int_0^2 \text{SIN}\left(\frac{\pi\cdot t\_^2}{2}\right) dt\_$

#25: $\text{FRESNEL\_SIN}(2) = 0.3434156782$

0.14 sec calculation time

Piotr's Fresnel SIN Integral:

#26: $\text{FRESNEL\_S}(2) = \frac{\text{ERF}(\sqrt{\pi}\cdot(1 - i)) + \text{ERF}(\sqrt{\pi}\cdot(1 + i))}{4} + \frac{i\cdot(\text{ERF}(\sqrt{\pi}\cdot(1 + i)) - \text{ERF}(\sqrt{\pi}\cdot(1 - i)))}{4}$

#27: $\text{FRESNEL\_S}(2) = 0.3434156783$

0.03 sec calculation time

It is interesting comparing the velocity plotting the spiral when using the *DERIVE* Utility functions – FR(z) – and Piotr's functions – FRESNEL(z) with $-2\pi \le z \le 2\pi$.

#28: FR(z) := FRESNEL_COS(z) + ί·FRESNEL_SIN(z)

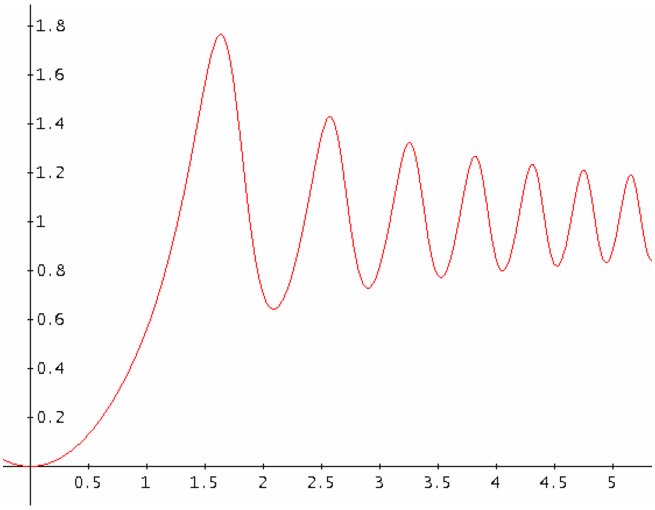
$$\#29:\quad FR(z) := \int_0^z \cos\left(\frac{\pi \cdot t\_^2}{2}\right) dt\_ + ί \cdot \int_0^z \sin\left(\frac{\pi \cdot t\_^2}{2}\right) dt\_$$

#30: [RE(FR(z)), IM(FR(z))]

#31: [RE(FRESNEL(z)), IM(FRESNEL(z))]

In Eric Weisstein's Encyclopedia I found the plot of the slope of the Cornu Spiral:

$$\#32:\quad \frac{FRESNEL\_S(z)}{FRESNEL\_C(z)}$$



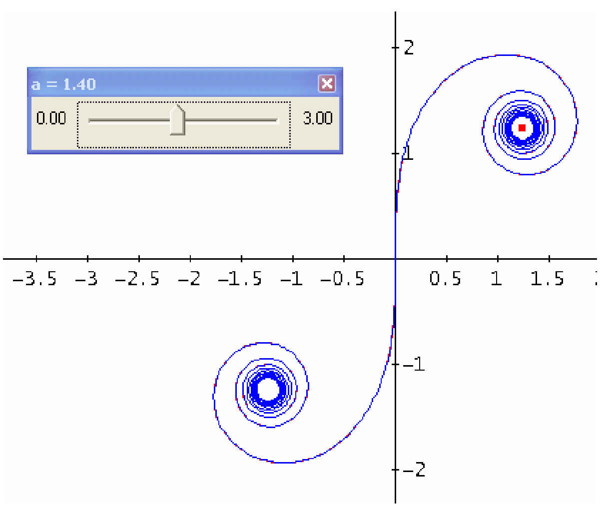
One of my favourite books dealing with differential geometry is Alfred Gray's *Modern Differential Geometry of Curves and Surfaces*.

Gray presents a generalization of this spiral with exchanged *x*- and *y*-coordinates. `cloth(1,a)` is the clothoid

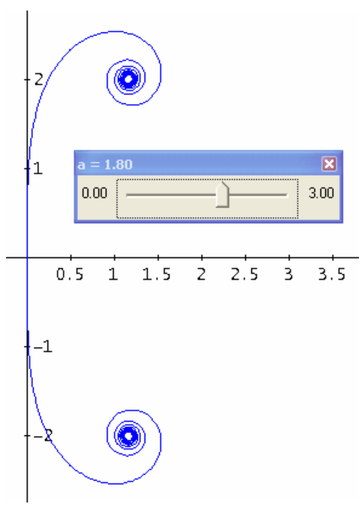
$$\#33:\quad cloth(n, a, s) := a \cdot \left[\int_0^s \sin\left(\frac{t^{n+1}}{n+1}\right) dt, \int_0^s \cos\left(\frac{t^{n+1}}{n+1}\right) dt\right]$$

$$\#34:\quad \lim_{s \to \infty} cloth(1, a) = \left[\frac{\sqrt{\pi} \cdot a}{2}, \frac{\sqrt{\pi} \cdot a}{2}\right]$$

#35: cloth(1, a)



#36: cloth(2, a)

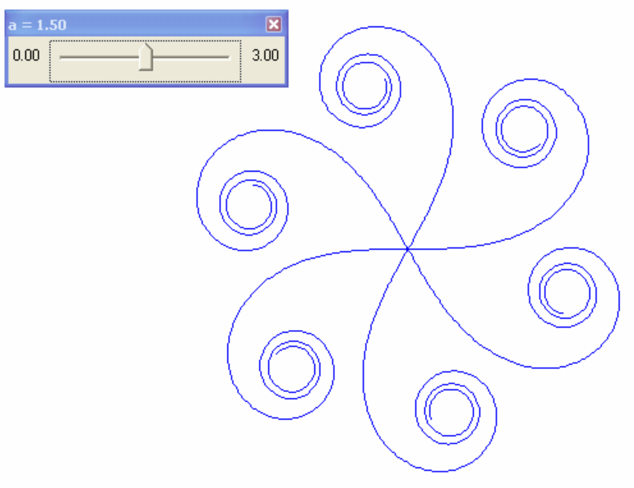


There is one chapter in Gray's book treating *Plotting plane curves with given curvature*. Gray uses *MATHEMATICA* and offers a special tool to find the natural or intrinsic equation[*] of curves solving a system of DEs.

I cannot transfer this tool directly to *DERIVE* but I can reproduce a nice *Clothoid Flower*:

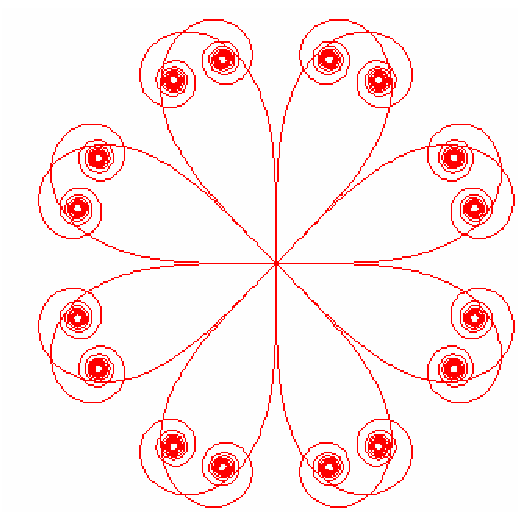
$$\#37:\quad \mathrm{cloth2}(n,\ a,\ s) := a\cdot\left[\int_0^s \mathrm{COS}\left(\frac{t^{n+1}}{n+1}\right)\,dt,\ \int_0^s \mathrm{SIN}\left(\frac{t^{n+1}}{n+1}\right)\,dt\right]$$

$$\#38:\quad \mathrm{VECTOR}\left(\mathrm{cloth2}(1,\ a)\cdot\begin{bmatrix}\mathrm{COS}(\alpha) & \mathrm{SIN}(\alpha)\\ -\ \mathrm{SIN}(\alpha) & \mathrm{COS}(\alpha)\end{bmatrix},\ \alpha,\ 0,\ \frac{2\cdot\pi}{3},\ \frac{\pi}{3}\right)$$



And here is another one (which is not shown in Gray's book):

$$\#39:\quad \mathrm{VECTOR}\left(\mathrm{cloth2}(2,\ 1.5)\cdot\begin{bmatrix}\mathrm{COS}(\alpha) & \mathrm{SIN}(\alpha)\\ -\ \mathrm{SIN}(\alpha) & \mathrm{COS}(\alpha)\end{bmatrix},\ \alpha,\ 0,\ 2\cdot\pi,\ \frac{\pi}{4}\right)$$



[*] Find more about the *Natural or Intrinsic Equation* in DNLs#29 and #30 (David Halprin's articles). See also the rich list of references at the end of this contribution, Josef

References

Books:

[1] Eric Weisstein, *CRC Concise Encyclopedia of Mathematics*, CRC Press 1999

[2] Abramowitz & Stegun, *Handbook of Mathematicsl Functions*, Dover

[3] Vladimir Rovenski, *Geometry of Curves and Surfaces with MAPLE*, Birkhäuser 2000

[4] Alfred Gray, *Modern Differential Geometry of Curves and Surfaces*, CRC Press 1994

[5] Alfred Gray, *Differentialgeometrie*, Spektrum Akademischer Verlag, 1994

[6] Georg Glaeser, *Geometrie und ihre Anwendungen in Kunst, Natur und Technik*, Spektrum 2005

Websites:

[7] Fresnel Integrals
http://functions.wolfram.com/GammaBetaErf/FresnelS/introductions/FresnelIntegrals/ShowAll.html

[8] How to evaluate Fresnel Integrals
http://www.thomasbeatty.com/MATH%20PAGES/ARCHIVES%20-%20NOTES/Complex%20Variables/How%20to%20evaluate%20Fresnel%20Integrals.pdf

[9] The Euler Spiral: A mathematical history
http://www.eecs.berkeley.edu/Pubs/TechRpts/2008/EECS-2008-111.pdf

[10] Eigenschaften ebener und räumlicher Kurven
http://page.mi.fu-berlin.de/sfroehli/ss2007/vorlesung02.pdf

[11] Natürliche Gleichung einer Kurve (Hans Walser)
http://jones.math.unibas.ch/~walser/Miniaturen/N/Natuerliche_Gleichung/Natuerliche_Gleichung.pdf

[12] Krümmung von Kurven und Flächen
http://www.soedernet.de/math/1samstage/05/vortrag.pdf

[13] Parameterisierung nach der Bogenlänge und Krümmung
http://www.blaesius-anne.de/arbeit_kurven/kurvenDb.html

[14] The elastica: a mathematical history
http://levien.com/phd/elastica_hist.pdf

[15] Formeln zur Klothoide
http://ww3.cad.de/foren/ubb/uploads/Clayton/Klothoide-Formeln.pdf

[16] Klothoide (GeoGebra-Applet)
http://www.projekt.didaktik.mathematik.uni-wuerzburg.de/klothoide/Klothoide.html

[17] Vorlesung Differrentialgeometrie (Matthias Bergner)
http://www.mathematik.uni-ulm.de/analysis/lehre/diffgeo_ss07/dg06.pdf

[18] David Halprin, Super Duper Osculants, DNLs #29, #30 and #32

[19] Ibrahim de Córdoba, Evaluation of Fresnel Integrals, DNL #39

[20] David Halprin, River Meander and Elastica, DNL #39

I was so much inspired by some of the references – especially by [4] and [11] that I tried to find an easy to use *DERIVE* implementation for plotting curves with given functions of their curvature dependent on their curve length. Josef

Natural (Intrinsic) Equations of Curves

Josef Böhm, Würmla, Austria

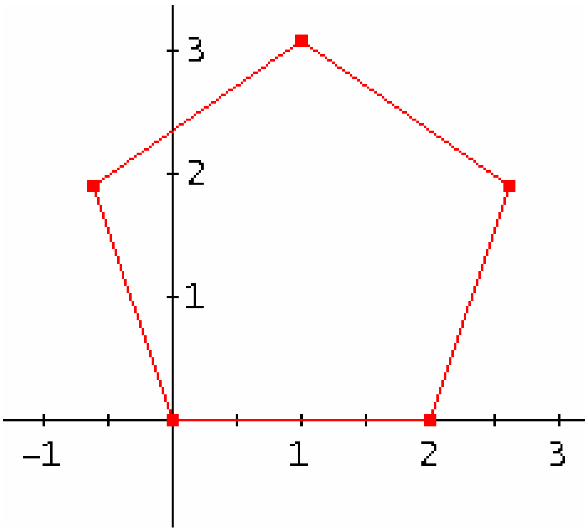
Hans Walser presents in his paper [11] a Turtle-Graph method using MuPad to find curves with given initial point, given start direction and a function κ(*s*) describing the change of direction of the tangent dependent on the arc length *s*. κ(*s*) is the curvature, its reciprocal value is the radius of the osculation circle.

Walser starts with an introductory example – plotting a pentagon:

```
      poly_turt(n, l, a, start, xo, yo, dir, xn, yn, pts, i) :=
        Prog
          xo := start↓1
          yo := start↓2
          dir := start↓3
          pts := [[xo, yo]]
          i := 1
          Loop
#1:         If i > n
               RETURN pts
            [xn := xo + l·COS(dir), yn := yo + l·SIN(dir)]
            pts := APPEND(pts, [[xn, yn]])
            dir := dir + a
            xo := xn
            yo := yn
            i :+ 1
```

$$\#2:\quad \text{poly\_turt}\left(5,\ 2,\ \frac{2\cdot\pi}{5},\ [0,\ 0,\ 0]\right)$$

We perform *n* moves (turning *n* times to the left by angle *a* and proceeding the distance *l* in the now new given direction. *start* gives the *x*- and *y*-coordinates of the starting point and as third component the initial direction. Repeating turning and proceeding 5 times results in a pentagon



We start at any position and move to position *s* where we have to change our direction according a function of *s*. Performing a numerical approximation we can do this only executing very small steps *ds*. It turns out to be something like a numerical approximation of a differential equation:

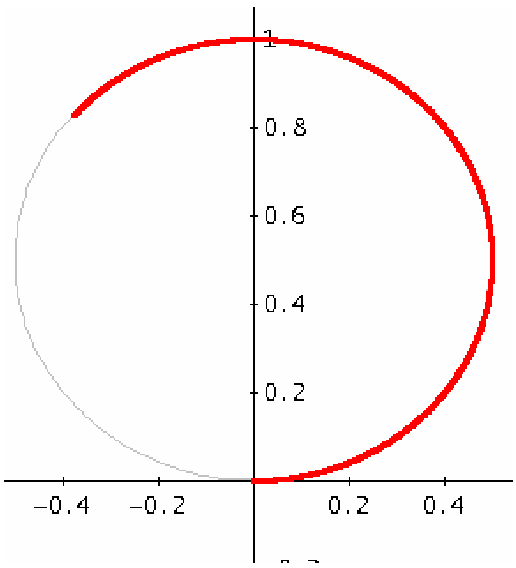
```
       nat_equ(κ, l, ds, start, n, xo, yo, dir, xn, yn, pts, i) :=
         Prog
           xo := start↓1
           yo := start↓2
           dir := start↓3
           pts := [[xo, yo]]
           n := l/ds
           i := 1
           Loop
#3:          If i > n
                RETURN pts
             xn := xo + ds·COS(dir)
             yn := yo + ds·SIN(dir)
             pts := APPEND(pts, [[xn, yn]])
             dir := dir + ds·SUBST(κ, s, i·ds)
             xo := xn
             yo := yn
             i :+ 1
```

κ(*s*) is the curvature function of the arc length, *l* is the total distance on the curve, *ds* is the step width and *start* are the initial values as explained above.

Take the first example: What is the curve with curvature = 2 in all of its points? I am quite sure that you know this:

```
#4:    nat_equ(2, 2, 0.001, [0, 0, 0])
```

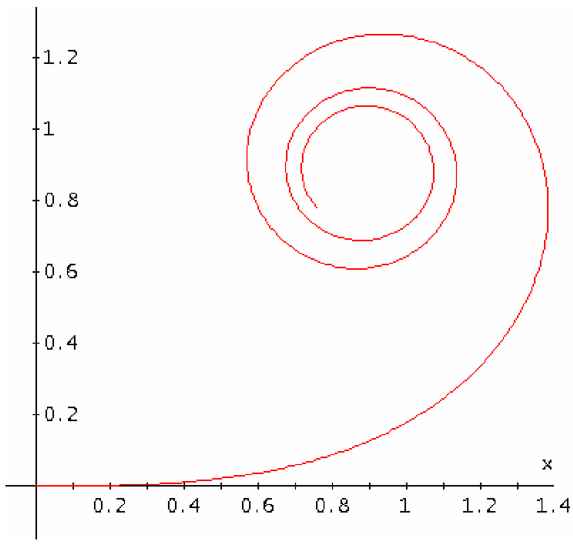
$$\#5: \quad x^2 + \left(y - \frac{1}{2}\right)^2 = \frac{1}{4}$$



We start at (0,0) looking in *x*-direction and change the direction after each step in the same constant way, i.e. by the value 2. This gives a circle with radius 0.5. You see the result of nat_equ and superimposed the respective circle (in grey colour). My approximation fits pretty well! I am satisfied.

We started with Piotr's Clothoid. The curvature is proportinal to the distance made:

```
#6:    nat_equ(s, 6, 0.01, [0, 0, 0])
```



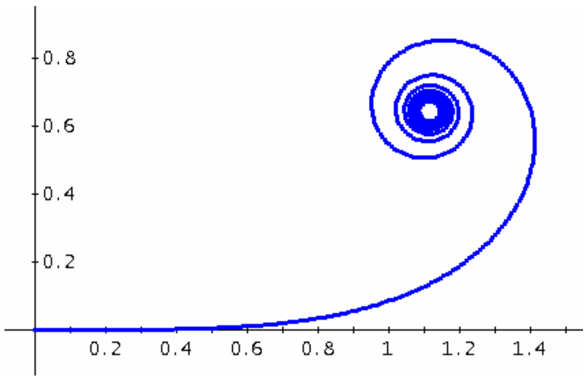
One could add any parameter *a* and then plot the curve like

```
nat_equ(0.5·s, 6, 0.01, [0, 0, 0])
```

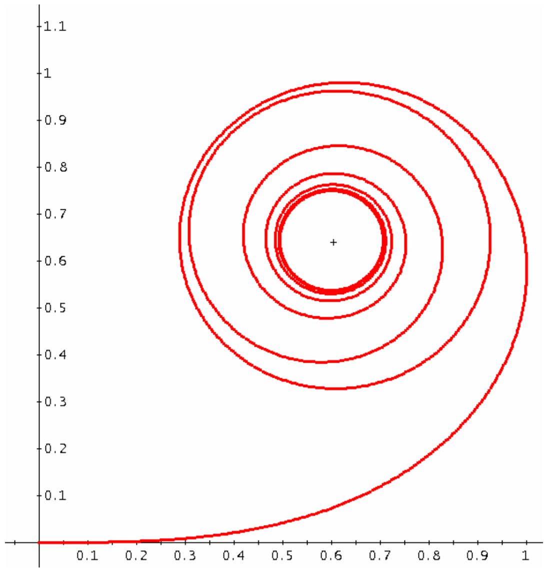
nat_equ() invites to reproduce some of Gray's examples and to experiment with own functions κ(*s*).

Our reward are very strange looking figures as can be seen on the next page.

$\text{nat\_equ}(s^2, 6, 0.01, [0, 0, 0])$

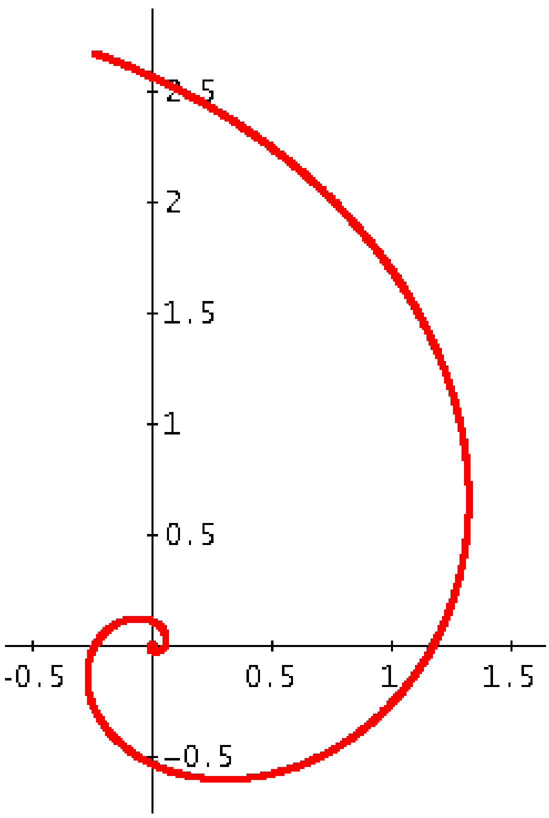


$\text{nat\_equ}(s + \text{SIN}(s), 10, 0.005, [0, 0, 0])$

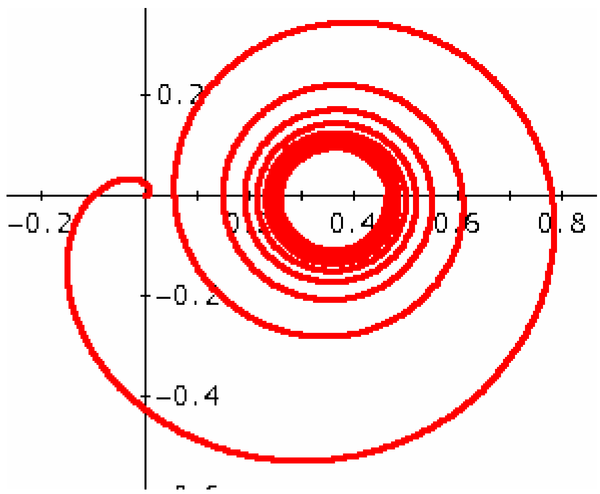


Do you remember Piotr's explanation that the Clothoid is the equivalent to the logarithmic spiral?

$\text{nat\_equ}\left(\frac{2}{s}, 6, 0.005, [0, 0, 0]\right)$

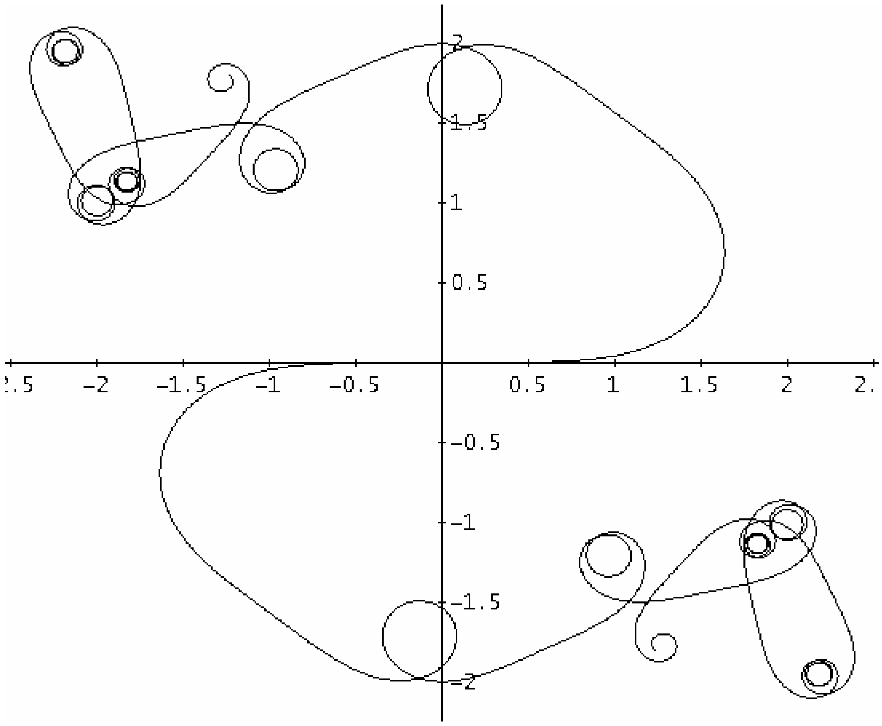


$\text{nat\_equ}\left(\frac{1}{s} + s, 10, 0.005, [0, 0, 0]\right)$



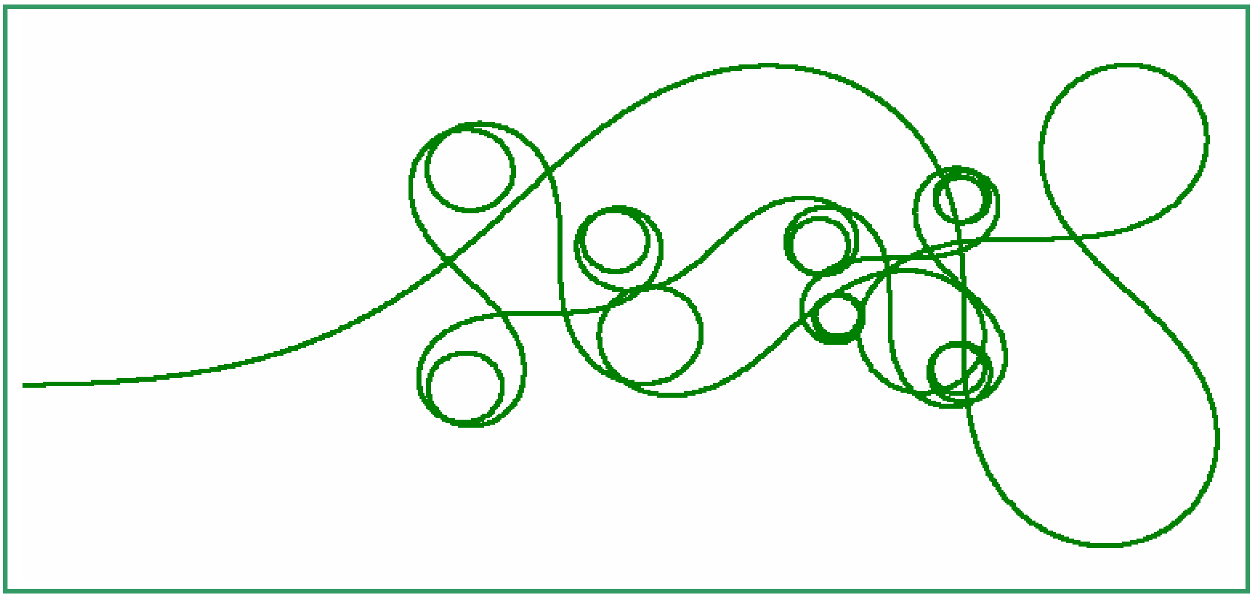
$$\text{nat\_equ}(s\cdot \text{SIN}(s)^2,\ 20,\ 0.01,\ [0,\ 0,\ 0])$$

$$\text{nat\_equ}(s\cdot \text{SIN}(s)^2,\ -20,\ -0.01,\ [0,\ 0,\ 0])$$



And this is my final natural equation curve:

```
nat_equ(2·s·COS(s)·SIN(s), 20, 0.005, [1, 1, 0])
```



Many thanks to Piotr Trebisz for his mail about improving the Fresnel Integrals, Josef.

Can we do this with TI-Nspire, too?

Natural Equations of Curves & TI-Nspire

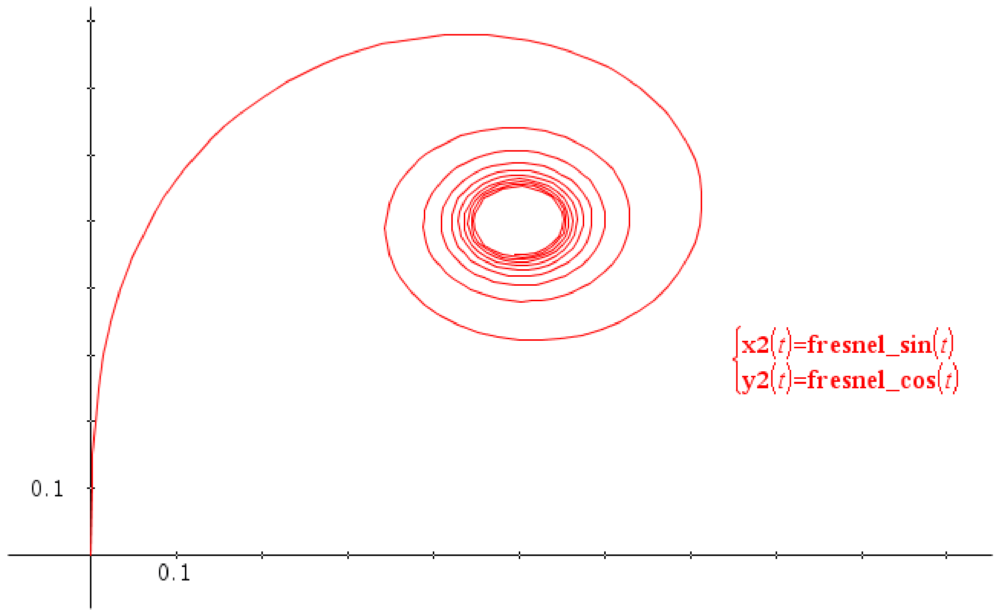
It is not too difficult to transfer the *DERIVE*-treatment on the TI-Nspire environment. What we cannot do is defining the Fresnel integrals supported by the error function because it is not implemented there.

$$fresnel\_sin(z):=\int_0^z \sin\left(\frac{\pi\cdot u^2}{2}\right)du:fresnel\_cos(z):=\int_0^z \cos\left(\frac{\pi\cdot u^2}{2}\right)du \qquad \text{Done}$$

$fresnel\_sin(2)$	0.3434156784
$cloth(t):=[fresnel\_sin(t) \quad fresnel\_cos(t)]$	Done
$cloth(10)$	$[0.4681699786 \quad 0.4998986942]$
$cloth(20)$	$[0.4840845359 \quad 0.4999873350]$
$cloth(50)$	$[0.4936338026 \quad 0.4999991894]$
$cloth(100)$	$[0.4968169005 \quad 0.4999998986]$

7/99

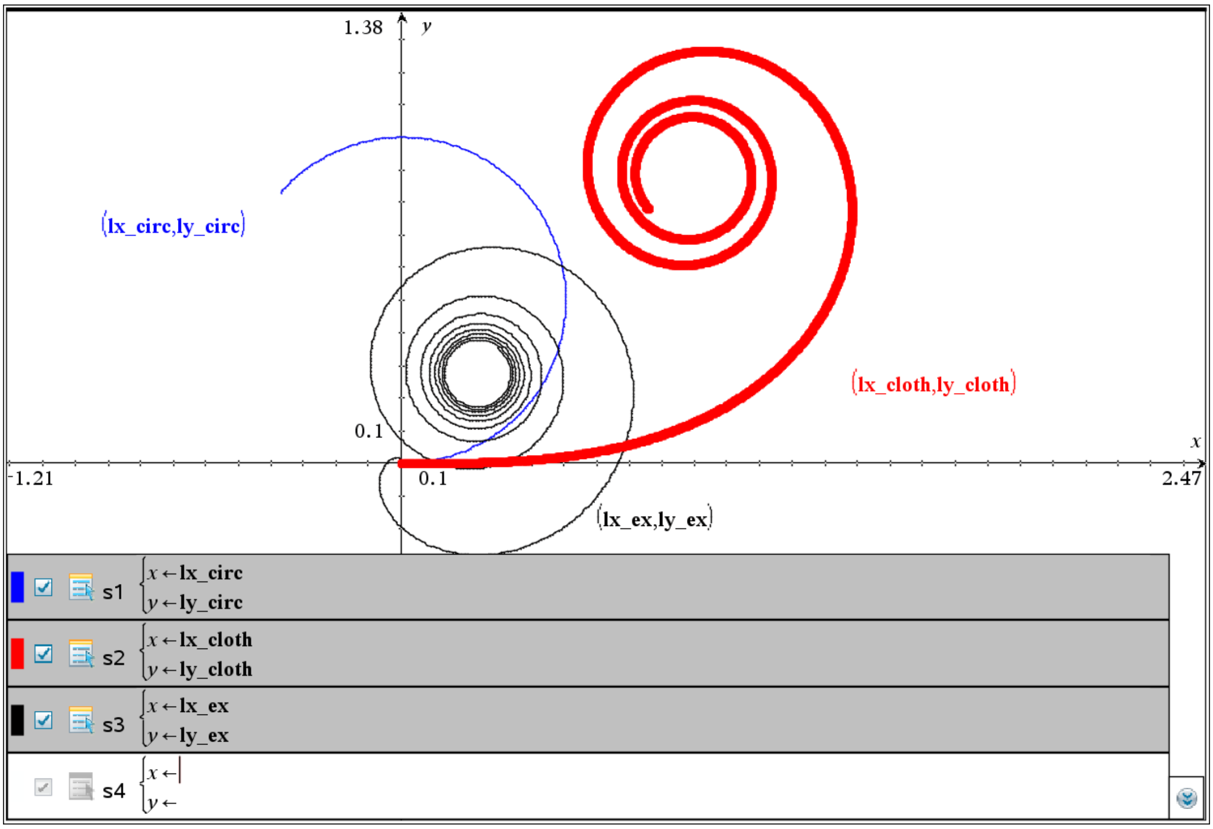
We cannot find the asymptotic center of the spiral by calculating the limit. But we see that the curve approaches (0.5, 0.5). Plotting the clothoid needs some time but it works – on the PC. I didn't try on the handheld.



The program to plot the natural or intrinsic equation is quite identical to the *DERIVE* procedure. The lists of the coordinates are stored as *lx* and *ly*. I stored them under new names in order to produce a plot of three curves on same axes. This works in a very reasonable time.

```
nat_equ(2,2,0.01,{0,0,0})
                         lx and ly are ready to be plotted!
                                                       Done
lx_circ:=lx
 [0  0.01  0.019998  0.02999  0.039972  0.04994  0.05▸
ly_circ:=ly
 [0  0.  0.0002  0.0006  0.0012  0.001999  0.002997 ▸
nat_equ(s,6,0.005,{0,0,0})
                         lx and ly are ready to be plotted!
                                                       Done
lx_cloth:=lx
 [0  0.005  0.01  0.015  0.02  0.025  0.03  0.035  0.0▸
ly_cloth:=ly
 [0  0.  1.25E-7  5.E-7  0.000001  0.000002  0.000004▸
nat_equ(1/s+s,10,0.01,{0,0,π/2})
                         lx and ly are ready to be plotted!
                                                       Done
lx_ex:=lx:ly_ex:=ly
 [0  0.01  0.015402  0.016107  0.013505  0.008593  0▸
                                                       8/99
```

```
nat_equ                                                1/13
Define nat_equ(kappa,l,ds,start)=
Prgm
Local n,xo,y0,dir,xn,yn,i
xo:={start[1]}:yo:={start[2]}:dir:=start[3]
lx:={xo}:ly:={yo}
n:=l/ds
For i,1,n
  xn:=xo+ds·cos(dir)
  yn:=yo+ds·sin(dir)
  lx:=augment(lx,{xn})
  ly:=augment(ly,{yn})
  dir:=dir+ds·  lim    (kappa)
              s→i·ds
  xo:=xn:yo:=yn
EndFor
Disp "lx and ly are ready to be plotted!"
EndPrgm
```



A Student's Problem with Homework Files

... My name is Michael Klamecki. My teacher, Prof. Haller recommended to contact you because of two nasty *DERIVE* problems which I came across just recently.

My first problem:
I am able to open *DERIVE* on my notebook only as administrator. Opening it just the „normal way" and trying to solve any equation it does not work at all: the window becomes blue and I am unable to perform any mouse click.

My second problem:
It happened when I loaded a previously saved *DERIVE* file. In a small pop-up window an error message appeared reporting that there is an error „parsing expression #7". After confirming the Ok-button another error was reported and finally the file was opened.

It would be great if you could give any advice.

with best regards
Michael

DNL:

Dear Michael,
What concerns your "administrator-problem", I asked Günter Schödl, the DERIVE-expert. This is his answer:

This is a problem of the administrator rights. I know this problem it appears sometimes when students install DERIVE on their laptops. My standard advice is:

- Uninstall DERIVE completely (including all remains)
- Install again (right click ... perform as administrator).

You can also try to change the rights in the DERIVE folder for "All Users".

If this does not work:
Look for another laptop where DERIVE works as you would like, remove DERIVE from your device and then copy it back (using an USB-stick) from the other laptop. This is how we could resolve this nasty problem.

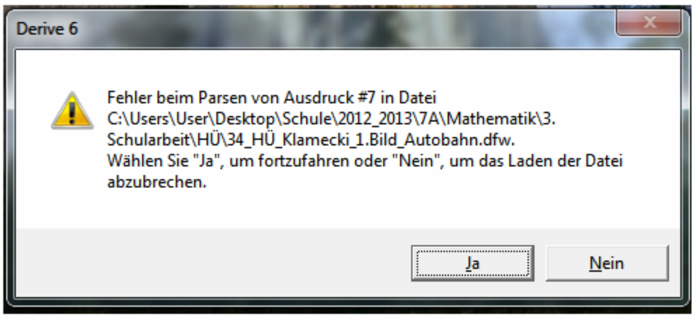
I don't know the reason for this problem. Probably it is connected with the security settings of some computers. I didn't investigate the reason, I have solved the problem in most cases.

Hope this helps, many thanks to Günter.
I cannot give any advice for your second problem (the parsing error) without knowing the respective file. So please send the file.
Regards
Josef

Michael came back to me:

... I uninstalled DERIVE manually and removed all respective files (except my homework files). Then I reinstalled DERIVE but the same problem occurred. Ok, I will work with *DERIVE* as administrator in the future.

I attach two screenshots and the *DERIVE* file causing the "parsing error" in expressions #7 and #8. I noticed that the fractions in #8 are printed in an unusual way and the AND-operator is written as AND (∧).




```
1) Der geradlinige Teil der Autobahn, in GeoGebra (PQ)

#3:   P := [0, 975]

#4:   Q := [1000, 0]

#5:   g(x) := k(x) + d

#6:   [g(0) = 975, g(1000) = 0]

#7:   SOLVE([g(0)=975,g(1000)=0],[d,k],Real)

#8:   [d=975 AND k=-39/40]
```

Best regards
Michael

DNL:

Dear Michael,

the error was easy to find – if one knows how to detect it.

Look at expression #5: You use variable k for the slope in the linear function g(x). This should read g(x) := k·x + d. But #5: g(x) := k(x) + d. Something must have happened.

Copy the "wrong" expressions #7 and #8 into the Entry Line, then you will notice that they are under quotes, i.e. they are not mathematical expressions but have changed to strings. This is the reason that the fractions are looking strange. See my screenshot below:

Why that? Go to expression #31! What have you done there?

```
"SOLVE([g(0)=975,g(1000)=0],[d,k],Real)"

#1:   CaseMode := Sensitive

#2:   InputMode := Word

1) Der geradlinige Teil der Autobahn, in GeoGebra (PQ)

#3:   P := [0, 975]

#4:   Q := [1000, 0]

#5:   g(x) := k(x) + d

#6:   [g(0) = 975, g(1000) = 0]

#7:   SOLVE([g(0)=975,g(1000)=0],[d,k],Real)
```

```
                    2
#31:  k(x) := a·x  + b·x + c

#32:  [k(402) = 626, k(571) = 695, k'(571) = h'(571)]

#33:  SOLVE([k(402) = 626, k(571) = 695, k'(571) = h'(571)], [a, b, c], Real)
```

You defined a quadratic function k(x). This is ok at this position and you will not face any problems saving the file.

Reloading this file *DERIVE* finds k(x) among the stored functions. Performing then all commands from the very beginning *DERIVE* substitutes k·x by k(x) which leads immediately to problems when solving the system in #7.

I renamed k(x) as k1(x), stored the file and everything works. So take care in the future naming variables and functions. I recommend using more-letters-function names, like curve(x), mo_way(x), ku1(x), ku2(x), etc.

I wish much success in mathematics and much fun with CAS.

Best regards to you and to my colleague Wilhelm Haller,
Josef

ITERATES again! (from Michel Beaudin)

Dear Josef, as you showed in the panel discussion in Estonia, the « iterates » function of Derive works well when exported to Nspire CAS (in fact, the first export was at the Bonn conference as you wrote in DNL). But it seems to be less general than the one Derive has.

For example, in Derive, iterates(diff(f,x), f, x^10, 3) returns [x^10, 10x^9, 90x^8, 720x^7]. In Nspire CAS, the SAME command gives [x^10, 0, 0, 0]. It seems that Nspire CAS computes diff(f, x) BEFORE replacing f by x^10. Do you have an explanation?

The reason for my question is the following : I have defined many functions in Nspire CAS and, in some cases, I was unable to use the iterates function. So I have used recursion which becomes slow. With iterates, it would be so fast! But I am not a good programmer compared to you. So, maybe you can help me!

Best regards and hope to see you and Noor soon,
Michel

Dear Michel,

this seems to be really difficult.

I tried several approaches. The problem is – as you mentioned – the immediate evaluation of the differential operator. When we call dif(f,x) Nspire evaluates to zero.

We see the bold “d” which cannot be transferred into the program as unevaluated operator.

I have a – very weak – solution writing the differential quotient into the program.

Maybe that experts like Philippe Fortin will provide a more general solution.

$newton\_its(f,v,v0,n):=iterates\left(v-\frac{f}{\frac{d}{dv}(f)},v,v0,n\right)$ Done

$newton\_its(t^2-2,t,0.2,5)$ $\{0.2,5.1,2.74608,1.73719,1.44424,1.41453\}$

$newton\_its(u^5-2000,u,2.,20)[21]$ 4.573050519

$2000^{0.2}$ 4.573050519

© The Tribonacci Sequence iterative

$trib\_seq(n):=(iterates(\{w[2],w[3],w[1]+w[2]+w[3]\},w,\{1,1,2\},n-1))^T[1]$ Done

$trib\_seq(15)$

$[1\ \ 1\ \ 2\ \ 4\ \ 7\ \ 13\ \ 24\ \ 44\ \ 81\ \ 149\ \ 274\ \ 504\ \ 927\ \ 1705\ \ 3136]$

$$newton(f,v,v0,n):=\begin{cases} v0, & n=0 \\ newton(f,v,v0,n-1)-\lim\limits_{v\to newton(f,v,v0,n-1)}\left(\frac{f}{\frac{d}{dv}(f)}\right), & n>0 \end{cases}$$ Done

$newton(x^5-2000,x,2.,20)$ 4.573050519

© The Hermite polynomials

$herms(n):=iterates(\{w[2],2\cdot x\cdot w[2]-2\cdot(w[3]+1)\cdot w[1],w[3]+1\},w,\{1,2\cdot x,0\},n)$ Done

$$herms(4)\quad \begin{bmatrix} 1 & 2\cdot x & 0 \\ 2\cdot x & 4\cdot x^2-2 & 1 \\ 4\cdot x^2-2 & 4\cdot x\cdot(2\cdot x^2-3) & 2 \\ 4\cdot x\cdot(2\cdot x^2-3) & 16\cdot x^4-48\cdot x^2+12 & 3 \\ 16\cdot x^4-48\cdot x^2+12 & 8\cdot x\cdot(4\cdot x^4-20\cdot x^2+15) & 4 \end{bmatrix}$$

1/14

iterates 6/16

```
Define iterates(vect,var,start,numb)=
Func
Local inc,in,iterlist,numvar,tvect
If left(string(var),1)="{" or left(string(start),1)="{": Goto vector
iterlist:={start}
For inc,1,numb
start:=vect|var=start
iterlist:=augment(iterlist,{start})
EndFor
Goto end
Lbl vector
iterlist:=start
For inc,1,numb
start:=vect|var=start
iterlist:=augment(iterlist,start)
EndFor
list▸mat(iterlist,dim(start))
Lbl end
```

iterates_f 5/7

```
Define LibPub iterates_f(f0,n)=
Func
Local inc,iterlist
iterlist:={f0}
For inc,1,n
f0:=d/dx(f0)
iterlist:=augment(iterlist,{f0})
EndFor
iterlist
EndFunc
```

My iterates_f does what you asked.

$iterates\_f(x^{10},4)$ $\{x^{10},10\cdot x^9,90\cdot x^8,720\cdot x^7,5040\cdot x^6\}$

$iterates\_f(\sin(2\cdot x),5)$

$\{\sin(2\cdot x),2\cdot\cos(2\cdot x),-4\cdot\sin(2\cdot x),-8\cdot\cos(2\cdot x),16\cdot\sin(2\cdot x),32\cdot\cos(2\cdot x)\}$

Regards as ever, Josef

From Michel

Thank you dear Josef. Your solution is not so weak and can be useful!

This example shows (again) how Derive was special... And this kind of example (and others) could make a good presentation for the Derive/Nspire session in Malaga. Is there any chance you attend ACA 2013? If not, let us promise to have a good beer together in Austria in 2014!

Michel

Christoph Küderli's problem with Error Trapping (TI-NspireCAS)

In a TI-NspireCAS program I tried to trap the error „Division by 0" by the construct Try...Else...EndTry. This does not work because the Else-block will not be executed.

I informed in the online manual and didn't find any advice and I had no success with a Google research. Trapping other errors – e.g. dimension error – is possible applying Try ... Else ... EndTry.

Can you please forward this deficiency to TI? I don't find any possibility to do this by myself.

```
prob0()
                    w= 1
                    s= 0
                    pmax= 17/23
                              Done
                              1/99
```

```
prob0                                        13/18
Define prob0()=
Prgm
Local p,p2,i,j,w,s
p:=0
p2:=0
For i,0,12
  For j,0,12
   If (i≠0 or j≠0) and (i≠12 or j≠12) Then
    p2:=(i/(i+j)+(12-i)/(24-i-j))/2
   EndIf
   If p2>p Then
    p:=p2
    w:=i
    s:=j
   EndIf
  EndFor
EndFor
Disp "w=",w
Disp "s=",s
Disp "pmax=",p
EndPrgm
```

See the program without (above) and with the – non working – error trapping routine (below).

```
prob()
     "Error: A test did not resolve to TRUE or FALSE"
                                            1/99
```

```
prob                                         20/20
Define prob()=
Prgm
Local p,p2,i,j,w,s
p:=0
For i,0,12
  For j,0,12
  Try
    p2:=(i/(i+j)+(12-i)/(24-i-j))/2
  Else
    p2:=1/4
    ClrErr
  EndTry
  If p2>p Then
    p:=p2
    w:=i
    s:=j
   EndIf
  EndFor
EndFor
Disp "w=",w
Disp "s=",s
Disp "pmax=",p
EndPrgm
```

I sent a mail to our TI-NspireCAS expert, Philippe Fortin:

Dear Philippe,
How are you? Here I am again with a Nspire-question (provided by a DUG-member):
Is it possible to catch a "Division by Zero" error using the Try - EndTry procedure?
So many errors can be handled by Try - EndTry. How to do with this one?

I attach the respective program.
It is no problem to circumvent the problem using an IF-contruction.

Best regards
Josef

This was Philippe's answer:

I hope everything is fine for you !

In fact, computations as 1/0 or 0/0 doesn't cause an error… we do get a result, equal to undef. There must be some reasons for this choice (but I am not sure of which ones !).

The question is then : is it possible to check that p2 contains the special value undef.

It is not possible to use

if p2=undef then…

If p2 is actually equal to undef, the result of this test will be undef = undef, which is not true or false…

It would be possible to use something like

If when(p2 = undef, false, false, true) then…

Another solution is to test the equivalent string value…

If string(p2) = "undef" then…

Last, since the expected result should be a number, it is possible here to test

If getType(p2) = "EXPR" then…

But this would not work if p2 could possibly contains values as sqrt(2) which also have the type EXPR…

None of these solutions is perfect (all are using odd constructions… and the last one is not correct for all situations). It would be better to have a function directly testing for a undef value.

Of course, a Try … EndTry including the test statement if p2>p then… would also be a possible approach…

Best regards,
Philippe

Dear Philippe,

many thanks for your extended answer.

Like you I tried many approaches … but not the last one. It works.

I will publish your answer + solution of the problem in the next DNL.

It would be interesting to learn why "Division by 0" is a "special error" which needs an extra treatment.

Best regards and thank you once more,

Josef

And here is Philippe's final comment:

Dear Joseph,

Just a thought concerning the use of undef… (which is not an Error). It could be related to **limit computation**…

When entering

lim(1/x,x,0)

it is rather a good point to get undef instead of an error message…

by the way, you probably noticed that

1/0 returns undef

1/0^2 returns infinity

This is also consistent with limit computation…

(this could be included in the answer you plan to publish !)

Best regards,

Philippe

An additional note on Brussels Gate sent by Erik van Lantschoot:

The Brussels Gate does not end fascinating me; at first I wanted to learn more about the Dutch engineers who built all three gates (Brussels, Mechelen and Gent – the last one has been destroyed) in 1822. In the meanwhile I found out that the head of the engineers was a certain Cornelis Alewyn. Then I wanted to lear from where C. A. has got his experience. Who knows history of technics knows that one has to do some research in France because the succeeders of Vauban started applying algebraic methods for calculation and planning military buildings. And I was lucky!! The inventor of the draw bridge with a counterweight on a curved track is Bernard Forest de Bélidor (1697-1761) who gave the equation of the track as a formula. Applications can be found in the United States (two double line road bridges) and in four or five French towns and in Switzerland. I will prepare a respective paper investigatig the difference between this curve and the parabola which I calculated in Dendermonde.

Best regards
EvL

Two good Turns

David Halprin, Australia

This was written in 1992 and made use of Mathcad 2.0. forgive this transgression.

There are many traps for the unwary when attempting to find a functional relationship between two variables so as to represent a plane curve in some coordinate system.

Since the Cartesian framework is the usual default coordinate system let us examine a common pitfall. Many curves have a symmetry about one axis, (at least), and if one requires the simplest equation to represent the curve, then perhaps one arc of the curve will be preferable, since the symmetrical arc will follow with ease.

A trivial example is the circle. The simplest equation in Cartesian Coordinates is for a semicircle, where we have y = f(x), whereas for the whole circle we have to combine it with y = -f(x) to have the expression [y - f(x)][y + f(x)] = 0 which gives us the usual quadratic form. One should note well that:-

1) Such a quadratic form allows 2 values for y for any value of x as it does also allow 2 values for x for any value of y.

$x^2 + y^2 = c^2$.

2) The circle is a composite curve, when expressed as y = f(x). If we had chosen a Polar Coordinate System then we would have a much simpler functional relationship between two variables and the circle is a simple curve of one arc, r = c

One can manipulate symbols in such a way that two different curves, or two arches, or branches, of related curves, (reflected, rotated and/or translated), can be combined in one equation to make it look like one curve, when it may be a simpler exercise, and much simpler equation, to analyse it as a composite curve. For instance any curve which lies above the X axis may be conjoined with its reflection beneath the X axis and the resultant equation may be encompassing the two parts as though they are two branches of the same curve, but the equation will be of a higher degree, and it is far easier to look at each branch separately.

In the problem of the Railway Transition Curve, one can fall into the trap of assuming that the entire curve is best represented:-

1) in Cartesian Coordinates and

2) in one equation.

However, staying with Cartesian Coordinates, and looking for only one of the symmetrical arcs, one would have obtained a far simpler solution, even more significant than in the case of the circle example.

One necessary quality of a point of inflection is that it has no curvature at the point itself, but on either side of that point the curvature takes opposite directions, meaning that a concave arc changes to a convex arc.

The force on the flange of the train's wheels has to be constant as it rounds the bend. The only way to state this with exactitude, is to talk about the relationship of the curvature to the arc length, namely:

A Railway Transition Curve, which is also used by road builders (road engineers), who want it to be such that the radius of curvature does not change discontinuously, since this would involve discontinuities in the centrifugal (centripetal) forces between the rails and wheel flanges of the trains, or the tyres and road for cars and trucks. So what is required is that the radius of curvature is a continuous function of the arc length.

The simplest and most natural functional relationship is that curvature (the reciprocal of the radius of curvature) is proportional to the arc length.

$$\kappa = \frac{1}{\rho} \propto s.$$

This equation uses two intrinsic variables, which are not to an external framework, as are the extrinsic Cartesian (x,y) or the extrinsic Polar (r,θ). There are many advantages to be gained from familiarity with the intrinsic variables, of which there are three, from which we may choose any pair.

Radius of Curvature, rho (ρ)

Arc length, (s)

Tangential Angle, phi (φ), where

$$\rho = \frac{ds}{d\varphi},\ \frac{dx}{ds} = \cos\varphi,\ \frac{dy}{ds} = \sin\varphi,\ \frac{dx}{d\varphi} = \rho\cdot\cos\varphi,\ \frac{dy}{d\varphi} = \rho\cdot\sin\varphi.$$

This curve has been studied for some time under various and has also proven to have been useful in optics. It is known as Klothoid, Clothoid, Cornu's Spiral, Euler's Spiral and Railway Transition Curve.

In addition to the use of a pair of variables in a functional relationship to define a plane curve, such an equation can be split into two equations by the use of a parameter, which is often referred to as a dummy variable. When one eliminates the parameter from the pair of equations, one is back to the original equation in two variables. However, if one looks for a geometrical image for the parameter, it is often an interesting finding. Commonly, when transforming coordinates from one system to another, one is left with two parametric equations and the parameter is one of the variables associated with the coordinate system, from which one is transforming.

e.g. Going from polar to Cartesian, the polar angle, θ is commonly the parameter, and it is a very convenient one, especially when instructing a plotting program to draw the curve, since one tells it to range θ from 0 to 2π and the curve is drawn with great facility.

However, there are serious pitfalls to acceptance without inspection of any and every such parameter, that may appear when transforming, as described, depending on the particular curve and the geometrical meaning of the variable, pertaining to that particular curve.

e.g. When transforming from Intrinsic to Cartesian Coordinates, there are s, φ and ρ. Usually there is no problem with s and φ, since with most curves one can range their values fairly well, but the radius of curvature may not range, and hence one may obtain, at most, a small arc of the curve and, at worst, an impossible representation of the original curve.

There are certain differential equations, involving elliptic integrals, called Fresnel Integrals, which have no closed form of solution, but which can be approximated with a strange looking series. These integrals can be used as parametric pair of equations to define the Clothoid.

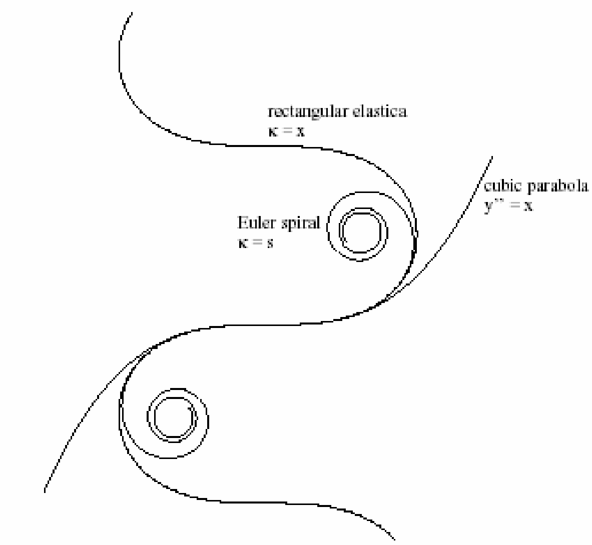
In fact, since it is the clothoid and can be specified very in the two variables, s (arclength) and ρ (radius of curvature), it is only a matter of numerical approximation, which will find the point of inflection. Further, the curve is compounded from two specially selected arcs of the clothoid, in the first and second quadrants respectively, and the omitted parts of the curve from the diagram are useless to railway engineers, since they are spirals about asymptotic points. The curve, as you will see below, is made up of two spirals, which are interconnected through a point of inflection, so the arc chosen is anywhere from the point of inflection up to as much of the spiral as it suits the engineers to use.

Family of Railway Transition Curves in Intrinsic Coordinates

1. $\dfrac{1}{\rho} = \kappa = \dfrac{d\varphi}{ds} = 2Ks$ Cesaro Intrinsic Equation
2. $\varphi = K \cdot s^2 + L$ Whewell Intrinsic Equation

   (Let $s = 0$ at $\varphi = 0 \;\therefore L = 0$)
3. $s = \sqrt{\dfrac{\varphi}{K}}$
4. $\rho = \dfrac{ds}{d\varphi} = \dfrac{1}{2\sqrt{K \cdot \varphi}}$ Euler Intrinsic Equation
5. $\dfrac{dx}{ds} = \cos\varphi = \cos(K \cdot s^2)\;\; and \;\; \dfrac{dy}{ds} = \sin\varphi = \sin(K \cdot s^2)$
6. $\dfrac{dx}{d\varphi} = \rho \cdot \cos\varphi = \dfrac{\cos\varphi}{2\sqrt{K}\sqrt{\varphi}} \;\; and \;\; \dfrac{dy}{ds} = \rho \cdot \sin\varphi = \dfrac{\sin\varphi}{2\sqrt{K}\sqrt{\varphi}}$

Here is a challenge for the DUG-Members:

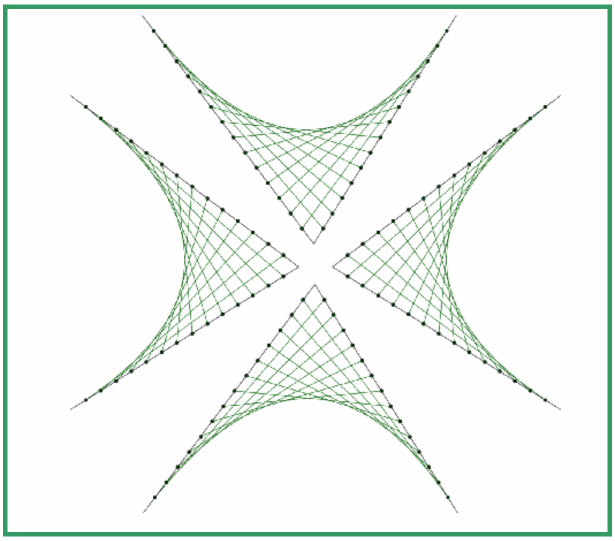


In Levies paper ([14] on page 22) I found this pretty figure presenting osculating cubic, clothoid and elastica. Can you reproduce this figure? The parameter form of the elastica is given below, Josef.

See: *David Halprin, River Meander and Elastica* in DNL#39.

$$\left[\int_0^s \mathrm{COS}(a \cdot \mathrm{SIN}(b \cdot s))\, ds,\; \int_0^s \mathrm{SIN}(a \cdot \mathrm{SIN}(b \cdot s))\, ds\right]$$

Christine Kova's Students:

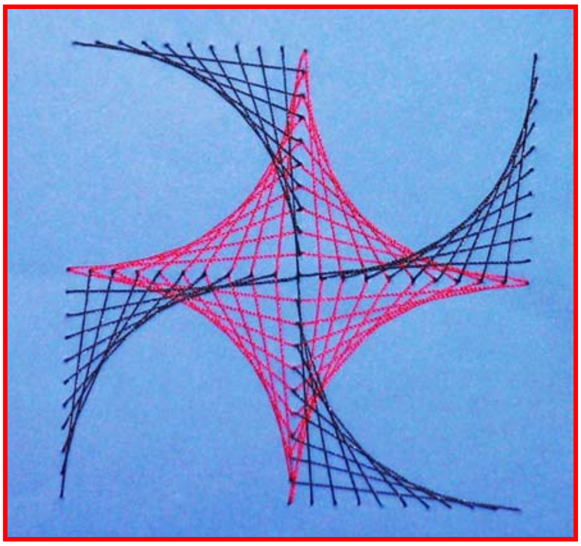


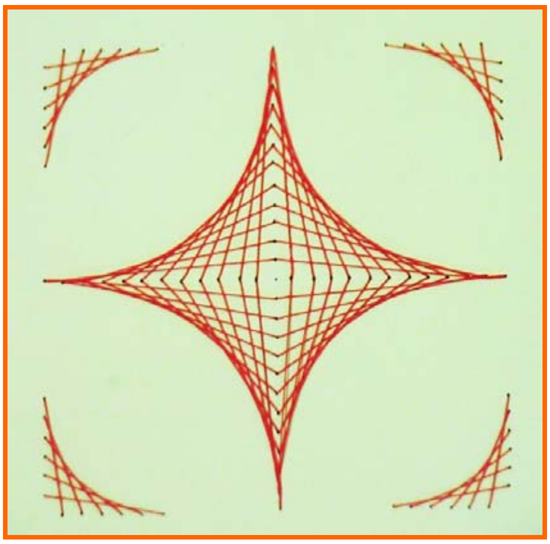
This is one of the examples we produced in a workshop (DERIVE & GeoGebra) several years ago.

One can underlie background pictures of ready made stitching (scans from books, pictures from the web, or own designs).

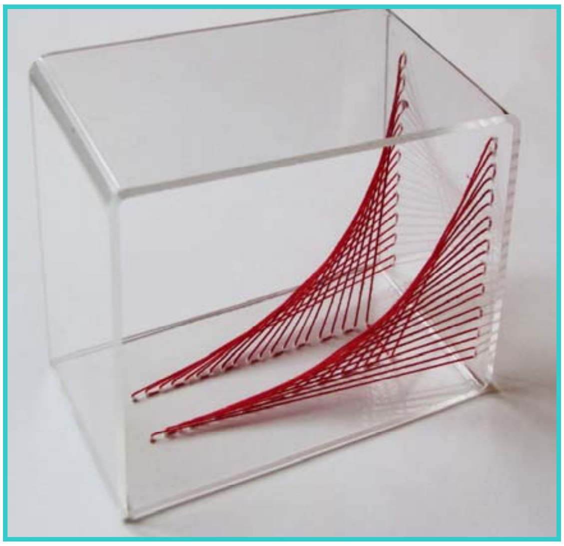
Students learn to apply parameter form and families of segments and arcs.

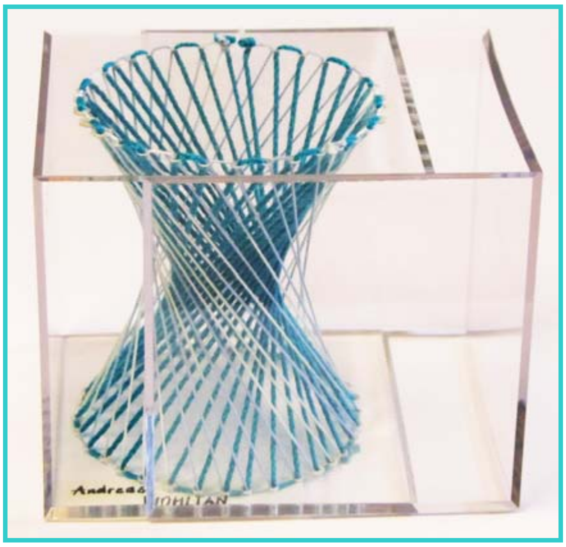
Student's works:





3D-patterns fixed in acryl cubes:





This is a more complex pattern. I took it as background picture in DERIVE and you can see left below the first step modelling the figure. Josef