THE DERIVE - NEWSLETTER #24

ISSN 1990-7079

THE BULLETIN OF THE

DERIVE

USER GROUP

+ TI 92

Contents:

1 Letter of the Editor

2 Editorial - Preview

3 *DERIVE* User Forum (including ROMBERG.MTH and Comments)

10 Some REVIEWS

Josef Böhm
18 *DERIVE* – ACD – *ACROSPIN*

R. Contreras and F. Gutiérrez
35 3rd Order DIFFERENTIAL EQUATIONS

Thomas Weth
40 Lexicon of Curves (9) – Snail of Pascal

44 AC DC 3 by Alfonso Población

Peter Mitic
45 Probability Distributions: Proof and Computations (1)

Johann Wiesenbauer
48 Titbits (9)

53 The TI-92 Corner

W.Pröpper, SOLSYST for systems of linear equations

E.Laughbaum, Sums of Absolute Value Functions

T.Etchells, My favourite DERIVE functions for the TI-92

revised version 2010 December 1996

[1] **Mathematikunterricht mit Computeralgebra-Systemen,** H. Heugl, W. Klinger und J. Lechner
307 Seiten, DM 59,90; öS 443.00; sFr 48,00; Addison-Wesley, 1996, ISBN 3 8273 1082 2
Ein didaktisches Lehrerbuch mit Erfahrungen aus dem österreichischen DERIVE-Projekt. Eine Kurzbesprechung folgt im nächsten DNL.

[2] **TI 92 - du lycée à la prépa,** Henri Lemberg
DUNOD / Texas Instruments France, Paris Mai 1996, ISBN 2 10 003039-6
This book recapitulates on 314 pages the mathematics curriculum of French gymnasiums and shows up all the concepts and algorithms which a French student needs to know for the entrance examination for French universities.

[3] **TI 92 - les programmes!,** Jean-Michel Ferrard
DUNOD / Texas Instruments France, Paris Mai 1996, ISBN 2 10 003104-X
This is a real treasury of 480 pages full with TI 92 functions and programs. All the programs can be used for their own but they also can be assembled within a library divided in different folders containing their own menus which allow an easy approach to the programs. The sections covered are: Arithmetique et Trigonometrie, Polynomes, Matrices, Geometrie, Developpements Limites, Analyse, Geometrie Differentielle, Fonctions Speciales, Probabilites. Even if you only have a very poor knowledge of French - like me - it is easy to follow. Fortunately the language of maths is international. If you want to learn and to train programming with the TI 92 then I´d strongly recommend this book.

[4] **TI 92 - le "top" des jeux!,** Vincent Bastid et Emmanuel Neuville, DUNOD / Texas Instruments France, Paris Mai 1996, ISBN 2 10 003040-X
Transformez votre calculatrice TI 92 en console de jeux!
A collection of games containing TI-tris, Démineur (Minesweeper), Bataille-Navale (for two TIs!), TI-Invaders, Tic-Tac-Toe-3D and four more games.

I have to thank Mathias Makowsky from Marbach, Germany, who faxed the titles of the three French books. He saw them in a bookstore in Brittany during his holidays. Many thanks, the books you recommended are very useful. Josef

[5] **An Introduction to the Mathematics of Biology (With Computer Algebra Models)**
Yeagers, Shonkwiler and Herod, Birkhäuser Boston 1996, ISBN 0 8176 3809-1
You can find a short review on page 17.

[6] **AGNESI to ZENO, Over 100 Vignettes from the History of Math,** Sanderson Smith
Key Curriculum Press, Berkeley 1996, ISBN 1 55953 107 X
This is not a Computer Algebra Book, but it is a wonderful book to motivate teachers and students as well for investigations and projects and presents a lot of facts concerning history of mathematics and the men and women who wrote this history. The book is available from Jan Vermeylen´s Rhombus Shop. (See the address on page 34).

Call for partners

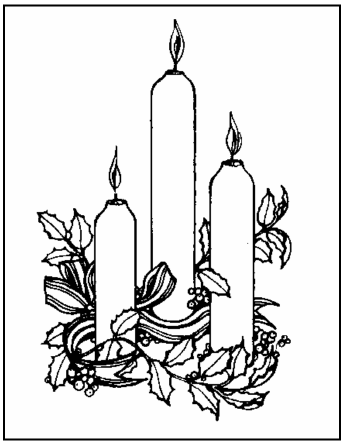
At the Information Day about the European Union's Educational Programs I was asked eventually to start a transnational project in the frame of the **COMENIUS Program**. So I'd like to ask for partners who could imagine cooperating. We have to be at least three participants from different European Union countries. I have two ideas in mind (but maybe there are better ones):

- Set up a transnational structure for math teacher's training in modern technologies.
- Exchange, evaluate, improve and customize teaching materials for teaching maths using modern technologies.

If you are interested, then please contact me as soon as possible. Please notice my new email address: Josef_Boehm@bboard.at *(is not valid since a couple of years!)*

Liebe DUG Mitglieder,

Einige recht arbeitsreiche Tage und Nächte liegen hinter mir. Aber es ist wieder gelungen: der letzte DNL des Jahres 1996 ist fertig. Während er nun zum Drucken geht, wird noch rasch die Diskette des Jahres 96 randvoll gepackt und überprüft, ob auch alle Dateien und „Weihnachtsgeschenke“ drauf sind. Dann werden meine Frau Noor und dieses Mal auch meine Tochter Astrid einige Hundert Kopien ziehen, die Newsletter mit Diskette, 3D-Brille und Renewal Form in ein großen Kuvert stecken, überall die notwendigen Stempel anbringen und sie dann auf die weite Reise schicken.



Bitte beachten Sie meine neue email-Adresse am Ende dieser Seite. Endlich habe ich meinen eigenen Internetzugang an der Schule. Dass dieses Medium bereits genützt wird, zeigt das reiche User Forum. Es bietet sich auch ein neues Service der DUG an: Falls Sie den einen oder anderen Artikel aus einem DNL - auch von früheren - als Textfile brauchen könnten, kann ich Ihnen diesen gerne über email schicken. Heute habe ich wieder in meinem elektronischen Postkasten gefischt und hatte Anglerglück: es gibt eine Antwort von Al Rich auf das Matrizenproblem im User Forum und Terence Etchells machte eine aufregende Ankündigung für einen möglichen Beitrag im nächsten Jahr.

Ich möchte Sie auch nochmals auf meine Partnersuche für ein EU-Projekt im Rahmen des COMENIUS Programms auf der Infoseite aufmerksam machen. Ich konnte heuer zwei Klassen mit TI 92 ausstatten, daher wäre ich an einem Austausch an Unterrichtsmaterialien für den TI 92 - aber nach wie vor auch für DERIVE - sehr interessiert. Ich arbeite heuer besonders im Precalculus Bereich

Meinen letzten Brief dieses Jahres möchte ich aber nicht beenden, ohne Ihnen allen für die engagierte Mitarbeit zu danke. Ohne diese gäbe es längst keinen DNL mehr. Noch wertvoller aber finden wir die herzliche Freundschaft, die die DERIVIANER weltumspannend untereinander verbindet. Jedes Zusammentreffen, jede Konferenz ist ein deutliches Zeichen dafür. Viele persönliche Begegnungen und Freundschaften haben sicher nicht nur unser (Noor und Josef) Leben, sondern das von vielen unter uns bereichert.

In diesem Sinne wünschen ich allen ein frohes Weihnachtsfest und ein gesundes, glückliches und erfolgreiches Jahr 1997 - Jahr 1 nach DERIVE for Windows.

Ich hoffe Euch alle im nächsten Jahr wieder begrüßen zu dürfen.

Josef

(Bitte begleichen Sie Ihren Mitgliedsbeitrag 1997)

Dear DUG Members,

Some very busy days and nights are lying behind me. But it has lucked again: the last DNL of 96 is ready. While it is going to be printed, the Diskette of the Year has to be filled brimful and checked if all the files and the Christmas gifts are on it. Then my wife Noor and my daughter Astrid will produce several hundreds of copies, put them together with the DNL, the 3D-spectacles and the renewal form in big envelops, affix the stamps and address labels and send them on their big journeys.

Please note my new email address at the end of the page. I am happy to have my own email access now at my school. And I like to use this media, as you can see in the rich User Forum. So I´d like to offer another DUG service: if you would like to have the text file of any DNL article - even from earlier DNLs - then I could email it to you. Today I was fishing again in my electronic mailbox, and I was lucky. Among others I found Al Rich´s answer to the matrix problem in the User Forum and my friend Terence Etchells made an exciting announcement for a possible contribution for 1997.

I want to focus your attention once more on my call for partners for an EU-project within the COMENIUS program on the Information page. I was able to equip two of my classes with TI 92s and so I am very interested to exchange teaching materials for the TI - but also for DERIVE. In these courses I deal mainly with precalculus stuff.

I don´t want to finish my last letter of 96 without thanking you all for your enthusiastic cooperation. Without this there would be no DNL any longer. But we find much more valuable the warm hearted friendship which connects the DERIVIANS from all over the world. Each meeting, each conference is a witness of this fact. Many personal friendships and meetings have not only enriched our lives (Josef and Noor) but - I am sure - those of many of us.

In that sense I wish you all a Merry Christmas and a healthy, happy and successful 1997 - year 1 after Derive for Windows. Hope to meet you all next year again!

Sincerely yours

Josef

(Please settle your membership fee for 1997)

email: nojo.boehm@pgv.at

The *DERIVE-NEWSLETTER* is the Bulletin of the *DERIVE User Group*. It is published at least four times a year with a contents of 40 pages minimum. The goals of the *DNL* are to enable the exchange of experience made with *DERIVE* as well as to create a group to discuss the possibilities of new methodical and didactical manners in teaching mathematics.
We include now a section dealing with the use of the TI-92.

Editor: Mag. Josef Böhm.
A-3042 Würmla
D´Lust 1
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 *D-N-L*. The more contributions you will send, the more lively and richer in contents the *DERIVE Newsletter* will be.

Preview: Contributions for the next issues

LOGO in DERIVE, Lechner & Roanes Lozano, AUT & ESP
3D-Geometry, Reichel, AUT
Algebra at A-Level, Goldstein, UK
Graphic Integration, Linear Programming, Various Projections, Böhm, AUT
Tilgung fremderregter Schwingungen, Klingen, GER
A Utility file for complex dynamic systems, Lechner, AUT
Examples for Statistics, Roeloffs, NL
Linear Mappings and Computer Graphics, Kümmel, GER
Solving Word problems (Textaufgaben) with DERIVE, Böhm, AUT
Line Searching with DERIVE, Collie, UK
About the "Cesaro Glove-Osculant", Halprin, AUS
Tangrams with DERIVE, Población, ESP
Hidden Lines, Weller, GER
Fractals and other Graphics, Koth, AUS
Experimenting with GRAM-SCHMIDT, Schonefeld, USA

The TI-92 Section, Waits a.o.
and
Setif, FRA; Vermeylen, Belgium; Leinbach, USA; Halprin, AUS; Biryukow, RUS; Weth, GER; Wiesenbauer, AUT; Keunecke, GER; Aue, GER;
Stahl, USA; Mitic, UK; Sirota, RUS;and ....

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

Terence Etchells, Liverpool, UK

As you may be aware I have a new job lecturing in mathematics at the John Moores University. I am sending an attached file called ROMBERG.MTH. This is a file that I recently wrote to perform Romberg integration and produce the table of successive Romberg approximations.

ROMBERG(f,x,a,b,n) estimates INT(f,x,a,b,n) via Romberg's method starting with Simpson's rule estimates for 2, 4, 8, ... 2^n strips.

ROMBERG_TABLE(f,x,a,b,n) produces a table of the successive Romberg approximations computed by ROMBERG().

You may be interested to know that the function ROMBERG() is significantly faster than classic DERIVE's internal approximation on certain integrals.

Try INT(SIN(x)/x, x, 0, 4) with precision set to 10 digits and ROMBERG(SIN(x)/x, x, 0, 4, 5). Something funny happens in DERIVEXM 3.10 as the functions take longer to evaluate.

Best wishes, Terence

```
"This file performs Romberg integration on the integral INT(f,x,a,b,n)"

M(f,x,a,b,n):=SUM((b-a)/n*LIM(f,x,a+(2*r-1)*(b-a)/(2*n)),r,1,n)

T(f,x,a,b,n):=(b-a)/(2*n)*(LIM(f,x,a)+2*SUM(LIM(f,x,a+r*(b-a)/n),r,1,
              n-1)+LIM(f,x,b))

S(f,x,a,b,n):=(2*M(f,x,a,b,n)+T(f,x,a,b,n))/3

ROMBERG_START(f,x,a,b,n):=VECTOR(S(f,x,a,b,2^r),r,1,n)

ROMBERG_AUX(f,x,a,b,n):=ITERATES([VECTOR((2^k*v SUB (r+1)-v SUB r)/
           (2^k-1),r,1,n-c),k+2,c+1],[v,k,c],[ROMBERG_START(f,x,a,b,n),4,1],n-1)

ROMBERG_ADD(v,n):=VECTOR(APPEND(v SUB c,VECTOR("",r,1,c-1)),c,1,n)

ROMBERG_EXTRACT(v,n):=v SUB n SUB 1 SUB 1

ROMBERG(f,x,a,b,n):=ROMBERG_EXTRACT(ROMBERG_AUX(f,x,a,b,n),n)

ROMBERG_EXTRACT_COLUMN(v,n):=VECTOR(v SUB r SUB 1,r,1,n)

ROMBERG_AUX_TABLE(v,f,x,a,b,n):=ITERATES([VECTOR(2^k*v SUB (r+1)*v SUB r/
            (2^k-1),r,1,n-c),k+2,c+1],[v,k,c],[ROMBERG_START(f,x,a,b,n),4,1],n-1)

ROMBERG_TABLE(f,x,a,b,n):=ROMBERG_ADD(ROMBERG_EXTRACT_COLUMN(ROMBERG_AUX_TABLE(v,
                          f,x,a,b,n),n),n)`
```

#14: Precision := Approximate — User

#15: $\int_0^4 \frac{\text{SIN}(x)}{x}\,dx = 1.75820$ — User=Simp(User)

#16: $\text{ROMBERG}\left[\frac{\text{SIN}(x)}{x}, x, 0, 4, 5\right] = 1.75820$ — User=Simp(User)

#17: $\text{ROMBERG\_TABLE}\left[\frac{\text{SIN}(x)}{x}, x, 0, 4, 5\right]$ — User

#18: — Simp(#17)

1.75804	3.29705	11.0441	122.461	15013.0
1.75819	3.29734	11.0451	122.473	""
1.75820	3.29736	11.0451	""	""
1.75820	3.29736	""	""	""
1.75820	""	""	""	""

DNL: *Some days later there was another message from Terence which might be interesting for pure* ***DERIVIANS*** *and* ***TI-Users*** *as well:*

How are you? It was nice to meet up with you again in Bonn. (*I enjoyed our talks, too*)
Whilst discussing the TI 92 with David Stoutemyer at Bonn I was bemoaning the fact that the Ti did not have my favourite function ITERATES. He said "no problem, we'll write one": Dave then gave me a brief tutorial on writing functions in the TI 92 function programming language. Time was short so he scribbled a few ideas on paper. On the trip back from Bonn and over Summer I set to write a series of *DERIVE* functions for the TI 92, such as: ITERATES(), ITERATE(), ELEMENT(), DEL_ELEM(), (This is DELETE_ELEMENT(), but the TI 92 restricts function names to 8 characters), SWP_ELEM(), REV_ELEM(), RHS(), LHS(). Also, *DERIVE* will easily plot a $2 \times n$ matrix in a 2D-plot window. I could not find an easy way to do this with the TI 92 so I wrote a program PLOT_MAT that plots a $2 \times n$ matrix.

I will, very shortly, be putting these functions on my Web page; see the signature file below for the URL if you should wish to browse it (there are lots of my files to down load as well).

e mail: t.a.etchells@livjm.ac.uk
web page: http://www.cms.livjm.ac.uk/www/homepage/cmstetch/index.htm

DNL: *It would take a lot of space to print all the functions. You will find them on the Diskette of the Year in subdirectory TI, accompanied by W.Pröpper´s DIRA package for investigating functions and his SOLSYST-function to solve simultaneous equations in a DERIVE like way.That are the true DERIVIANS, who implement their favourite DERIVE functions on the TI-92. (See more in the TI-92 Corner and on page 43 – a demo of PLOT_MAT.)*

Johann Wiesenbauer, Vienna, Austria

In the following the polynomial system of equations:
(cf. The International DERIVE Journal, Vol.3, No.2,p.96)
will be solved by means of my routines (cf. DNL#23, Titbits(8)).

$$\begin{aligned} x\,y &= 1-z \\ x\,z &= 1-y \\ y\,z &= 1-x \end{aligned}$$

(Johann refers to an article dealing with the implementation of Groebner bases in DERIVE. Josef) Preload RED(u,v) and SOLVE2(u,v,x,y) from TITBITS8.MTH

First we prove that there doesn't exist a solution of the system above where x, y, z are all different.
The following equation

```
                                                       2
#1:  RED(x·z - 1 + y, y·z - 1 + x) = y·(1 - z ) + z - 1
```

shows that either $z = 1$ or $y = 1/(z + 1)$. For reasons of symmetry $z = 1$ or $x = 1/(z+1)$ holds as well. Therefore our assertion is certainly true for $z \neq 1$. But for $z = 1$ the system above simplifies to $x\,y = 0$, $x + y = 1$, which leads to $x = z$ or $y = z$ due to

```
                               ┌ 1  0 ┐
#2:  SOLVE2(x·y, x + y = 1) =  │      │                     U ser=Simp(User)
                               └ 0  1 ┘
```

Assuming w.l.o.g. that $y = z$ we are left with

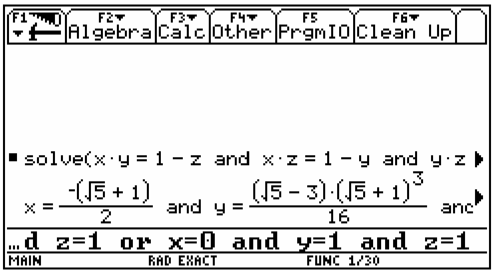
```
                                          ┌         0                1        ┐
                                          │                                   │
                                          │     √5      1        √5      1    │
                                2         │    ──── - ───       ──── - ───    │
#3:  SOLVE2(x·y - 1 + y, y  - 1 + x) =    │     2      2         2      2     │
                                          │                                   │
                                          │     √5      1        √5      1    │
                                          │ - ──── - ───     - ──── - ───     │
                                          └     2      2         2      2     ┘
```

The other two cases $x = y$ and $x = z$, respectively, are obtained by simple permutations of the solutions above. They yield two additional solutions, namely [1, 0, 1] and [1, 1, 0]. *TITBITS 9 deals also with this problem.*

SOLUTIONS of DERIVE 6 and solve of the TI-92 have no problems solving this non linear system, Josef 2010

SOLUTIONS(x·y = 1 − z ∧ x·z = 1 − y ∧ y·z = 1 − x, [x, y, z])

$$\begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ \frac{\sqrt{5}}{2} - \frac{1}{2} & \frac{\sqrt{5}}{2} - \frac{1}{2} & \frac{\sqrt{5}}{2} - \frac{1}{2} \\ -\frac{\sqrt{5}}{2} - \frac{1}{2} & -\frac{\sqrt{5}}{2} - \frac{1}{2} & -\frac{\sqrt{5}}{2} - \frac{1}{2} \end{bmatrix}$$



Heinz Rainer Geyer, St.Katharinen, Germany

Vielen Dank für Deine Materialien. Bei meinen Bemühungen um „Teilermengen" habe ich eine Funktion SET und die entsprechenden Mengenoperationen in *DERIVE* vermisst. Umständlich war auch, dass die IF-Funktion keine echte Leerausführung zulässt. Das "?" ist schwer zu bearbeiten . . . .

. . . . . Möglicherweise lässt sich die direkte Farbwahl ähnlich wie in BASIC direkt implementieren.

In meiner 6. Klasse bin ich gerade bei den Teilern und Vielfachen. Ich habe etwas herumexperimentiert. Hier findest Du die Ergebnisse meiner Zusammenfassung:

#1: Set of divisors, 1st attempt:--> √n

#2: $\text{DIVISOR1(n)} := \text{VECTOR}\left(\text{IF}\left(\frac{n}{i} = \text{FLOOR}\left(\frac{n}{i}\right), i, 0\right), i, 1, \sqrt{n}\right)$

#3: $\text{DIVISOR2(n)} := \text{VECTOR}\left(\text{IF}\left(\frac{n}{i} = \text{FLOOR}\left(\frac{n}{i}\right), i, 0\right), i, \text{FLOOR}(\sqrt{n}) + 1, n\right)$

#4: PART1(n) := SELECT(k ≠ 0, k, DIVISOR1(n))

#5: PART2(n) := SELECT(k ≠ 0, k, DIVISOR2(n))

#6: DIVSET(n) := APPEND(PART1(n), PART2(n))

#7: DIVSET(3450)

#8: [1, 2, 3, 5, 6, 10, 15, 23, 25, 30, 46, 50, 69, 75, 115, 138, 150, 230, 345, 575, 690, 1150, 1725, 3450]

#9: 0.11 sec – in 1996 10.8 seconds only by approXimating #7

#10: Attempt #2: Why not counting until n?

#11: $\text{DIVISOR0(n)} := \text{VECTOR}\left(\text{IF}\left(\frac{n}{i} = \text{FLOOR}\left(\frac{n}{i}\right), i, 0\right), i, 1, n\right)$

#12: PART0(n) := SELECT(k ≠ 0, k, DIVISOR0(n))

#13: PART0(3450)

```
#14:  [1, 2, 3, 5, 6, 10, 15, 23, 25, 30, 46, 50, 69, 75, 115, 138, 150, 230, 345, 575, 690, 1150,
        1725, 3450]

#15:  Again 0.11 sec - in 1996 it needed 10.3 sec

#16:  Attempt #3: it should work faster if working with half of the divisors!

                          ⎛ n             ⎞
#17:  PART3(n) := VECTOR ⎜ ─, k, PART1(n)⎟
                          ⎝ k             ⎠

#18:  PART3(3450) = [3450, 1725, 1150, 690, 575, 345, 230, 150, 138, 115, 75, 69]

#19:  D_SET(n) := APPEND(PART1(n), REVERSE(PART3(n)))

#20:  D_SET(3450) = [1, 2, 3, 5, 6, 10, 15, 23, 25, 30, 46, 50, 69, 75, 115, 138, 150, 230, 345,
        575, 690, 1150, 1725, 3450]

#21:  seems to work, but ...:

#22:  D_SET(625) = [1, 5, 25, 25, 125, 625]
#23:  ... hence:

      DSET(n) :=
        If √n ≠ FLOOR(√n)
#24:       D_SET(n)
           DELETE_ELEMENT(D_SET(n), DIM(D_SET(n))/2)

#25:  DSET(3450) = [1, 2, 3, 5, 6, 10, 15, 23, 25, 30, 46, 50, 69, 75, 115, 138, 150, 230, 345,
        575, 690, 1150, 1725, 3450]

#26:  DSET(625) = [1, 5, 25, 125, 625]
```

Ich bin sicher, es gibt bessere Lösungen. Jedenfalls ist mir der Unterschied zwischen **S**implify und appro**X** deutlich geworden.

*(Heinz Rainer complained that there are no set operations in DERIVE, he also would like to have a true "not - execution", the "?" is not very comfortable to work with. He also has the idea to choose the plot colour directly by a command (similar to BASIC). Working with divisors and multiples in his 6th form he experimented with sets of divisors. See Rainer's results. He is sure that there are better solutions, but he has learned the difference between **S**implify and appro**X**. In DERIVE 4.x you will find Set operations. And the DIVISORS are also implemented.)*

```
DIVISORS(3450) = [1, 2, 3, 5, 6, 10, 15, 23, 25, 30, 46, 50, 69, 75, 115, 138, 150, 230, 345,
   575, 690, 1150, 1725, 3450]
```

George Freeman and Al Köpf, Kuweit City

Dear Josef, we have come across this bug in *DERIVE* 3.02:

#1: $\left[F(x) := \dfrac{x}{\sqrt{(1+x^2)}},\ G(x) := \dfrac{x}{\sqrt{(1-x^2)}}\right]$

#2: $G(F(x)) = x$

#3: $F(G(x)) = -x\cdot SIGN(x^2 - 1)$

Can you tell us why?

DNL: *Try the following:*

```
#4:   x :∈ Real (-1, 1)
#5:   G(F(x)) = x
#6:   F(G(x)) = x
```

I believe I know George and Al. Köpf seems not to be an Arabic name. Yussuf.

Rüdiger Baumann, Celle, Germany

... Mit ihrer Punktfolge haben Carl Leinbach & Marvin Brubaker (DNL#22 rev, page 29) ein hübsches Beispiel geliefert. Verallgemeinert man die Konstruktion ein wenig, gelangt man zu Edward Sawada´s "misguided missile" (DNL#22 rev, page 8):

```
"MISMIS2.MTH"
[p0:=[0,0],p1:=[0.5,0.5*SQRT(3)],p2:=[1,0]]
P(r,s,k):=IF(k=0,p0,IF(k=1,p1,IF(k=2,p2,r*P(r,s,k-3)+s*P(r,s,k-2))))
FOLGE1(r,s,n):=VECTOR(P(r,s,k),k,0,n)
FOLGE1(0.9,0.1,15)
```

Wegen der entsetzlich langsamen Rekursion empfiehlt sich eine iterative Fassung:

```
FOLGE2(r,s,n):=ITERATES([b,c,r*a+s*b],[a,b,c],[p0,p1,p2],n)
```

Der Aufruf von

```
FOLGE2(0.9,0.1,15)
```

liefert das ominöse Sawad´sche "missile". Interessant sind auch die Fälle r + s > 1 und r + s < 1, beispielsweise *folge2(0.15,0.9,50)* und *folge2(0.05,0.9,80)*. Hier können die Schüler selbständig experimentieren und Vermutungen hinsichtlich Konvergenz bzw. Divergenz aufstellen.

Rüdiger Baumann shows a link between Carl Leinbach's & Marvin .Brubaker's recursive sequence of points in DNL#22 rev, p. 29 and E. Sawada's "misguided missile" on p. 8 (same issue). Generalizing the construction presented in the first contribution we obtain the "missile". As the recursive construction is very slow, Rüdiger recommends the iterative procedure. Furthermore he points out that there are interesting cases with r+s>1 and r+s<1. Students could set up conjectures concerning convergence and divergence.

Alfonso J. Población, Valladolid, Spain

Hello dear DERIVERS,
Try to plot the function x^(1/2) (I did using *DERIVE* 2.56 and 3.0). You can see that, obviously, this function takes [0,+inf) --> [0,+inf). Now plot ABS(x^(1/2)), and surprisingly, you will have a function defined in the whole real line. How can it be?
Then if you Simplify ABS(x^(1/2)), *DERIVE* gives you ABS(x^(1/2)). So, if you Simplify ABS((–4)^(1/2)), the result is 2 and this is not right. And the same happens for every even root.
I think this happens because *DERIVE* has implemented this simplification for odd roots for which it is valid. But I think this must be modified for even ones, don't you?
If I did something wrong or anyone has a further explanation, please let me know. Thanks.

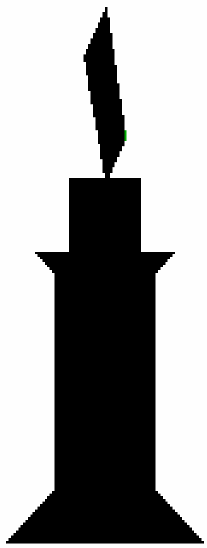
DNL: *There were some explanations in DERIVE News Group. In general all had the same content, so I´ll combine the different answers (michel-gosse@magic.fr, Al Rich, A. van der Meer, Josef, a.o):*
The problem with DERIVE is that it calculates with complex numbers. The absolute value of 2i then becomes 2.
Like virtually all CAS, DERIVE works in the complex domain, not just in the real domain.
You can define a function that will plot correctly, but that is a little tricky:

```
f(x) := if(x < 0, 0, abs(x^(1/2))).
```

Alfonso J. Población, Valladolid, Spain

Thank you very much about your message explaining what *DERIVE* does with the ABS-function. All the answers pointed out in the same direction, including one that I discovered later in the DNL#14, page 6, messages 2708 and 2710. I knew (but I forgot) that *DERIVE* works with complex numbers (that was a terrible oblivion), but what I did not know was that it plots in the way it does with these functions. A lot of our students usually work with *DERIVE* alone, and they can be confused about these behaviours, in case they detect them. Imagine a complicated function that involves ABS and SQRT and its plot: they believe in what they are seeing, so we must advise them (also, most of them do not know so much about complex numbers. In their first course they only deal with real numbers).

I have got a lot of tangrams. I will try to send them by e-mail ......

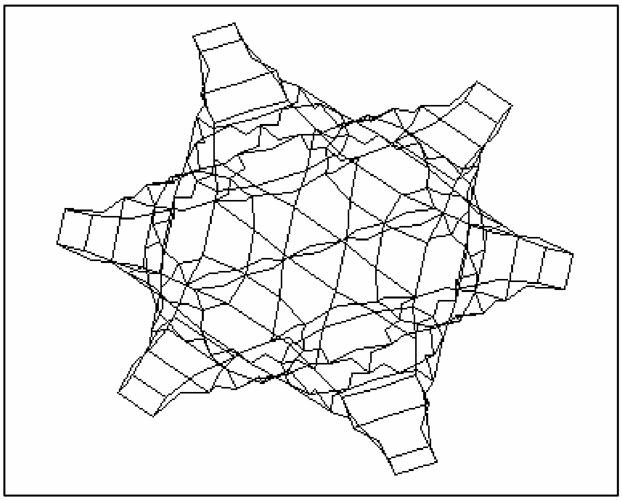


DNL: *Thanks for the tangrams. I´m sure they will show together with your contribution a new and still unknown facette of DERIVE and they combine in an ideal way mathematic tuition with entertainment.*

Sergey Biryukow, Moscow, Russia

Dear Josef, I have written functions for Implicit 3D Plots in *DERIVE* and I am writing DOC & DMO files now. Functions for Implicit 3D Plots in ACROSPIN are also written and return a vector of lines (2 × 2 numeric matrices). Is this format compatible with the tool you are going to present in the next DNL? I shall try to polish my IMPLICIT PLOTS utility as fast as possible in order to include it in the 1996 DNL diskette. Sincerely Sergey.

$$x^2\,y^2 + y^2\,z^2 + x^2\,z^2 = 1$$



DNL: *Yes, your results are compatible with my ACD, but it must be said that even without ACROSPIN you will obtain nice plots using DERIVE´s GRAPHICS.MTH.*

Dear DERIVERs, the DMO and DOC file are on my desk but I decided to include it in one of the next DNLs to not overload this issue with geometric contributions. So look forward to learning how to produce implicit 3D-plots. I add one screen dump to give an imagination of Sergey´s IMP_SURF.MTH (see above).

David Sjöstrand, Sweden

.... However I have some ideas concerning computer algebra and computer geometry. It is a challenge to do with *DERIVE* what you can easily do with Cabri II. So your mail really inspires me to make a contribution to the DNL. I am very fascinated by the TI 92 and some other TI graph calculators, Excel and Cabri, but I have the same feeling as you have: deep in my heart *DERIVE* is the number one mathematical software.

We are thinking of organizing a *DERIVE* - TI 92 - Cabri - Conference in Sweden in the end of summer 1997. It would be great to see you and Noor in Sweden then. Cheers, David.

DNL: *There is nothing to add except the fact, that we are quite sure to enjoy visiting Sweden if there is any chance to do so.*

kevans@ns.Sperry-Sun.COM

Is it possible to configure Classic *DERIVE* (V3.04) to print to another port, say LPT2, which points to a network printer?

M. Walkenhorst, BUNDOORA, Victoria, Australia (M.Walkenhorst@ee.latrobe.edu.au)

I was wondering, is it possible to do Z-Transforms with DERIVE?
For example the Z-Transform of E(s) = (s + 5)/(s^2*(s + 1)); for T = 0.01 sec

DNL: *Answer from* ***Al Rich, SWH***

It isn´t built in, but as I recall, the Z-transform is simply an infinite series. You might try that using the DERIVE sum function, and perhaps approximating infinity when an analytic result can´t be determined.

Another answer from ***znmeb@plaza.ds.adp.com:***

If you can do Laplace Transforms, you can do Z-transforms. I forget the formula, but there is a simple substitution that turns any Laplace transform into a Z-transform. Check books on signal processing, sampled data systems, etc. The two-sided Z-transform is a Laurent series, I think.

Harald Lang, Stockholm, Sweden (lang@math.kth.se)

My version of DERIVE Classic 3.13 has a bug. If I try to solve

$$(z - \#i)/(z + \#i) - (w - 1)/(w + \#i) = 0$$

for w, it says "memory full" after a few seconds. However, solving for *z* is fine. Both *z* and *w* are declared Complex. Cheers -- Harald Lang

Answer from **Alain Pomirol , Langon, France (Pomirol@aol.com)**
Same problem with 3.11 version of DERIVE XM. The given equation is simplified to

$$z/(z + \#i) + (1 - w) / (w + \#i) - \#i/(z + \#i)$$

but is not factorized. The solution is:

1. Factorize
2. Solve (equation, w)

It´s a problem of the employed method. Note: The TI 92 *csolves* this equation without any problems. Yours truly - Alain Pomirol

DNL: Same problem in DfW. Josef

#1: $\dfrac{z - i}{z + i} - \dfrac{w - i}{w + i} = 0$

#2: z :∈ Complex

Solution performed with DERIVE 6.

#3: w :∈ Complex

#4: $\text{SOLVE}\left(\dfrac{z - i}{z + i} - \dfrac{w - i}{w + i} = 0,\ w\right) = \left[w = - \dfrac{\sqrt{(- z^4 + 6\cdot z^2 - 1 + 4\cdot i\cdot z\cdot(z^2 - 1))} + i\cdot(- z^2 - 1)}{2 + 2\cdot i\cdot z} \lor w = \dfrac{\sqrt{(- z^4 + 6\cdot z^2 - 1 + 4\cdot i\cdot z\cdot(z^2 - 1))} + i\cdot(z^2 + 1)}{2 + 2\cdot i\cdot z}\right]$

#5: $\text{SOLVE}\left(\dfrac{z - i}{z + i} - \dfrac{w - i}{w + i} = 0,\ z\right) = \left[z = - \dfrac{\sqrt{(- w^4 + 6\cdot w^2 - 1 + 4\cdot i\cdot w\cdot(w^2 - 1))} + i\cdot(- w^2 - 1)}{2 + 2\cdot i\cdot w} \lor z = \dfrac{\sqrt{(- w^4 + 6\cdot w^2 - 1 + 4\cdot i\cdot w\cdot(w^2 - 1))} + i\cdot(w^2 + 1)}{2 + 2\cdot i\cdot w}\right]$

Gert von Morzé, Hannover, Germany

As a Power User of DERIVE I came across this bug in Version 3.02:

```
SUM(COMB(n,k),k,0,n) = ??
```

Can you tell me why?

DNL: *As I have learned form the DERIVE manual DERIVE uses "antidifferences" to calculate sums:*

"As with antiderivatives, closed-form antidifferences may not exist in terms of the operators and functions known to DERIVE. Even when such an antidifference exists, there is no known method that is guaranteed to find it." (DERIVE manual, page 202)..

It could be a little consolation that in this case unlike to the problem above, even the TI 92 gives up.

$$\#6: \quad \sum_{k=0}^{n} \text{COMB}(n, k) = 2^n$$

Solution performed with DERIVE 6.

Ian D´Souza, Montreal, Canada (ian@vir.com)

DERIVE Users: Help! I need answer for this one.
I´m running WINDOWS95 on a Pentium 120 with 32Mb RAM. DERIVE for WINDOWS can do the following integration without problems:

==> Declare a, b as Real (positive)
==> then Simplify

INT(x^4 EXP(–2a^2*b^2*x^2 / (a^2 + 3*b^2)),x,0,inf) ==> works as expected.

==> Now try to Simplify this one:

INT(x^4 EXP(–2a^2*b^2*x^2 / (a^2 + 3*b^2)) / (a^2 + 3*b^2)^(7/2),x,0,inf)

==> it gets stuck.

==> change the exponent of x from 4 to 5 (no change to the power of x inside the exponential function), Now it can do it!

It gets stuck for all even powers of $x \geq 4$ and works for all odd powers of x.

==> Substitute the denominator in the exponential function, (a^2 + 3*b^2), with G (declared Real - positive) and Simplify again - now it works no problem!! Also seems to work when substituting for the denominator of the expression itself.

The previous revision DERIVE v4.00 gets stuck for all powers of x that are ≥ 4 (even OR odd).
My old DERIVE v2.01 can do all these types of integrals with ease.

Integrals of the form $\int_0^\infty x^n e^{-kx^2}\, dx$ with k = positive are well defined.

*** What´s going on? *** Why does the substitution with G work?

$$\#1:\quad \int_0^{\infty} \frac{x^4 \cdot e^{-2 \cdot a^2 \cdot 2 \cdot b^2 \cdot x^2}}{(a^2 + 3 \cdot b^2)^{7/2}}\, dx = \frac{3 \cdot \sqrt{\pi}}{256 \cdot (a^2 + 3 \cdot b^2)^{7/2} \cdot |a \cdot b|^5}$$

Solution performed with DERIVE 6.

$$\#2:\quad \int_0^{\infty} \frac{x^5 \cdot e^{-2 \cdot a^2 \cdot 2 \cdot b^2 \cdot x^2}}{(a^2 + 3 \cdot b^2)^{7/2}}\, dx = \frac{1}{64 \cdot a^6 \cdot b^6 \cdot (a^2 + 3 \cdot b^2)^{7/2}}$$

Some of my other experiences with v4.00 point to a bug in the parser. Version 4.01 fixes some but others remain. DERIVE 4.00 and DERIVE XM have problems doing some integrals with very large (but constant) expressions of the form

INT(BigConstant, θ, 0, 2π)

where BigConstant is a very large expression, NOT a function of x. Where my old DERIVE v.2.01 correctly returns 2*π*Bigconstant, DERIVEXM and v4.00 seem to hang. I haven´t fully experimented with v4.01, but it can now do some trigonometric integrals that v2.01 can do, but that XM and v4.00 can´t do. I´ve played with all the Manage Trig settings.

One more thing: Note that v4.01 will not read a .MTH file (containing multicharacter Greek letters) created by v4.00! The extended ASCII characters used to represent concatenated Greek letters seems to have changed.
Thanks in advance for any answers! Ian

<u>Llorens Fuster, Valencia, Spain (llorens@mat.upv.es)</u>

Dear DERIVERS, Let a function F(u,i):=
(where u is for example a matrix, but that is not important). I want to programme the ´iterative - nested´ function:

F(u,1)
F(F(u,1),2)
F(F(F(u,1),2),3)
....
F(F(F......F(u,1),2),3).......)n)

where n (the number of iterations) depends on u.
IS IT POSSIBLE?

Another question: Is it possible to sort the elements of a vector? Thank you!

<u>DNL:</u> *I can only answer the second question: In DNL#13 you can find a sort routine:*
(The routine is on the next page. It is only of "historical" interest, because in the meanwhile the SORT-routine has been implemented, Josef)

```
SWAP_ELEMENTS(v,i,j):=VECTOR(IF(m_=i,ELEMENT(v,j),IF(m_=j,ELEMENT(v,i),
                           ELEMENT(v,m_))),m_,DIMENSION(v))

FIND_MIN(v,k,m):=IF(k>DIMENSION(v),m,IF(ELEMENT(v,k)<ELEMENT(v,m),
                     FIND_MIN(v,k+1,k),FIND_MIN(v,k+1,m)))

SORT_AUX(v,i):=IF(i>=DIMENSION(v),v,SORT_AUX(SWAP_ELEMENTS(v,i,
                   FIND_MIN(v,i,i)),i+1))

SORT_(v):=SORT_AUX(v,1)

SORT_([7,4,3,-9,9,5])=[-9,3,4,5,7,9]
```

I forwarded the other question to one of our ITERATES - RECURSION specialists - Josef Lechner - and hope to receive an answer.

Some Comments on the ROMBERG-Method

Josef Böhm

The Romberg Method (Werner Romberg, 1909 – 2003) is an improvement of the trapezoidal method for numerical integration applying the "Richardson Acceleration or Richardson Extrapolation" in order to improve the convergence of the method.

In Steven Schonefeld's book[1] I found:

The plan is to calculate the trapezoidal rule approximation for $h_1 = b - a$, $h_2 = h_1/2$, $h_3 = h_2/2$, ..., and then apply Richardson's improvement several times to increase the accuracy of the approximation to the integral. Of course, the first Richardson improvement on the trapezoidal rule results in Simpson´s rule.

Steven provides a DERIVE implementation (for DERIVE for DOS, of course) based on several connected functions. There was no programming possible in these times (1996).

Among many resources describing this famous numerical method I found one in Wikipedia[2]], showing the recursive procedure of this method:

$$R(0,0) = \frac{1}{2}(b-a)(f(a)+f(b))$$

$$R(n,0) = \frac{1}{2}R(n-1,0) + h_n \sum_{k=1}^{2^{n-1}} f(a+(2k-1)h_n)$$

$$R(n,m) = R(n,m-1) + \frac{1}{4^m-1}(R(n,m-1) - R(n-1,m-1))$$

or

$$R(n,m) = \frac{1}{4^m-1}(4^m R(n,m-1) - R(n-1,m-1))$$

where

$$n \geq 1$$
$$m \geq 1$$
$$h_n = \frac{b-a}{2^n}.$$

It is followed by a table for calculating ERF(1) with an accuracy of 10^{-8} (which I will use to check my Romberg-program – together with Terence Etchell´s results, of course).

[1] *Numerical Analysis via Derive*, Steven Schonefeld, Mathware 1994
[2] http://enwikipedia.org/wiji/Romberg's_method

The ERF(1) table (WIKI):

0.77174333				
0.82526296	0.84310283			
0.83836778	0.84273605	0.84271160		
0.84161922	0.84270304	0.84270083	0.84270066	
0.84243051	0.84270093	0.84270079	0.84270079	0.84270079

$$ERF(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, dt$$

ERF(1)

0.842700792949714

$$\int_0^1 \frac{2}{\sqrt{\pi}} \cdot \mathrm{EXP}(-t^2)\, dt$$

0.842700792949714

I found also a German Wiki-information, but according to my understanding there are some typos in it and following the instructions there the algorithm should not work (wrong subscripts?)

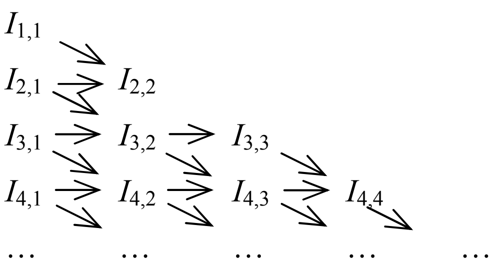
I wrote a DERIVE program following this recursive procedure – which is based on the corrected procedure provided on the German WIKI-page[3].

$$I_{n,k} = I_{n-1,k-1} + \frac{I_{n-1,k-1} - I_{n,k-1}}{2^{2-2k} - 1}$$

$$\text{with } I_{n,1} = \frac{h_n}{2}\left(f(a) + f(b) + 2\sum_{i=1}^{2^{n-1}-1} f(a + i \cdot h_n) \right) \text{ and } h_n = \frac{b-a}{2^{n-1}}$$

Then $I_{n,n}$ will deliver the n^{th} approximation.

The next diagram explains the calculation scheme:



The program is displayed on the next page.

Now I wanted to know – if my implementation is working and how the calculation times are performing compared with DERIVE's numerical integration, and with Terence's and Steven's procedures.
I am starting with Terence's integral from page 3.

Let at first do DERIVE its job – needing nearly half of a second for performing the integration:

[3] http://de.wikipedia.org/wiki/Romberg-Integration

$$\#3:\quad \int_0^4 \frac{\mathrm{SIN}(x)}{x}\,dx$$

#4: 1.75820313894905

0.469 sec

This is my DERIVE Romberg program:

```
rom_prog(f, x, a, b, n, eps, st, rtab, n_, k_, r_, i) :=
  Prog
    st := VECTOR((b - a)/2^(n_ - 1)/2·(LIM(f, x, a) + LIM(f, x, b) +
         2·∑(LIM(f, x, a + i·(b - a)/2^(n_ - 1)), i, 1, 2^(n_ - 1) - 1)), n_, n)
    rtab := [APPEND([st↓1], VECTOR("---", j, n - 1))]
    n_ := 2
    Loop
      If n_ > n exit
      k_ := 2
      r_ := [st↓n_]
      Loop
        If k_ > n_ exit
        i := rtab↓(n_ - 1)↓(k_ - 1) + (rtab↓(n_ - 1)↓(k_ - 1) -
            FIRST(REVERSE(r_)))/(2^(2 - 2·k_) - 1)
        r_ := APPEND(r_, [i])
        k_ :+ 1
      rtab := APPEND(rtab, [APPEND(r_, VECTOR("---", j, n - n_))])
      If ABS(rtab↓n_↓n_ - rtab↓(n_ - 1)↓(n_ - 1)) < eps ∧
         ABS(rtab↓(n_ - 2)↓(n_ - 2) - rtab↓(n_ - 1)↓(n_ - 1)) < eps
         RETURN rtab↓n_↓n_
      n_ :+ 1
    rtab
```

Now let´s compare:

Terence Etchells: 0.078 sec

#6:

1.75804691765892	28.0139169547487	1792.45321974934	$4.58861105626941\cdot10^{5}$	$4.6987333062835\cdot10^{8}$	$1.92460104930111\cdot10^{12}$
1.75819500516204	28.0140294546536	1792.45365177833	$4.58861112473565\cdot10^{5}$	$4.69873331065501\cdot10^{8}$	
1.75820265343736	28.0140362330084	1792.45367852957	$4.58861112900477\cdot10^{5}$		
1.75820310895234	28.0140366526776	1792.45368019761			
1.75820313707965	28.0140366788448				
1.7582031388323					

The first column contains the initial values which are found by Simpson's approximation. According to Steven's explanation, these values are the 2nd step values in his procedure starting with approximations obtained by applying the trapezoidal rule. Try finding the Simpson values in the next tables!

Steven Schonefeld: 0.047 sec

#5:

1	4	1.62159875234603	***	***	***
2	2	1.72009680299869	1.75292948654958	***	***
4	1	1.74855938899386	1.75804691765892	1.75838807973288	***
8	0.5	1.75578610112	1.75819500516204	1.75820487766225	1.75820196969287
16	0.25	1.75759851535802	1.75820265343736	1.75820316332238	1.75820313611063
32	0.125	1.75805196055376	1.75820310895234	1.75820313932001	1.75820313893902
64	0.0625	1.75816534294818	1.75820313707965	1.75820313895481	1.75820313894901

And this is my rom_prog:

#7: $\text{rom\_prog}\left(\frac{\text{SIN}(x)}{x}, x, 0, 4, 7\right)$

#8:

1.62159875234603	---	---	---	---	---
1.72009680299869	1.75292948654958	---	---	---	---
1.74855938899386	1.75804691765892	1.75838807973288	---	---	---
1.75578610112	1.75819500516204	1.75820487766225	1.75820196969287	---	---
1.75759851535802	1.75820265343736	1.75820316332238	1.75820313611063	1.75820314068482	---
1.75805196055376	1.75820310895234	1.75820313932001	1.75820313893902	1.75820313895011	1.75820313894841
1.75816534294818	1.75820313707965	1.75820313895481	1.75820313894901	1.75820313894905	1.75820313894905

My rom_prog: 0.047 sec

#9: $\text{rom\_prog}\left(\frac{\text{SIN}(x)}{x}, x, 0, 4, 7, 10^{-8}\right) = 1.75820313894905$

Including considering the required accuracy: 0.047 sec

The last column of the matrix in #8 is not displayed.

One can enter the required accuracy as last parameter and then the output is only the value of the integral if the number of steps is sufficient enough.

Finally I ask rom_prog to calculate the ERF(1)-value:

#10: $\text{rom\_prog}\left(\frac{2}{\sqrt{\pi}}\cdot\text{EXP}(-x^2), x, 0, 1, 6\right)$

#11:

0.771743332258053	---	---	---	---	---
0.825262955596749	0.843102830042981	---	---	---	---
0.838367777441205	0.842736051389356	0.842711599479115	---	---	---
0.841619221244767	0.842703035845955	0.842700834809728	0.84270066394196	---	---
0.842430505490232	0.842700933572054	0.84270079342046	0.842700792763488	0.84270079326867	---
0.842633227681257	0.842700801744931	0.842700792956457	0.842700792949091	0.842700792949819	0.842700792949508

0.016 sec

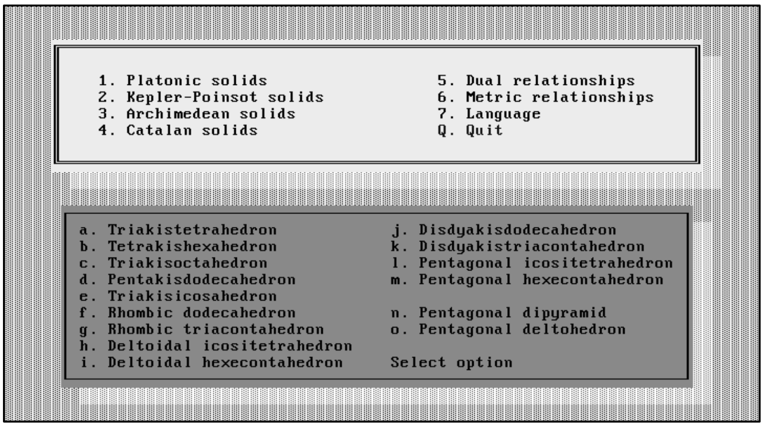
#12: $\text{rom\_prog}\left(\frac{2}{\sqrt{\pi}}\cdot\text{EXP}(-x^2), x, 0, 1, 8, 10^{-6}\right) = 0.842700792949508$

0.046 sec

This looks pretty good, doesn't it?

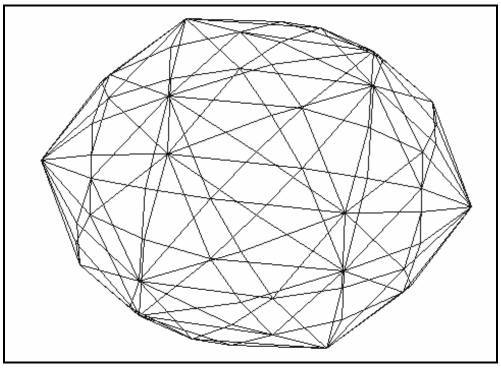
POLIEDROS 1.0

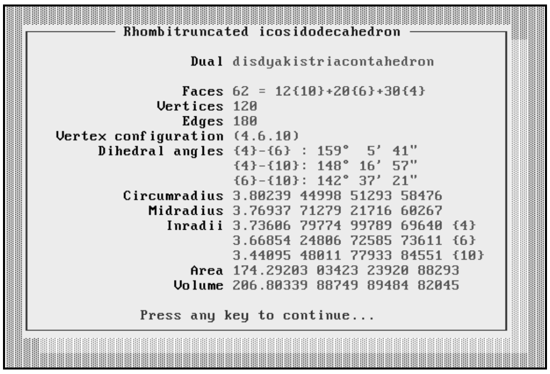
P.Familar Ramos



POLIEDROS Version 1.0 is a PC program that represents graphically regular polyhedra (Platonic and Kepler-Poinsot solids) and semi-regular polyhedra and their duals (Archimedian and Catalan solids) through AcroSpin™ . The program includes a study of the metric properties of each polyhedron (informing about the number of faces, vertices and edges, the vertex configuration, the dihedral angles, the circumradius, midradius and inradius, the surface area and volume). The calculations of these metric relationships have been avaluated with *DERIVE*™. The user is allowed to select the program language: English or Spanish.

Do you know a "Disdyakistriacontahedron"? And if so, do you also know that it is dual to a "Rhombitruncated Icosidodecahedron"? And at last, do you how it looks like? POLIEDROS gives an answer:





Very interesting for you is the fact, that POLIEDROS includes the ACROSPIN.EXE file. Math Ware has granted AWR Software a non-transferrable license agreement, subject to renegotiation, to use AcroSpin in POLIEDROS.

Available at AWR Software, Huertos 21, 46500 SAGUNTO (Valencia), Spain, FAX 96 266 34 07 (3700 Ptas)

An Introduction to the Mathematics of Biology

by Yeagers, Shonkwiler and Herod

The authors of this textbook have adopted the philosophy that mathematical biology is not merely the intrusion of one science into another but has a unity of its own. The biology and mathematics are equal; they are complete and flow smoothly into and out of one another.

The book has several important features that the authors have developed from their classroom experience. A unique feature is the use of a CAS, Maple, in parts of every chapter. The models can easily be transferred to other CA-systems as DERIVE. Graphic visualizations are provided for all the mathematical results.

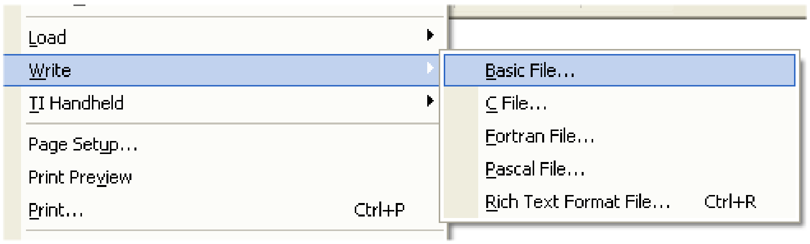
The chapters are: *Biology, Mathematics, and a Mathematical Biology Laboratory; Some Mathematical Tools; Reproduction and the Drive for Survival; Interactions between Organisms and their Environment; Age-Dependent Population Structures; Random Movements in Space and Time; The Biological Disposition of Drugs and Inorganic Toxins; Neurophysiology; The Biochemistry of Cells; A Biomathematical Approach to HIV and AIDS; Genetics*.

The text has extensive exercises, problems, and examples, along with references for further study.

Birkhäuser Boston 1996, ISBN 1 55953 107 X, 434 pages, Hardcover, DM 118.-, AS 862.-, sFr 98.-

This contribution from 1996 is of “historical and nostalgic” interest. In 1996 there was no 3D plot possible from within *DERIVE*. What we could do was producing 3D-projections (isometric and other parallel projections, or perspective projections applying self made projection procedures). And there was David Parker’s inexpensive *ACROSPIN*. We could call *ACROSPIN* from *DERIVE* – but only for functions of two variables – not for all other kinds of 3D objects.

What I was demonstrating in 1996 can be reproduced in the DOS-environment even today – if you have *ACROSPIN* available[1]. But there is one obstacle. Compare the two files below. Both files have the same source and are saved as BASIC-files (what can be done in *DERIVE* 6 via the Write-option).



This is how it looked like with *DERIVE* 3.14.


```
Derive 3.14

[[2,0.83,1.7],[2,-0.83,1.7],[2.9,0,1.2],[2,0.83,1.7],[2.9,0,1.2],[3.7,-
1.53,0],[4,0,0],[3.7,1.53,0],[2.9,0,1.2],[4,0,0],[3.7,-1.53,0],[2,0,-
2.2],[4,0,0],[3.7,1.53,0],[2,0,-2.2],[0,0,-4]]
□

Derive 6.10

[[2,0.83,1.7],[2,-8.3E-1,1.7],[2.9,0,1.2],[2,0.83,1.7],[2.9,0,1.2],[3.7,-
1.53,0],[4,0,0],[3.7,1.53,0],[2.9,0,1.2],[4,0,0],[3.7,-1.53,0],[2,0,-
2.2],[4,0,0],[3.7,1.53,0],[2,0,-2.2],[0,0,-4]]
□
```

Compare –0.83 and -8.3E-1. I believe that it would be wasted time to change my BASIC-program ACD from 1996 and compile it again, because we do have now 3D-plots in *DERIVE*.

In the following contribution I will add the respective *DERIVE 6* plots. You may compare.

Much fun reading or rereading *DERIVE* – ACD – *ACROSPIN*.

Josef

[1]*I have a little present for all of you who have still the DERIVE DOS-version: David Parker gave permission to distribute the original ACROSPIN to the DUG-Members. you can download it among the other DNL24-files. Many thanks to David.*

Interaction between *DERIVE* and *ACROSPIN* with ACD.

Josef Böhm, Würmla, Austria

Using *DERIVE 3.x* you can save surfaces as ACD-files if the surface is defined as a function of two variables *f(x,y)*. For other 3D solids it is not possible to have a direct conversion to an ACD-file for running *ACROSPIN*-animations. It is also impossible to produce *ACROSPIN*-files for space curves and for polyhedrons from the *DERIVE*-environment. I must admit that I am a fan of *ACROSPIN* and I would like to recommend strongly to buy this piece of software. Nevertheless you may find this contribution also useful if you don´t want to use ACROSPIN. Using the functions ISOMETRIC and COPROJECTION from *DERIVE*´s utility file GRAPHICS.MTH you can also produce impressive plots. See the end of this paper. Fortunately there are a lot of other DERIVIANS which like geometry, so I am glad to announce for the next DNLs some contributions to produce several mappings of objects (H. Kümmel a.o.) and - a hidden line algorithm for *DERIVE* from our friend Hubert Weller.

It is easy analysing the format of an ACD-file and then to edit a few 3D-points for simple objects for one´s own ACD-file. To do so for objects consisting of many points this will be a boring work.

So I remembered my "programming past" and wrote a tool, **ACD.EXE** which can help. I am sure that *DERIVE* and *ACROSPIN* as well may benefit

Now you are able to animate space curves, polyhedrons and surfaces (given in any parameter form) in any combination of objects, colours and layers. Using an algorithm of Richard Schorn from Kaufbeuren you can produce analglyphs (red-green-pictures) to obtain a stereographic presentation of the object. At this place I want to thank Mr Schorn for his comments and support. Many letters were exchanged between Kaufbeuren and Würmla to share experiences.

First you have to produce the list of 3D-points of the object in *DERIVE*. Work in **Approximate Mode** and use **3 digits** in **Notation** (from the Options submenu).

If you connect the points then they will form a space curve or a polyhedron or only a polygon in space. Save the DERIVE-expression as a **BASIC**-file. So the most important thing is to create lists of points in the right order. These lists of points can be used for many other projections (parallel or central perspective, ...) or the projection process can be done by another tool - like ACROSPIN. You can also use this lists in connection with ISOMETRIC.

So I will start with a toolbox for producing this lists of points for 3D objects.

```
"P3D.MTH"

[InputMode:=Word,Precision:=Approximate]

[PrecisionDigits:=4,Notation:=Decimal,NotationDigits:=3]

"The next 7 functions are from:"

"File GRAPHICS.MTH, copyright (c) 1990 by Soft Warehouse, Inc."

COPROJECTION(v):=VECTOR(VECTOR(u_ SUB n_,u_,v),n_,DIMENSION(v SUB 1))

SPHERE(r,theta,phi):=r*[SIN(phi)*COS(theta),SIN(phi)*SIN(theta),COS(phi)]
```

```
CYLINDER(r,theta,z):=[r*COS(theta),r*SIN(theta),z]

CONE(alpha,theta,z):=[z*SIN(alpha)*COS(theta),z*SIN(alpha)*SIN(theta),z]

NORMAL_VECTOR(v,t):=SIGN(DIF(v,t,2))

BINORMAL(v,t):=SIGN(CROSS(DIF(v,t),DIF(v,t,2)))

SPACE_TUBE(v,t,r,phi):=v+r*(SIN(phi)*NORMAL_VECTOR(v,t)+COS(phi)*
                           BINORMAL(v,t))

"The following is not part of GRAPHICS.MTH:"

FIG(v_,p1,p2,p1a,p1e,n,p2a,p2e,m):=VECTOR(VECTOR(v_,p2,p2a,p2e,
   (p2e-p2a)/m),p1,p1a,p1e,(p1e-p1a)/n)

RO(obj,n):=VECTOR(obj . ROTATE_Z(2*pi*k/n),k,1,n)

FIGUR_W(p_,w_):=APPEND(VECTOR([ELEMENT(p_,ELEMENT(w_,k1_))],k1_,
                DIMENSION(w_)))
FIGUR_E(p_,e_):=APPEND(VECTOR([ELEMENT(p_,ELEMENT(e_,k1_,1)),ELEMENT(p_,
                ELEMENT(e_,k1_,2)),[inf,inf,inf]],k1_,DIMENSION(e_)))

P1(m,zn):=VECTOR(VECTOR([ELEMENT(m,i,1)*COS(phi_),ELEMENT(m,i,1)*SIN(phi_),
          ELEMENT(m,i,2)],phi_,0,2*pi,2*pi/zn),i,1,DIMENSION(m))

P2(m,zn):=COPROJECTION(P1(m,zn))

REV(m,zn):=APPEND(P1(m,zn),P2(m,zn))
```

You can create a polyhedron with a list containing all the edges of the solid: [[P1,P2],[P2,P3],.....[Pi,Pj],......], with Pi = [xi,yi,zi]. Save this file as a **BASIC**-file. I prepared some *DERIVE* - functions to produce polyhedrons from a list of its points and either a list describing the way how to connect the points or a list (matrix) containing the edges: P_WAY(points,way) and P_EDG(points,edges).

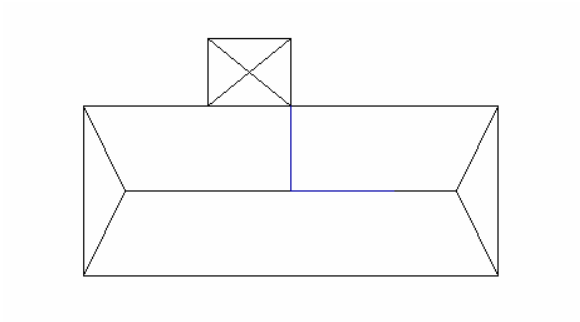
```
points:=[[0,0,0],[3,-2,-1],[1,5,10],[1,2,5]

way:=[1,2,3,1,4,3,4,2]

edges:=[[1,2],[2,3],[3,1],[1,4],[2,4],[3,4]]

P_WAY(points,way)

P_EDG(points,edges)
```

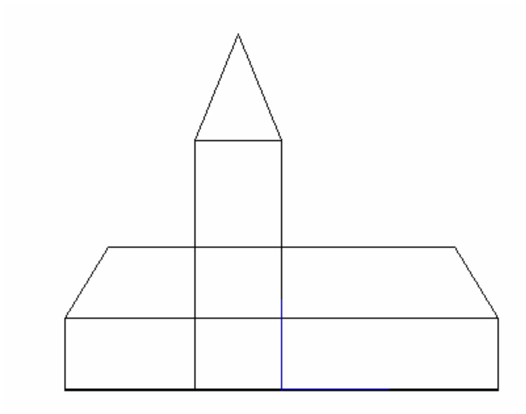
Both functions will when **Simplified** return the points of a pyramid.
Or you have produced a family of parameter lines with parameters *u* and *v* using the powerful VECTOR(VECTOR(f(u,v),.......))-construction. Using COPROJECTION from GRAPHICS.MTH you will have both families of parameter lines with either *u* or *v* constant. Save the two expressions as different **BASIC**-files, leave *DERIVE* and then call **ACD**. The desire to produce *ACROSPIN*-demonstrations of solids of revolutions was the inspiring idea to create a tool like ACD to make that possible.
Executing **ACD**, you will find a menu in which you are asked to enter the type of object which you want to prepare for *ACROSPIN*. You are able to combine several different objects in one ACD-file with both individual colours and layers. See now some examples:

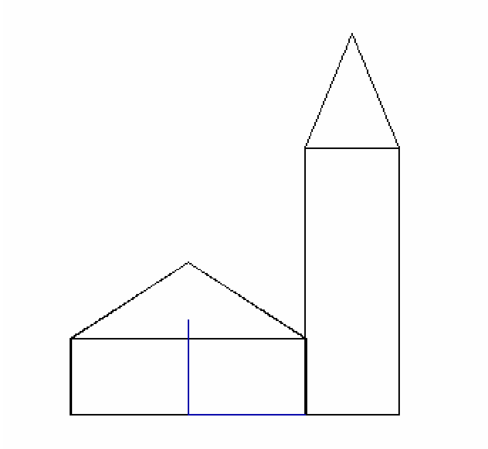
```
                         From DERIVE to ACROSPIN


    Which kind of object do you want to display? Make your choice:

    Space Curve or Polyhedron (from a list of points) ... 1
                in Stereo - Vision ...................... 4

    Surface (show the families of parameter lines) ...... 2
                in Stereo - Vision ...................... 5

    or show the complete surface ........................ 3
                in Stereo - Vision ...................... 6

    Polyhedron (from a list of edges) ................... 2
                in Stereo - Vision ...................... 5

    Discrete points ...................................-1
                in Stereo - Vision ...................... 7

                Quit .................................... 0

                   Your choice please:
```

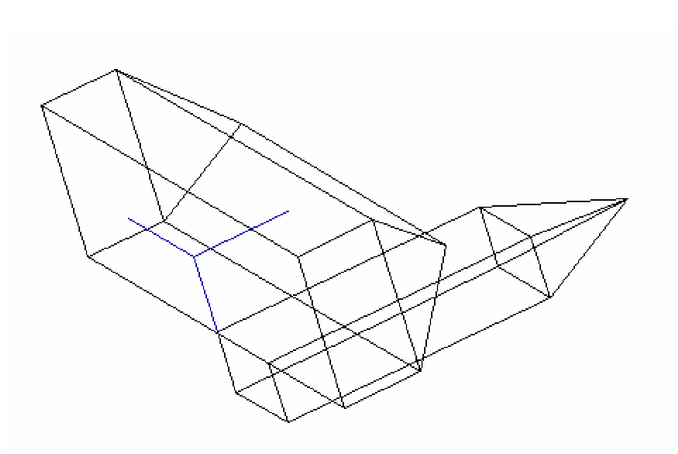
From my point of view the most interesting thing is the fact that I can interact between the wonderful capabilities of *DERIVE* and the demonstrating facilities of *ACROSPIN*. We could use this tool to train the students´ imagination of 3D space. Give them the task to design 3D objects by the coordinates of the vertices and the edges. Let them construct well known geometric solids like tetrahedrons, octahedrons, pyramids, parts of a cube, and so on. They immediately can see if they are right or not.

Rotating in *ACROSPIN* you will obtain top- front- and side view of your object. You have to imagine all the following pictures in different colours and layers. So you can switch off and on different parts of the objects. See a composition of the objects `axes`, `house` and `tower`.



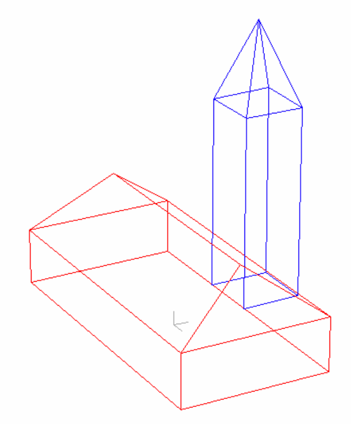






In DERIVE 6 you have to Insert (F4) the objects separately
(FIGUR_W(axes, axesw),
FIGUR_W(tow, toww),
and FIGUR_W(house, housew) respectively.

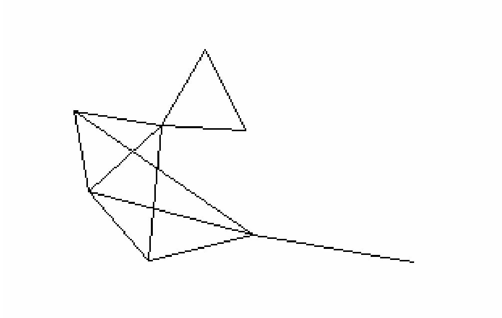
Then change the Scheme in the Color Plot Window to Custom and choose the color of your choice for the Grid.

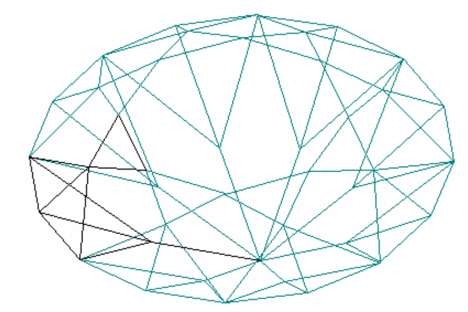


I called the first combination of objects "VILLAGE". In the second figure you can see the eighth of a diamond. Rotating this part seven times using *DERIVE* the whole figure will emerge:
(You can find the objects in P_3D.MTH).

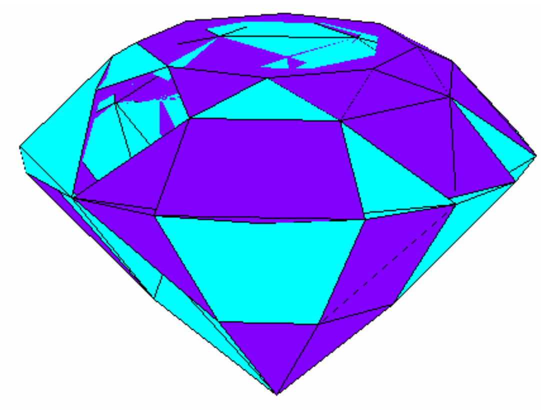
```
"The Diamond – (the girls' best friend)"

diam:=[[2,-0.83,1.7],[2.9,0,1.2],[3.7,-1.53,0],[4,0,0],[2,0,-2.2],
      [0,0,-4],[2,0.83,1.7],[3.7,1.53,0]]

diamw:=[7,1,2,7,2,3,4,8,2,4,3,5,4,8,5,6]

FIGUR_W(diam,diamw)

RO(FIGUR_W(diam,diamw),8)
```

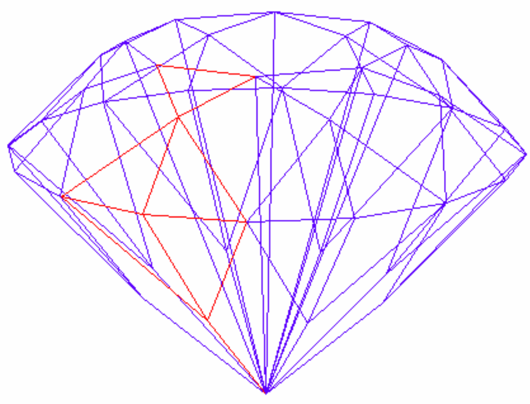
Save both results as different BAS-files, then run ACD. For the first part apply option 1 and then append the whole object in another colour using option 2.



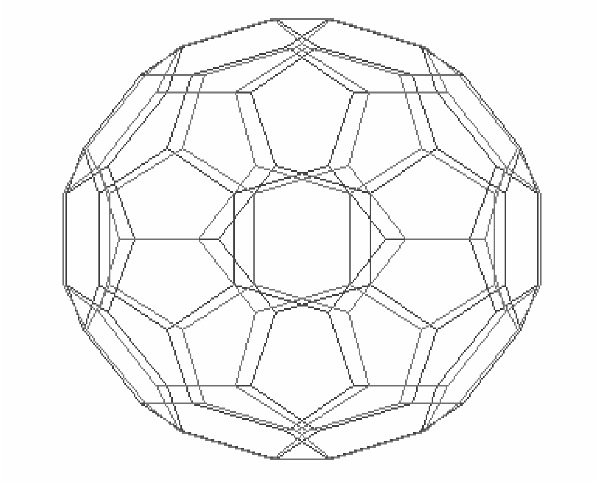


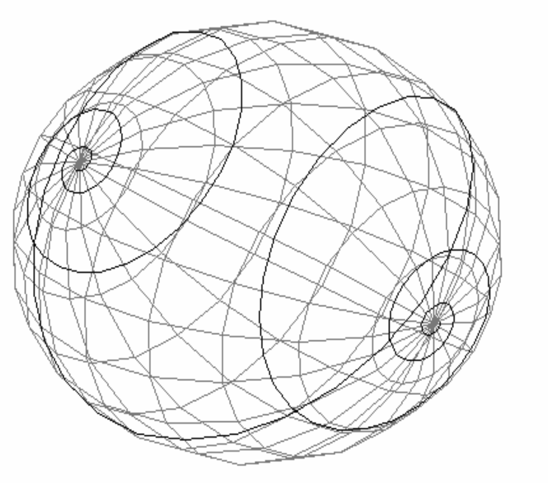
`RO(FIGUR_W(diam,diamw),8)` results in the left plot in DERIVE 6, you have to
`APPEND(RO(FIGUR_W(diam, diamw), 8))`



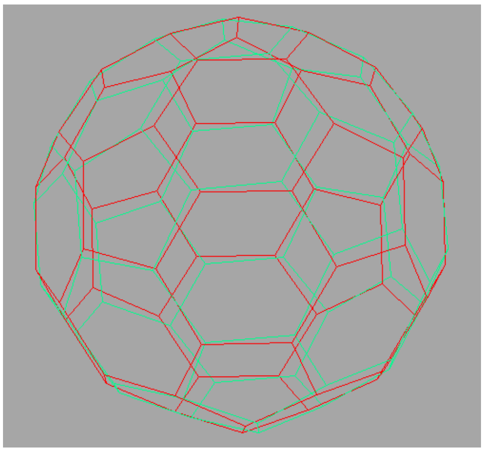


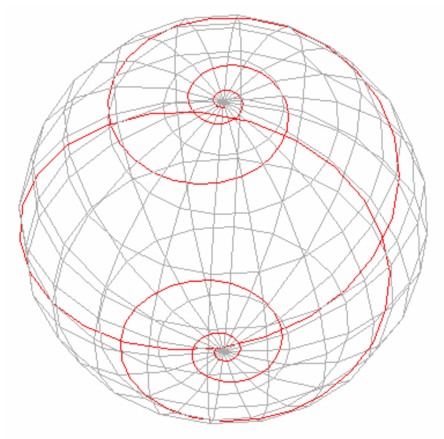
The left picture is an analglyph of a C60-molecule model (see the contribution submitted by Richard Schorn in DNL#21). You have to imagine red and green lines and viewing them through a red-green-glass.





The right picture is composed from a sphere and a space curve - a loxodrome. You can make visible the single points applying option 1 and then produce the closed curve using option 2. I'll lead you the way how to do it.





At first we have to produce the list of points using *DERIVE*:

```
#28:[c:=0.2,r1:=#e^(c*phi)]

#29:[r0:=1/SQRT(1+r1*r1),w:=ATAN(r1),r:=r0*SIN(w)]

#30:VECTOR([r*COS(phi),r*SIN(phi),r0*COS(w)],phi,-20,0,0.25)

#31:VECTOR([r*COS(phi),r*SIN(phi),r0*COS(w)],phi,0,20,0.25)

#32:VECTOR(VECTOR([0.5*SIN(phi)*COS(theta),0.5*SIN(phi)*SIN(theta),
    0.5*COS(phi)+0.5],phi,0,2*pi,pi/10),theta,0,2*pi,pi/10)

#33:FIG([0.5*SIN(phi)*COS(theta),0.5*SIN(phi)*SIN(theta),0.5*COS(phi)+0.5],
    theta,phi,0,2*pi,20,0,2*pi,20)
```

#30 and #31 give the points of the loxodrome. I had to divide into two parts, because ACD can work only with vectors consisting of 100 elements (each of which can contain again 100 components).

Expression #33 is the parameter form of a sphere. Using my FIG-function you can avoid the bulky `VECTOR(VECTOR(......))` - command.

ApproXimate #30, #31 and #33.

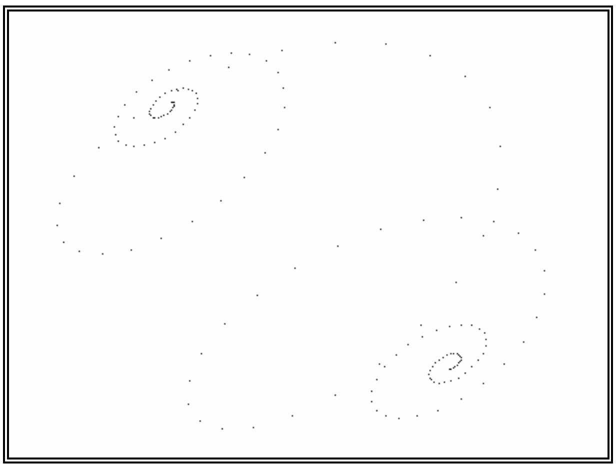
Save each of the results as a single BAS-file with different names, eg LOX1, LOX2 and SPH respectively. Quit *DERIVE*. Take care that ACD.EXE and ACROSPIN.EXE are in the same directory. Then call ACD.

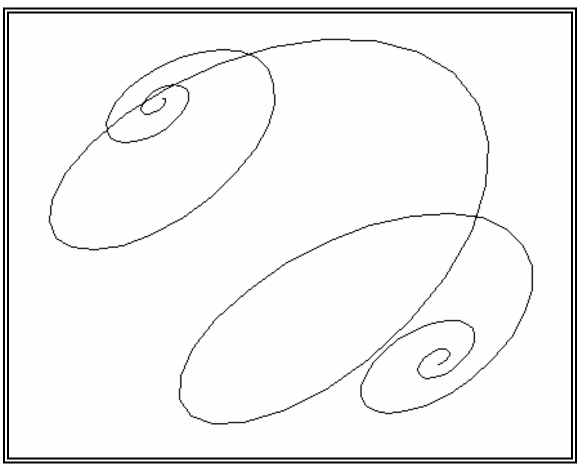
Choose option 1, because we want to see the points which build up the space curve,

LOX1	[ENTER]	*the name of the first BAS-file*
LOXO	[ENTER]	*that´s the name of the ACD-file to be created now*
15	[ENTER]	*the points should be white (colour code 15)*
1	[ENTER]	*the 1st layer*
y		*because we have not finished, repeat with* LOX1*, you will not be asked for the ACD-file´s name once more. If you would finish now you would obtain only the points (next figure!!), but we will continue:*
1	[ENTER]	*we will see the space curve*
LOX1	[ENTER]	*the same point list will be used*
4	[ENTER]	*we want to have a red curve*
2	[ENTER]	*it should be another layer*
y	[ENTER]	*because we are not yet ready, repeat the two steps with* LOX2.

If we would finish now we could see the points and the curve in two layers, which could be toggled on and off using the [ENTER]–key in combination with the [1]- and[2]-key. As we want to add the sphere we go on once more and proceed:

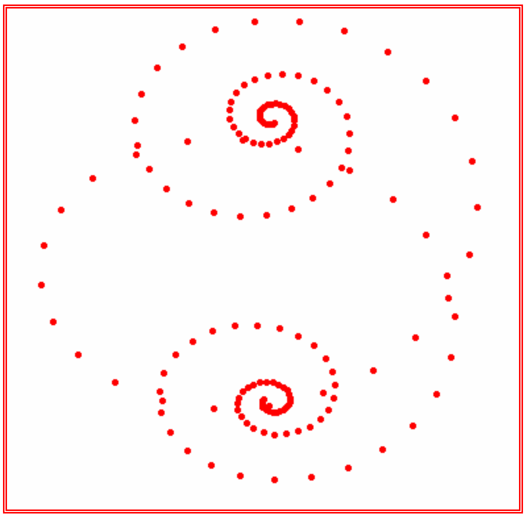
3	[ENTER]	*the whole surface of the sphere*
SPH	[ENTER]	*the name of the corresponding BAS-file*
14	[ENTER]	*let´s have a yellow sphere*
3	[ENTER]	*in the 3rd layer*
n		*we have finished. Now start typing ACROSPIN LOXO* [ENTER]





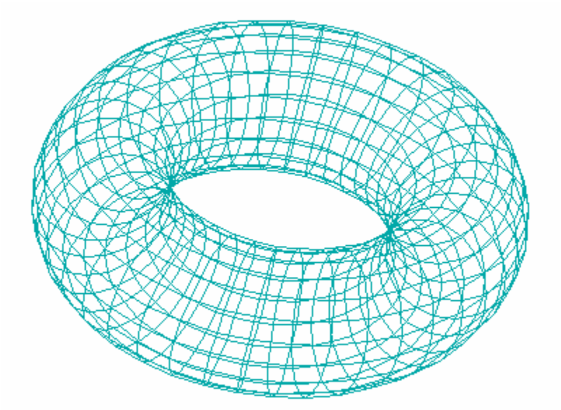
These are the DERIVE 6 commands:

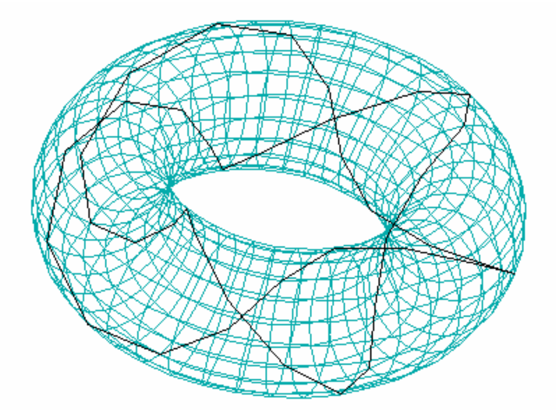
```
APPEND(COPROJECTION(FIG([0.5·SIN(φ)·COS(θ), 0.5·SIN(φ)·SIN(θ), 0.5·COS(φ) + 0.5], θ, φ, 0, 2·π, 20, 0, 2·π, 20)))
APPEND(FIG([0.5·SIN(φ)·COS(θ), 0.5·SIN(φ)·SIN(θ), 0.5·COS(φ) + 0.5], θ, φ, 0, 2·π, 20, 0, 2·π, 20))
VECTOR([r·COS(φ), r·SIN(φ), r0·COS(w)], φ, -20, 20, 0.25)
VECTOR([[r·COS(φ), r·SIN(φ), r0·COS(w)]], φ, -20, 20, 0.25)
```



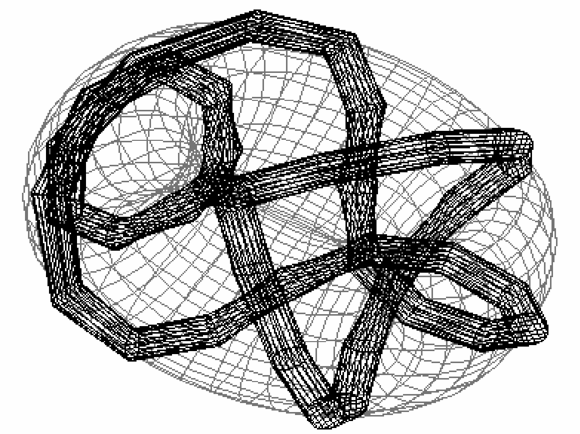
Inspired by books dealing with other CAS - packages I tried to produce a more sophisticated animation: an elliptic torus with a torus knot line on it. And to make the space curve more impressive I superimposed its tube. (Using the utility file GRAPHICS.MTH from SWHH).

```
#103:TKN(a, b, c, p, q, t) := [(a + b·COS(q·t))·COS(p·t),
                               (a + b·COS(q·t))·SIN(p·t), c·SIN(q·t)]

#104:TKN(8, 3, 5, 2, 5, t)

#105:[3·COS(2·t)·COS(5·t) + 8·COS(2·t), 3·SIN(2·t)·COS(5·t) + 8·SIN(2·t),
      5·SIN(5·t)]

#106:VECTOR(TKN(8, 3, 5, 2, 5, t), t, 0, 2·π, π/40)

#107:ELL_TOR(a, b, c, Θ, Φ) := [(a + b·COS(Φ))·COS(Θ), a + b·COS(Φ)·SIN(Θ),
                                c·SIN(Φ)]

#108:FIG(ELL_TOR(8, 3, 5, Θ, Φ), Θ, Φ, 0, 2·π, 40, 0, 2·π, 40)

#109:SPACE_TUBE(TKN(8, 3, 5, 2, 5, t), t, 2, Φ)

#110:FIG(SPACE_TUBE(TKN(8, 3, 5, 2, 5, t), t, 1, Φ), t, Φ, 0, 2·π, 80, 0,
          2·π, 20)
```

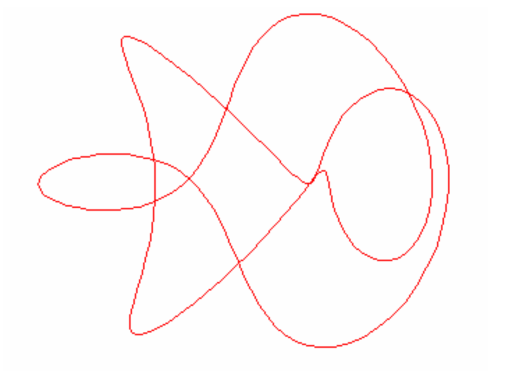


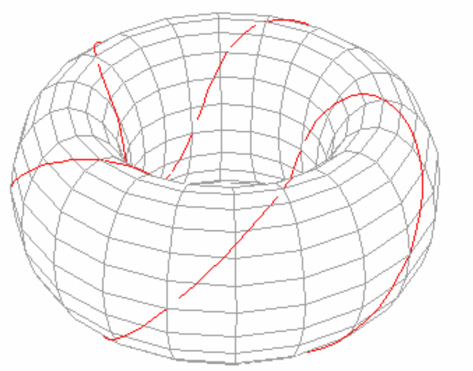


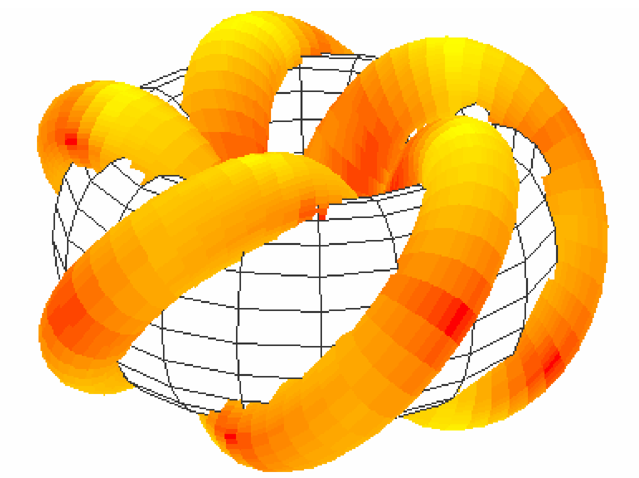
You have to approximate the expressions #106, #108 and #110, save them in three different BAS-files and then use ACD. Please be patient approximating expression #110. Using ACD I chose colour 1 for the space curve, colour 4 for the torus and colour 14 for the tube.



DERIVE 6 graphs are below.







If you know the parameter form of a surface it is very easy to produce the family of parameter lines with a VECTOR(VECTOR(.......)) function and then to apply option 2 or 3 from **ACD.EXE** to create the animation of the surface. You all will know the famous Moebius Strip. Its parameter form is given by:

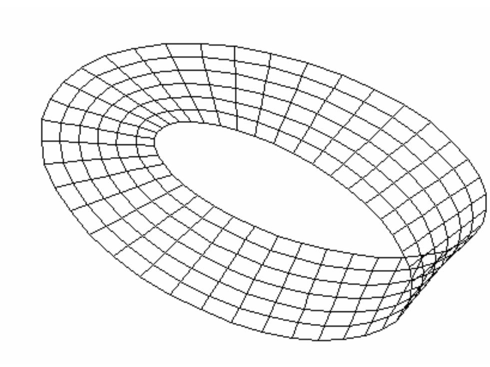
$$F(u,v) \;=\; a\left[\;\cos u\left(1\;+\;v\cos\left(\frac{u}{2}\right)\right),\;\sin u\left(1+v\cos\left(\frac{u}{2}\right)\right),\,v\sin\left(\frac{u}{2}\right)\right].$$

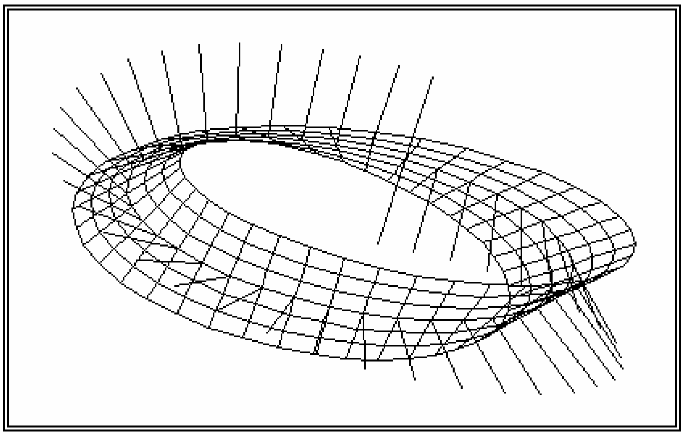
With a = 1 we obtain the *DERIVE* representation.

```
VECTOR(VECTOR([COS(u)+v*COS(u/2)*COS(u),SIN(u)+v*COS(u/2)*SIN(u),
      v*SIN(u/2)],u,0,2*pi,pi/20),v,-0.3,0.3,0.1)
```

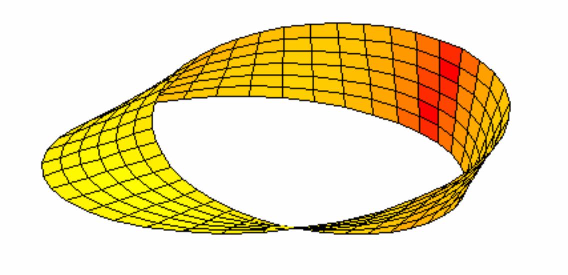
instead of `VECTOR(VECTOR([........))` you can use my `FIG()`-function:

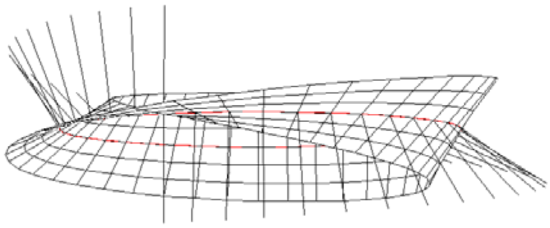
```
FIG([COS(u)+v*COS(u/2)*COS(u),SIN(u)+v*COS(u/2)*SIN(u),v*SIN(u/2)],u,v,
   0,2*pi,40,-0.3,0.3,60)
```





```
                     ┌   u  ┐                    ┌  u  ┐             ┌  u ┐┐
#110:mb := ┌COS(u) + v·COS│ ─── │·COS(u), SIN(u) + v·COS│ ─── │·SIN(u), v·SIN│ ── ││
           └              └  2  ┘                    └  2  ┘             └  2 ┘┘

#111:FIG(mb, u, v, 0, 2·π, 40, -0.3, 0.3, 6)

                               ┌       d               d    ┐
#112:(lim  mb) + t·CROSS│lim  ── mb, lim  ── mb│
      v->0                     └v->0 du       v->0 dv   ┘

#113:[t·COS(u)·SIN(0.5·u) + COS(u), t·SIN(u)·SIN(0.5·u) + SIN(u), -t·COS(0.5·u)]

#114:"v = 0 gives the circle in the middle of the strip"

#115:[0·COS(u)·SIN(0.5·u) + COS(u), 0·SIN(u)·SIN(0.5·u) + SIN(u), -0·COS(0.5·u)]

#116:"t = 0 --> pedal points of the normals"

#117:[COS(u), SIN(u), 0]

#118:"t = 0.5 --> end points of the normals"

#119:[0.5·COS(u)·SIN(0.5·u) + COS(u), 0.5·SIN(u)·SIN(0.5·u) + SIN(u),
     -5·COS(0.5·u)]

                          COS(u)                          SIN(u)
#120:VECTOR┌┌
           └└ 0.5·COS(u)·SIN(0.5·u) + COS(u)   0.5·SIN(u)·SIN(0.5·u) + SIN(u)   -

           0                              π
                         ┐, u, 0, 2·π, ───┐
    0.5·COS(0.5·u)       ┘              20  ┘
```



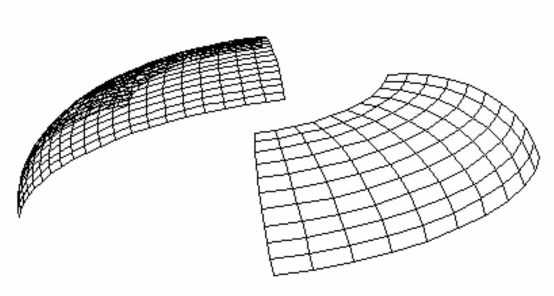


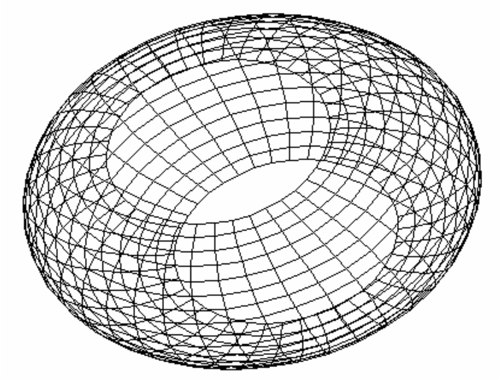
As you can imagine you can use this interaction between *DERIVE* and ACROSPIN not only to produce "nice" pictures. In a comfortable way you can make visible results from differential geometry. So you see on the other Moebius strip the normals. You could add the tangents, the normals and binormals to space curves, the tangent planes and, and, ...

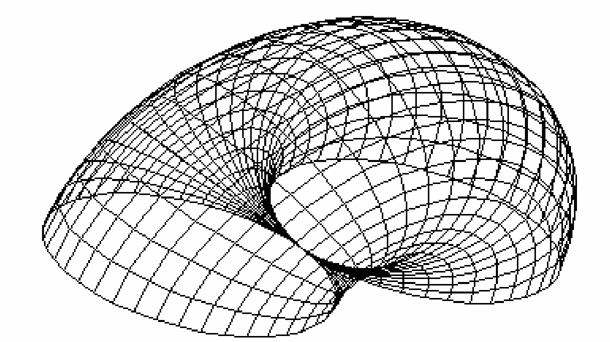
Approximate expressions #111 and #120 and save them as you have done before. Then apply option 3 for the strip and option 2 for family of normals (try option 3, then you will obtain a second strip!!??).

There is another idea to use this interaction: make clear the meaning of the parameters. Let me demonstrate my idea:

We create a spindle torus and then try to produce certain sections:



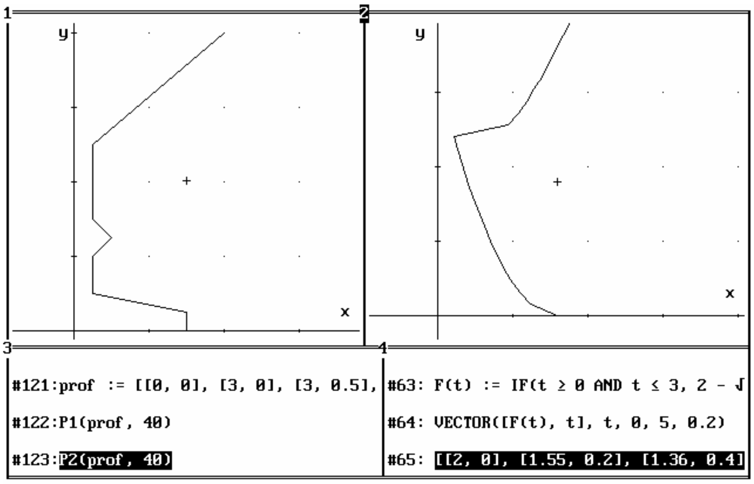




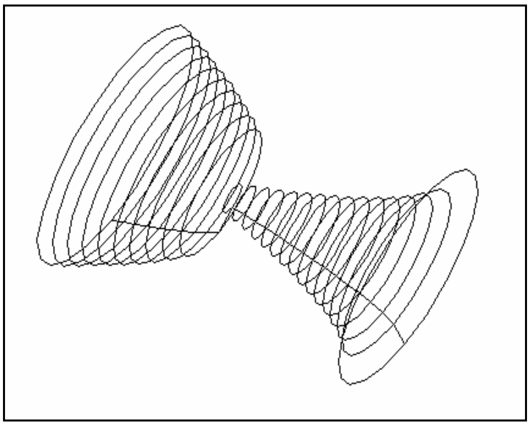
Which values for the parameters are responsible for the various torus parts?

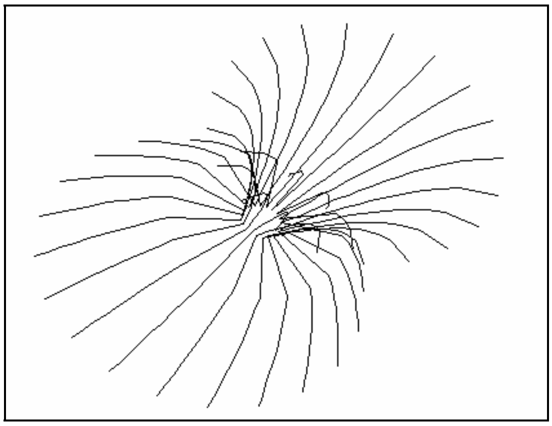
Unfortunately I cannot print in colours - maybe in some years we will have a colour print DNL???
(Yes, now we have!!)

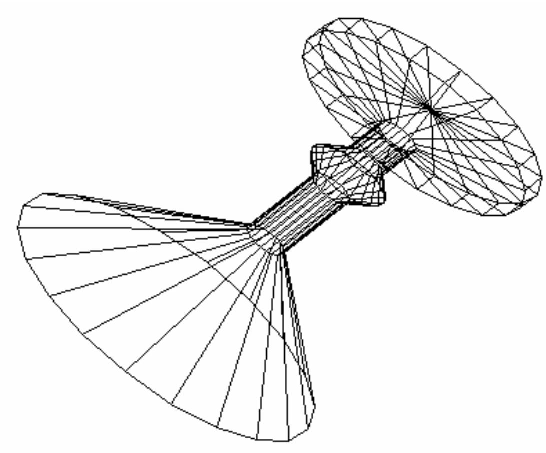
As I have mentioned before, I wanted to demonstrate solids of revolution. I also wanted to make visible the two families of parameter lines. So I produced a *DERIVE* function to create these surfaces from any given profile. We can design a profile by a list of points or by a – piecewise defined – function. I show one example for each possibility (left and right plot windows):

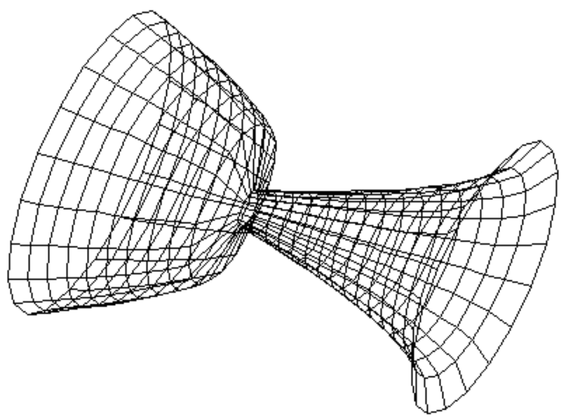


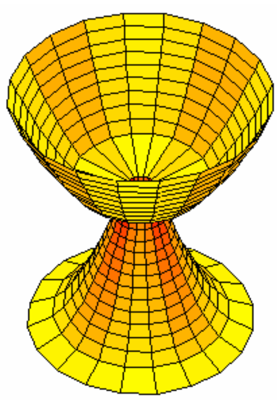
The next plots show the solids: you can see the profile together with the parallel circles, then the longitudinal intersections and at last the full grid. It is very nice to overlay the different layers and then to rotate the glasses round their axes, to zoom in and out, to translate, to accelerate, ...





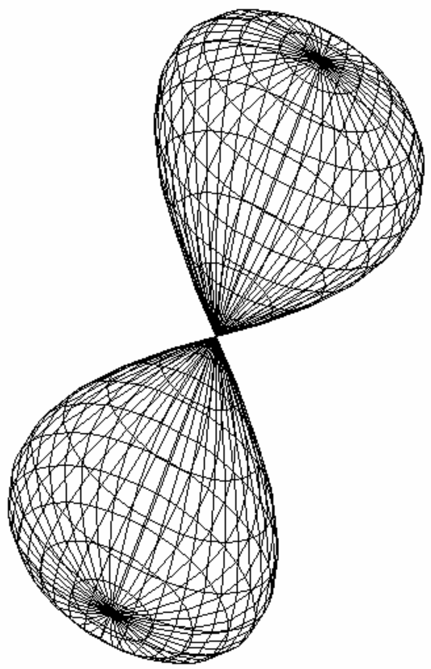


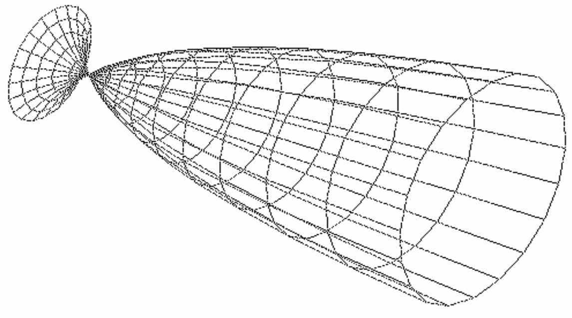


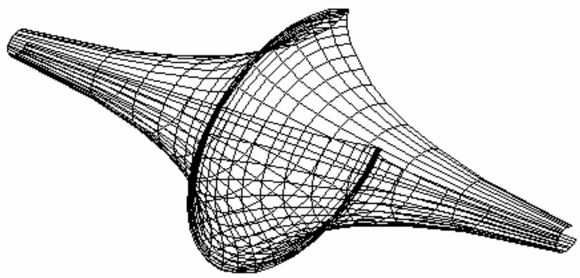


In this collection you will find a rotated lemniscate, three quarters of a pseudosphere and another rotated curve. There exists a simple formula for producing a solid of revolution using any curve given in parameter form [x(t), y(t)]:

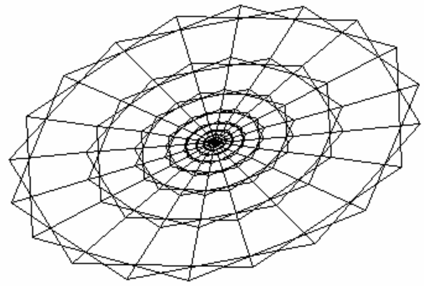
$$F(t,\varphi) = [x(t)\cos\varphi,\ x(t)\sin\varphi,\ y(t)]$$



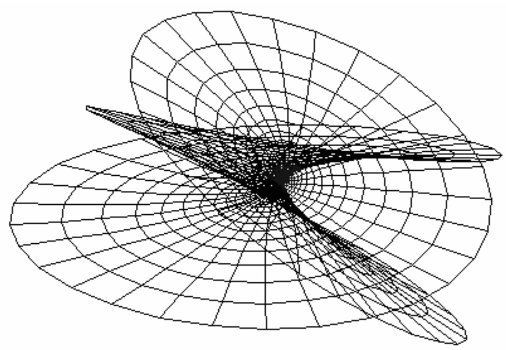


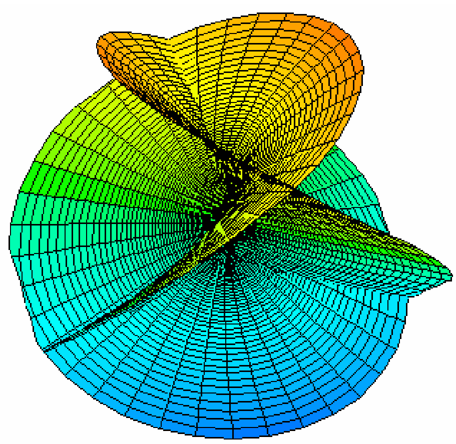


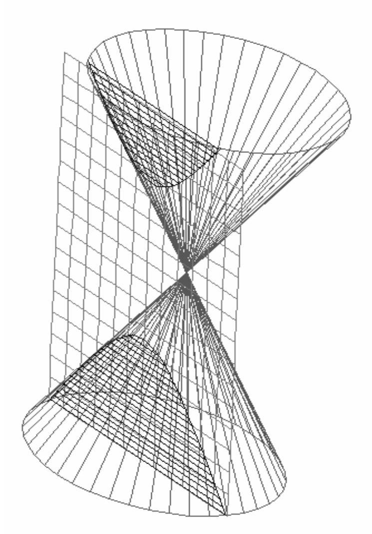
The next example uses the zooming abilities of ACROSPIN:
I produced the **Henneberg Surface** in different scales:
I took another scale and zoomed in. Observe the dark spot in the centre of the graph at the right and what is hidden in it:



$$\left[2\cdot\text{SINH}(s)\cdot\text{COS}(t) - \frac{2}{3}\cdot\text{SINH}(3\cdot s)\cdot\text{COS}(3\cdot t),\right.$$
$$2\cdot\text{SINH}(s)\cdot\text{SIN}(t) + \frac{2}{3}\cdot\text{SINH}(3\cdot s)\cdot\text{SIN}(3\cdot t),$$
$$\left.2\cdot\text{COSH}(2\cdot s)\cdot\text{COS}(2\cdot t)\right]$$







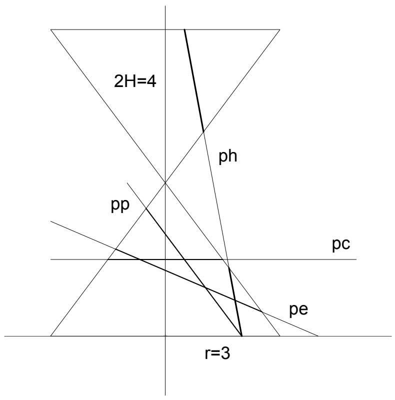
It is obvious that it cannot be too difficult to produce presentations of intersecting surfaces together with the intersection curves, tangents, normals, binormals, osculating planes, ... You can also add labels as you will see in one of the next examples.

I teach in a secondary school and I have some models of the conics sections. But using a tool like *DERIVE* it is convenient to produce a double cone, the intersecting plane, then to calculate the intersecting curve as a space curve and a bit more ambitious to shade the intersecting surface. All this is hard to do by hands only, but let *DERIVE* do the calculations. The students have to know the strategy to obtain results which can be used to be represented by *ACROSPIN*.

You can turn the model round and observe it from all directions. The different colours enforce the imagination. Switch off and on the layers. Compare the hyperbolic section with the parabolic and the other ones ...

I want to represent a double cone, the intersecting planes, the intersection curves and I will try to add a shading for the intersection figure to make the picture more impressive.

You can find the whole calculation in CONICS.MTH. The various CONIC*.ACD files show the different conic sections.



I want to show the start of the "parabolic" part to give some comments on it.

#3: $\text{cone} := [3\cdot t\cdot \text{COS}(\Phi),\ 3\cdot t\cdot \text{SIN}(\Phi),\ 4 - 4\cdot t]$

#4: $\text{VECTOR}\left[\text{VECTOR}\left[\text{cone},\ \Phi,\ 0,\ 2\cdot\pi,\ \frac{\pi}{20}\right],\ t,\ -1,\ 1,\ 2\right]$

#52: $\text{pp} := [-1,\ 0,\ 4] + u\cdot[3,\ 0,\ -4] + v\cdot[0,\ 1,\ 0]$

#54: $\text{sc\_3} := \frac{1}{\sqrt{([3,\ 0,\ -4]\cdot[3,\ 0,\ -4])}}$

#55: $\text{VECTOR}\left[\text{VECTOR}\left[\text{pp},\ u,\ 0,\ 1,\ \frac{\text{sc\_3}}{2}\right],\ v,\ -3,\ 3,\ 0.5\right]$

#57: $\text{SOLVE}(\text{pp} = \text{cone},\ [t,\ u,\ v])$

#58: $\left[\ t = \frac{1}{3\cdot(1 - \text{COS}(\Phi))}\quad u = \frac{1}{3\cdot(1 - \text{COS}(\Phi))}\quad v = \frac{\text{SIN}(\Phi)}{1 - \text{COS}(\Phi)}\ \right]$

#60: $[3\cdot t\cdot \text{COS}(\Phi),\ 3\cdot t\cdot \text{SIN}(\Phi),\ 4 - 4\cdot t]$

#61: $\left[3\cdot\frac{1}{3\cdot(1 - \text{COS}(\Phi))}\cdot\text{COS}(\Phi),\ 3\cdot\frac{1}{3\cdot(1 - \text{COS}(\Phi))}\cdot\text{SIN}(\Phi),\ 4 - 4\cdot\frac{1}{3\cdot(1 - \text{COS}(\Phi))}\right]$

#63: $\left[\frac{\text{COS}(\Phi)}{1 - \text{COS}(\Phi)},\ \frac{\text{SIN}(\Phi)}{1 - \text{COS}(\Phi)},\ \frac{4\cdot(3\cdot\text{COS}(\Phi) - 2)}{3\cdot(\text{COS}(\Phi) - 1)}\right]$

$$\#64:\ \text{SOLVE}\left[\frac{4\cdot(3\cdot\text{COS}(\Phi)-2)}{3\cdot(\text{COS}(\Phi)-1)}=0,\ \Phi\right]$$

$$\#65:\ \left[\Phi=\text{ASIN}\left[\frac{2}{3}\right]-\frac{\pi}{2},\ \Phi=\text{ASIN}\left[\frac{2}{3}\right]+\frac{3\cdot\pi}{2},\ \Phi=\frac{\pi}{2}-\text{ASIN}\left[\frac{2}{3}\right]\right]$$

$$\#66:\ [\Phi=-0.841,\ \Phi=5.44,\ \Phi=0.841]$$

$$\#67:\ \text{VECTOR}\left[\left[\frac{\text{COS}(\Phi)}{1-\text{COS}(\Phi)},\ \frac{\text{SIN}(\Phi)}{1-\text{COS}(\Phi)},\ \frac{4\cdot(3\cdot\text{COS}(\Phi)-2)}{3\cdot(\text{COS}(\Phi)-1)}\right],\ \Phi,\ 0.841,\ 5.44,\ \frac{5.44-0.841}{40}\right]$$

$$\#84:\ \text{VECTOR}\left[\begin{bmatrix} w & -\sqrt{(2\cdot w+1)}\cdot\text{SIGN}(w+1) & \frac{4\cdot(2-w)}{3} \\ w & \sqrt{(2\cdot w+1)}\cdot\text{SIGN}(w+1) & \frac{4\cdot(2-w)}{3} \end{bmatrix},\ w,\ -0.5,\ 2,\ 0.05\right]$$

In line #3 cone is the parameter form of the double cone, it is intersected by the plane pp in #52. Approximating #4 returns the family of lines building the cone, approximating #55 gives the grid of the plane. #54 is necessary to obtain a scaling factor to have a 1 by 1 grid. We intersect cone and plane in line #57, substitute for *t* in the cone's parameter form and find a "space curve" in #63, which is our conic section – a parabola. As we don't want to plot a parabola coming from anywhere and leaving for anywhere we have to find the parameter values which represent the intersection points of the parabola with the base.

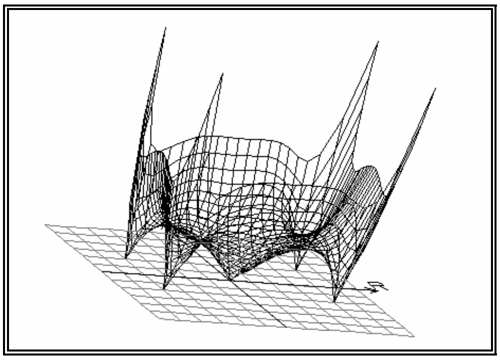
The shading should run normal to the parabola's axis, so we reparametrize the curve to end with the shading's vector in #84. We then approximate #4, #55, #67 and #84, save in different BAS-files, run ACD, ... and so on.

The next plot shows one of the favourite examples of Bert Waits. So it is an honour for me to dedicate this *DERIVE* – **ACD** – *ACROSPIN*-product Bert and his "Power of Visualization". It is a real representation of the complex roots of a 5th order equation. I used DERIVE to find the modulo surface of

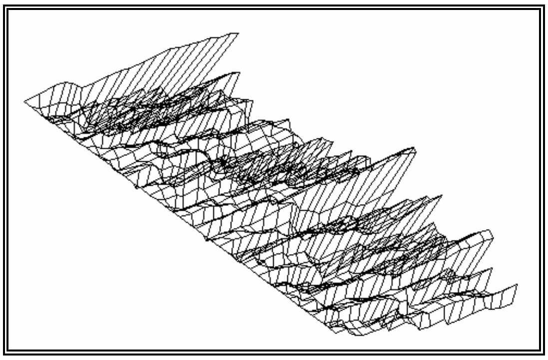
$$z^5 - 6z^3 + 25z = 0.$$

The peaks are in the complex plane at the positions of the five solutions. I added the complex plane, the axes and their names "R" and "I" – both letters as special "space curves", given by a list of points. (Look at the file BERT.MTH on the diskette!!). The "R" can be seen in this position. The other graph shows another artificial range: a "fractal landscape" generated by a recursive algorithm in *DERIVE*, which I will present in one of the next DNLs.

Bert´s Complex Range



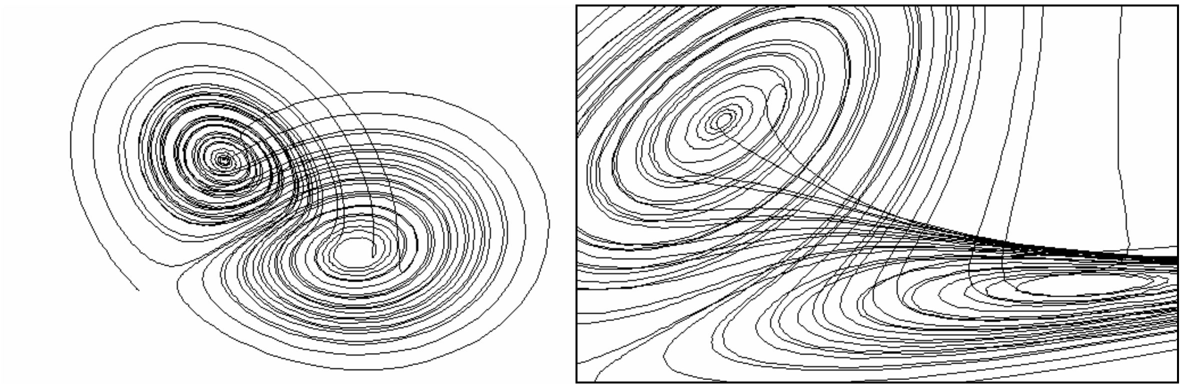
The Fractal Range



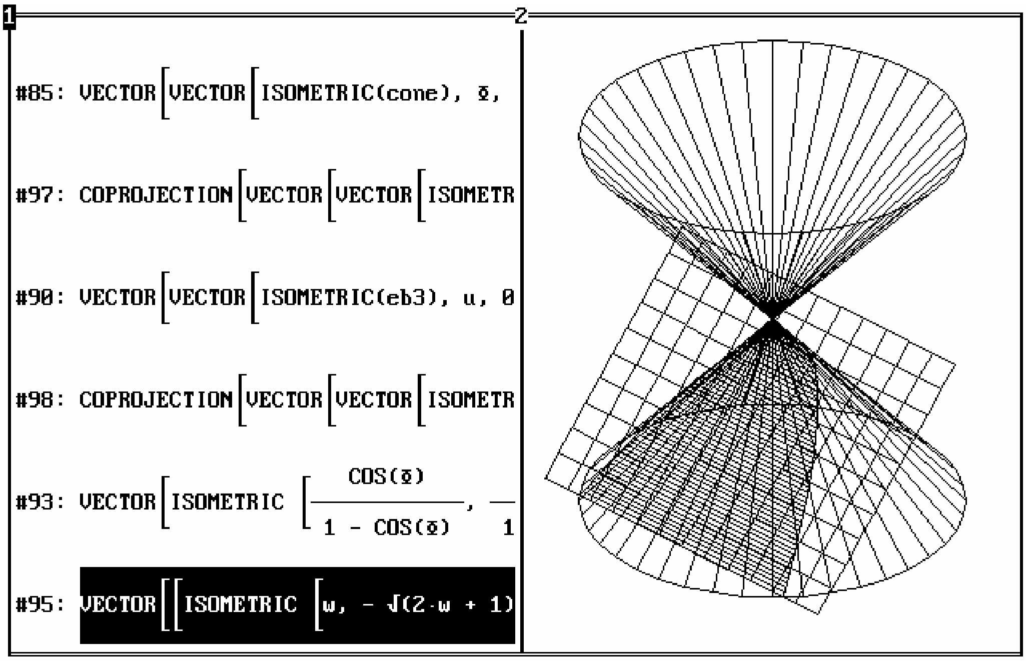
From fractals it is not very far to chaotic behaviour. Josef Lechner was the first - besides Richard Schorn - who checked ACD. He immediately tried successfully to visualize and to animate dynamic systems of three variables. During a phone call he mentioned the possibility now to animate the LORENZ attractor. So I will finish my interaction between *DERIVE* and *ACROSPIN* showing the famous LORENZ-attractor together with a zoom in from another "look out". The list of points was generated with Josef Lechner's *DERIVE* - tool INTEGRAPH.(Proceedings of the *DERIVE* Days Düsseldorf).

The following system of differential equations is the base of this attractor:

$$\frac{dx}{dy} = 10(y-x), \quad \frac{dy}{dt} = 28x - y - xz, \quad \frac{dz}{dt} = xy - \frac{8z}{3}$$



At last I want to give an impression how to work without *ACROSPIN*. I produce a "static" representation of the parabolic intersection of the double cone. As you can learn from the screen shot the trick is to to include ISOMETRIC and COPROJECTION from GRAPHICS.MTH at the appropriate place.
Please compare with the according file on pages 31 and 32. Don't forget to use COPROJECTION, other wise you would see only one family of parameter lines.



Among the downloadable files you can find in <ACD> all the MTH–files mentioned in this contribution accompanied by ACD.EXE and a self extracting compressed file ACDZIP.EXE containing a lot of ready made ACD–files. And you will also find *ACROSPIN*!!

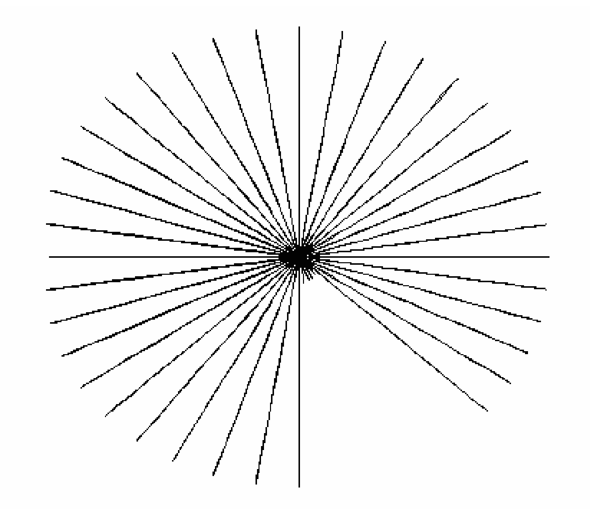
I would like to ask you to produce your own ACDs. And to enforce this "CALL for ACDs" I invite you for a **competition**

The ACD of the Year.

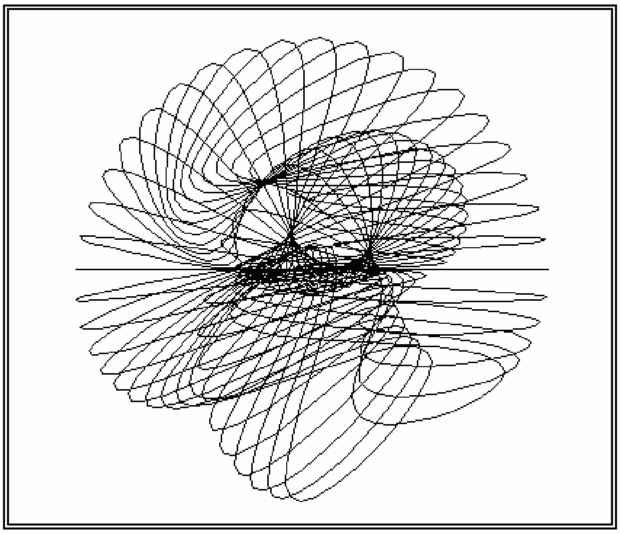
The "ACD of the Year" will win one year free DUG membership. Deadline is 31 May 1997. Much luck.

I am glad to announce another highlight for the next DNL. Our friend Sergey Biryukow from Moscow has produced a *DERIVE* tool to produce plots of implicit 3D functions. He saw ACD at Bonn, so he took care that his output is ACD compatible.

Last question: Which object is hidden in this star?



It is the *Bottle of Klein*!!



References:

[1] 3D-Programmierung mit BASIC, Glaeser, hpt 1986, ISBN 3 209 00626 1

[2] Atlas mathematischer Bilder, Leo H.Klingen, Addison-Wesley 1996, ISBN 3 89319 947 0

[3] Differentialgeometrie, Alfred Gray, Spektrum Akademischer Verlag, ISBN 3 86025 141 4

[4] Modern Differential Geometry of Curves and Surfaces, A.Gray, CRC Press,Inc, Boca Raton

[5] Computer Graphics, F.S.Hill Jr., Macmillan Publishing Company 1990, ISBN 0 02 354860 6

[6] DERIVE Days Düsseldorf Tagungsband, Bärbel Barzel, ed., Landesmedienentrum Rheinland--Pfalz

[7] DERIVE News Letter #21, C60-The Buckyball, Richard Schorn

Solving third-order linear differential equations with constant coefficients

Rodriguez Contreras, J.L.
Department of Mathematics, University of Norte and University of Atlantico
Baranquilla, Colombia

Fernández Gutiérrez, M.J.
Department of Mathematics, Universoty of Oviedo, 33071 Oviedo, Spain
E-mail: mjfg@pinon.ccu.uniovi.es

Abstract

In this work the general solution, or a particular one, to linear ordinary differential equations, homogeneous or non-homogeneous, of third order with constant coefficients, is given, using *DERIVE*. A file with the appropriate *DERIVE* functions is included.

1. INTRODUCTION

DERIVE cannot be considered a programming language. However, new functions that make use of the operations and functions included in the program can be defined. In the program handbook it is said that *DERIVE* is not more than a collection of mutually recursive functions.

We believe that (with the available information) differential equations have not been explored adequately.

In the hand book, we are informed that *DERIVE* contains three files ODE1.MTH, ODE2.MTH and ODE_APPR.MTH to solve respectively ordinary differential equations of first, second-order and numerically. We have designed a file LODE3.MTH, to find the general solution, or a particular one, of an homogeneous or non-homogeneous linear differential equation of third-order with constant coefficients:

$$y''' + a\,y'' + b\,y' + c\,y = f(x)$$

2. DESCRIPTION OF THE FILE

Firstly the roots of the characteristic polynomial $x^3 + a\,x^2 + b\,x + c$ are determined. Considering that these roots appear in vectorial form where each element is an equation in the form x = "root", the RHS function that selects the right member of the equation has been used. This function RHS is only present in DERIVE 2.58 and later.

Next, the function **F(a,b,c)** has been defined to control the different kinds of roots that the characteristic polynomial can have. Thus, if F(a,b,c) > 0, the polynomial has a real root and two conjugate complex ones. If F(a,b,c) < 0, the polynomial presents three different real roots; and if F(a,b,c) = 0 it has only one real root with multiplicity three if and only if $3b = a^2$.

When the polynomial presents two real roots, one simple and the other of double multiplicity, a function **P(a,b,c)** has been defined, due to the fact that the SOLVE-function gives a vector with two components (one of them being the double multiplicity root). The sign of P(a,b,c) together with the relative magnitude of the vector components determine the position of the multiple/simple root in the vector. Taking into account the previous considerations about the roots, the function **SGH(a,b,c,x,c1,c2,c3)** calculates the general solution of the homogeneous equation.

As Y1(*a,b,c,x*), Y2(*a,b,c,x*) and Y3(*a,b,c,x*) functions, defined the program, are linear independent solutions of the homogeneous linear differential equation, it can be assured that if the parameter variation methods for finding a particular solution of the non-homogeneous equation has been used, then the corresponding equation system has only one solution which can be obtained by means of Cramer´s rule.

Then a function **SGC(a,b,c,x,f,c1,c2,c3)** has been defined. It calculates the general solution of non-homogeneous differential equations.

Finally, the functions **SPH(a,b,c,x,x0,y01,y02,y03)** and **SPC(a,b,c,x,f,x0,ly01,y02,y03)** have been defined to calculate a particular solution to both homogeneous and non-homogeneous equations, respectively.

It is just necessary to remember the names and arguments of the functions SGH, SGC, SPH and SPC in their right order.

3. FILE LISTING

```
"File LODE3.MTH"
RPC(a,b,c):=SOLVE(x^3+a*x^2+b*x+c,x)
RCP(a,b,c):=VECTOR(RHS(ELEMENT(RPC(a,b,c),k)),k,1,3)
T1(a,b,c,x):=EXP(ELEMENT(RCP(a,b,c),1)*x)
T2(a,b,c,x):=EXP(ELEMENT(RCP(a,b,c),2)*x)
............
............
```

(As the file takes more than two pages and fortunately I can add the Diskette of the Year to this issue you can find the file LODE3.MTH on the diskette in subdirectory <MTH24>. So we will go on immediately to the examples. The paper submitted had included 8 pages with 81 examples. I´ll try to give a selection of some typical examples. Josef)

4. EXAMPLES

A 486 DX4/75 has been used in the solutions of the differential equations below. The file has been optimized in order to get

a) the solution to all third-order linear differential equations with constants of [1],[2] and

b) its minimum execution time.

I changed the original file for its use with later DERIVE versions. The first two expressions had to be adapted because of distinguishing between SOLVE and SOLUTIONS. Josef (2010). The first expressions read now:

```
#1:   File LODE3_V6.MTH

                                   3      2
#2:   RCP(a, b, c) := SOLUTIONS(x  + a·x  + b·x + c, x)

      T1(a, b, c, x) := EXP((RCP(a, b, c)) ·x)
#3:                                       1

      T2(a, b, c, x) := EXP((RCP(a, b, c)) ·x)
#4:                                       2
```

1) $y''' - 2y'' - 5y' + 6y = 0$; **Check the solution!**

```
#33: SGH(-2, -5, 6) = c3·e^(3·x) + c1·e^x + c2·e^(- 2·x)

#34: y(x) := c3·e^(3·x) + c1·e^x + c2·e^(- 2·x)

#35: y'''(x) - 2·y''(x) - 5·y'(x) + 6·y(x) = 0
```

2) $y''' - y'' = -2y$; **Check the solution!**

```
#37: SGH(-1, 0, 2) = e^x·(c2·COS(x) + c3·SIN(x)) + c1·e^(-x)

#38: y(x) := e^x·(c2·COS(x) + c3·SIN(x)) + c1·e^(-x)

#39: y'''(x) - y''(x) + 2·y(x) = 0
```

3) $y = \dfrac{y''' + 5y'' + 5y'}{11}$

```
#41: SGH(5, 5, -11) = c1·e^x + e^(- 3·x)·(c2·COS(√2·x) + c3·SIN(√2·x))
```

4) $y''' - 4y'' + 7y' = 6y$; $y(0)=1,\ y'(0)=y''(0)=0$

Find the solution and check the result!

```
#43: SPH(-4, 7, -6, x, 0, 1, 0, 0) = e^(2·x) - √2·e^x·SIN(√2·x)

#44: y(x) := e^(2·x) - √2·e^x·SIN(√2·x)

#45: y'''(x) - 4·y''(x) + 7·y'(x) - 6·y(x) = 0

#46: [y(0), y'(0), y''(0)] = [1, 0, 0]
```

5) $y''' - 3y'' + 4y = x\,e^{2x} - \cos x$; **Check the result!**

#48: SGC(-3, 0, 4, x, x·e^(2·x) - COS(x))

#49: $\dfrac{e^{2\cdot x}\cdot(9\cdot x^3 - 9\cdot x^2 + 6\cdot x\cdot(27\cdot c2 + 1) + 162\cdot c1 - 2)}{162} + c3\cdot e^{-x} - \dfrac{7\cdot COS(x)}{50} + \dfrac{SIN(x)}{50}$

#50: $y(x) := \dfrac{e^{2\cdot x}\cdot(9\cdot x^3 - 9\cdot x^2 + 6\cdot x\cdot(27\cdot c2 + 1) + 162\cdot c1 - 2)}{162} + c3\cdot e^{-x} - \dfrac{7\cdot COS(x)}{50} + \dfrac{SIN(x)}{50}$

#51: y'''(x) - 3·y''(x) + 4·y(x)

#52: $-\dfrac{e^{2\cdot x}\cdot(9\cdot x^2 - 15\cdot x + 2\cdot(27\cdot c2 - 2))}{27} + \dfrac{e^{2\cdot x}\cdot(9\cdot x^2 + 12\cdot x + 2\cdot(27\cdot c2 - 2))}{27} - COS(x)$

#53: needs resimplifying #52

#54: $x\cdot e^{2\cdot x} - COS(x)$

6) $y''' - 2y'' + 5y' = -24e^{3x}$; $y(0)=4,\ y'(0)=-1,\ y''(0)=5$; **Check the initial conditions!**

#56: $\text{SPC}(-2,\ 5,\ 0,\ x,\ -\ 24\cdot e^{3\cdot x},\ 0,\ 4,\ -1,\ 5)$

#57: $\text{SPC}(-2,\ 5,\ 0,\ x,\ -\ 24\cdot e^{3\cdot x},\ 0,\ 4,\ -1,\ 5) = -\ e^{3\cdot x} + e^{x}\cdot(2\cdot\text{SIN}(2\cdot x) - 2\cdot\text{COS}(2\cdot x)) + 7$

#58: $y(x) := -\ e^{3\cdot x} + e^{x}\cdot(2\cdot\text{SIN}(2\cdot x) - 2\cdot\text{COS}(2\cdot x)) + 7$

#59: $[y(0),\ y'(0),\ y''(0)] = [4,\ -1,\ 5]$

7) $y''' - 2y'' - 3y' + 10y = e^{-2x}(34x-16) - 10x^2 + 6x + 34$
$y(0)=3,\ y'(0)=y''(0)=0$

#61: $\text{SPC}(-2,\ -3,\ 10,\ x,\ e^{-\ 2\cdot x}\cdot(34\cdot x - 16) - 10\cdot x^2 + 6\cdot x + 34,\ 0,\ 3,\ 0,\ 0)$

#62: $x^2\cdot e^{-\ 2\cdot x} - x^2 + 3$

8) **Find the general solutions:**

$y''' + 5y'' + 26y' - 150y = 20e^{-x}$

$y''' + 5y'' + 26y' - 150y = 600x^3$

$y''' + 4y' = \dfrac{4}{\tan 2x}$

#64: $\text{SGC}(5,\ 26,\ -150,\ x,\ 20\cdot e^{-x})$

#65: $c1\cdot e^{3\cdot x} - \dfrac{5\cdot e^{-x}}{43} + e^{-\ 4\cdot x}\cdot(c2\cdot\text{COS}(\sqrt{34}\cdot x) + c3\cdot\text{SIN}(\sqrt{34}\cdot x))$

#66: $\text{SGC}(5,\ 26,\ -150,\ x,\ 600\cdot x^3)$

#67: $c1\cdot e^{3\cdot x} + e^{-\ 4\cdot x}\cdot(c2\cdot\text{COS}(\sqrt{34}\cdot x) + c3\cdot\text{SIN}(\sqrt{34}\cdot x)) - \dfrac{4\cdot(140625\cdot x^3 + 73125\cdot x^2 + 53475\cdot x + 19769)}{140625}$

#68: $\text{SGC}(5,\ -26,\ -150,\ x,\ 600\cdot x^3)$

#69: $\text{SGC}(5,\ -26,\ -150,\ x,\ 20\cdot e^{-x})$

#70: Memory Exhausted

#72: $\text{SGC}\left(0,\ 4,\ 0,\ x,\ \dfrac{4}{\text{TAN}(2\cdot x)}\right)$

#73: $\dfrac{\text{LN}(\text{SIN}(2\cdot x))}{2} - \dfrac{\text{COS}(2\cdot x)\cdot\text{LN}(\text{TAN}(x))}{2} + c2\cdot\text{COS}(2\cdot x) + c3\cdot\text{SIN}(2\cdot x) + c1 - \dfrac{1}{2}$

#74: $y(x) := \dfrac{\text{LN}(\text{SIN}(2\cdot x))}{2} - \dfrac{\text{COS}(2\cdot x)\cdot\text{LN}(\text{TAN}(x))}{2} + c2\cdot\text{COS}(2\cdot x) + c3\cdot\text{SIN}(2\cdot x) + c1 - \dfrac{1}{2}$

#75: $y'''(x) + 4\cdot y'(x) = \text{COS}(2\cdot x)\cdot\left(\dfrac{4}{\text{SIN}(2\cdot x)} + \dfrac{8}{\text{SIN}(2\cdot x)^3}\right) - \dfrac{2}{\text{SIN}(x)^3\cdot\text{COS}(x)} + \dfrac{1}{\text{SIN}(x)^3\cdot\text{COS}(x)^3}$

#76: Simplify the difference of both right sides:

#77: $\cos(2\cdot x)\cdot\left(\frac{4}{\sin(2\cdot x)}+\frac{8}{\sin(2\cdot x)^3}\right)-\frac{2}{\sin(x)^3\cdot\cos(x)}+\frac{1}{\sin(x)^3\cdot\cos(x)^3}-\frac{4}{\tan(2\cdot x)}$

#78: $\frac{8\cdot\cos(2\cdot x)}{\sin(2\cdot x)^3}-\frac{2}{\sin(x)^3\cdot\cos(x)}+\frac{1}{\sin(x)^3\cdot\cos(x)^3}$

#79: Trigonometry := Collect

#80: Trigpower := Sines

#81: 0

#82: Result from 1996:

#83: $-\frac{\cos(2\cdot x)\cdot\ln(\tan(x))}{4}+\frac{\ln(\sin(2\cdot x))}{2}+\frac{\cos(4\cdot x)}{4}+\frac{\cos(2\cdot x)\cdot\left(2\cdot c2-\int\frac{\cos(4\cdot x)}{\sin(2\cdot x)}\,dx\right)}{2}+c3\cdot\sin(2\cdot x)+c1-\frac{1}{4}$

#84: simplified:

#85: $\frac{\ln(\sin(2\cdot x))}{2}-\frac{\cos(2\cdot x)\cdot\ln(\tan(x))}{2}+c2\cdot\cos(2\cdot x)+c3\cdot\sin(2\cdot x)+c1-\frac{1}{2}$

9) $y''' - 3y'' + 2y' = \frac{e^{3x}}{1+e^{3x}}$

a) Find a particular solution

b) Find the solution with $y(1)=2,\ y'(1)=3,\ y''(1)=4$

#87: $\mathrm{SP}\left(-3,\ 2,\ 0,\ x,\ \frac{e^{3\cdot x}}{1+e^x}\right)$

#88: $e^{2\cdot x}\cdot\left(\frac{\ln(e^x+1)}{2}-\frac{3}{4}\right)+e^x\cdot\left(\ln(e^x+1)-\frac{1}{2}\right)+\frac{\ln(e^x+1)}{2}$

#89: $\mathrm{SPC}\left(-3,\ 2,\ 0,\ x,\ \frac{e^{3\cdot x}}{1+e^x},\ 1,\ 2,\ 3,\ 4\right)$

#90: $\frac{\ln\left(\frac{e^x+1}{e+1}\right)}{2}+e^{2\cdot x}\cdot\left(\frac{\ln\left(\frac{e^x+1}{e+1}\right)}{2}+\frac{e^{-2}}{2}-\frac{3}{4}\right)+e^x\cdot\left(\ln\left(\frac{e^x+1}{e+1}\right)+e+2\cdot e^{-1}-\frac{1}{2}\right)-\frac{e^2}{4}+\frac{e}{2}-\frac{1}{2}$

5. REFERENCES

[1] Bronson, R. (1989). 2500 Solved Problems in Differential Equations. McGraw-Hill

[2] Kent, R. Saff, E.B. (1992). Fundamentos de Ecuaciones Diferenciales. Addison-Wesley Iberoamericana

[3] Soft Warehouse (1992). "DERIVE User Manual"

This file shows once more the power of a CAS in the hands of an experienced user and it raises once more the question about the necessity of so many drill examples in the text books for the future. And the future may have still begun. Josef

Ebene Algebraische und Transzendente Kurven (9)

Thomas Weth, Würzburg, Germany

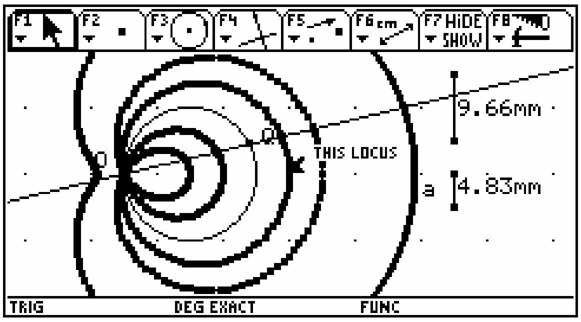
Pascalsche Schnecken – Snails of Pascal

„Wenn alle die Kräfte, die bei der Entwickelung einer Pflanze mitwirken, mathematisch erkannt wären und ebenso der innere Mechanismus ihrer Organe, so würde man im stande sein die ganze Lebensentwickelung durch Formeln darzustellen, insbesondere würde man die Gleichungen derjenigen Kurven erhalten können, welche den Umriß ihrer Blätter darstellen. Aber umgekehrt, wenn man auch diese Gleichungen kennte, würde man dennoch nicht das Leben jener Pflanze durch Formeln darstellen können; doch auch von diesem Ziele ist man noch weit entfernt indem man sich begnügen muß, die Blattumrisse durch Gleichungen darzustellen, die nicht exakt sondern nur in einfacher Weise angenähert diese wiedergeben“ (Loria, 1902, S.307). Bodo Habenicht versuchte ausgangs des 19. Jhdts. dem Geist der Zeit wissenschaftlicher und technischer Höchstleistungen (Röntgenstrahlen, Eiffelturm) entsprechend auch die Botanik durch die Sprache der Mathematik zu erfassen (vgl. *Die analytische Form der Blätter*, Quedlinburg, 1895). Eine der einfachsten Kurven, die sich in modifizierter Form unter den von Habenicht angegebenen „Blattkurven“ findet, ist die Pascalsche Schnecke, die auf Etienne Pascal, den Vater des bekannten Blaise Pascal, zurückgeht. Wie sich herausstellt, können spezielle Pascalsche Schnecken zur Dreiteilung des Winkels verwendet werden und spielen bei mechanischen Problemen ein Rolle, weswegen sie ursprünglich auch unter dem Namen "Sauveur's und de l'Hospital's Zugbrücke" bekannt waren. Bereits vorher hatten sich auch Johann und Jacob Bernoulli mit dieser Kurve beschäftigt.

"If all the forces which are contributing to the evolution of a plant would be recognized mathematically and also their internal mechanism of their organs then we were enabled to represent the whole process of its life by formulae. We specially would obtain the equations for the curves which form the contour of their leaves. But reversely, if we even knew the equations, we would yet be unable to express the plant’s life by formulae. This goal is very far as we have to be satisfied with equations which are reproducing the leaves’ contours only in a very approximative way. (Loria, 1902, p.307). At the end of the 19th century Bodo v. Habenicht inspired by scientific and technical supreme achievements (X-rays, Eiffel tower) tried to describe botany by the mathematical language. (*The Analytical Form of Leaves*, Quedlinburg, 1895). One of the simplest curves which can be found among Habenicht´s “Leave Curves" is the *Snail of Pascal*, which goes back to Etienne Pascal, father of the well known Blaise Pascal. It appears that special Snails of Pascal can be used for the trisection of an angle and they play a role in several mechanic problems. For that reason they were called "Sauveur’s and de l´Hospital’s Drawbridge". Before that time Jacob and Johann Bernoulli had dealt with this curve, too.

Konstruktion – Construction

Genauso wie die Konchoiden des Nikomedes (vgl. Folge 5) lassen sich Pascalsche Schnecken konstruieren - nur verwendet man nicht wie dort eine Gerade, sondern einen Kreis als Leitlinie. Kreiskonchoiden ergeben sich nach folgender Konstruktionsvorschrift:
Gegeben sind ein Kreis mit Mittelpunkt M und Durchmesser b und ein Punkt O auf der Kreislinie. Von einem Kreispunkt Q aus trägt man auf der Geraden OQ in die beiden möglichen Richtungen jeweils eine Strecke konstanter Länge a ab; die Endpunkte P_1 und P_2 dieser Strecken sind dann Konchoidenpunkte zum gegebenen Kreis.



Pascalsche Schnecken als Kreiskonchoiden für drei verschiedene Abstände a, erstellt mit dem TI 92.

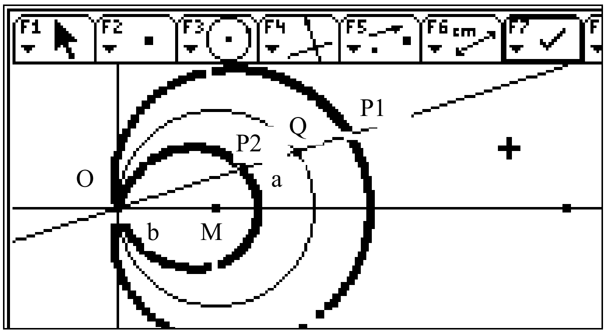
The snails can be constructed similar to the Conchoids of Nikomedes (Lexicon #5) - with a circle as directrix instead of a line.

Given is a circle with centre M and diameter b and a point O on the circle line. Take any other point Q on the circle line, draw the line OQ and find the two points P_1 and P_2 with $QP_1 = QP_2 = a$ = const. Then P_1 and P_2 are two points of the curve.

Herleitung der Kurvengleichung – Dervivation of the equation of the curve

Aus der Konstruktion ergibt sich für P_1 und P_2:
$OP_1 = OQ + QP_1$ bzw. $r = b \cos \varphi + a$ und
$OP_2 = OQ + QP_2$ bzw. $r = b \cos \varphi - a$.
Lässt man für *r* negative Werte zu, vereinfacht sich die Darstellung zur allgemeinen Polardarstellung Pascalscher Schnecken:

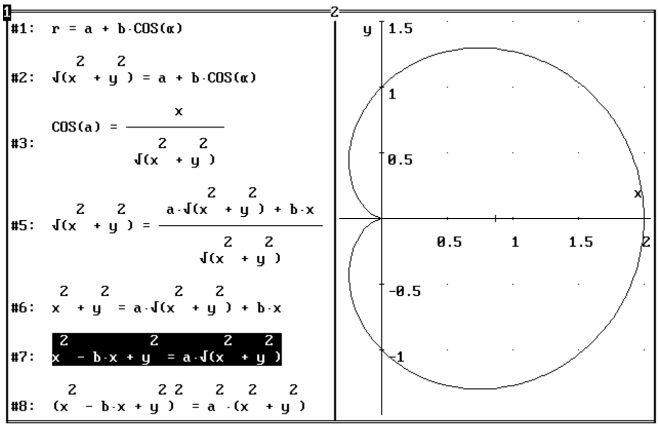
$$r = b \cos \varphi + a.$$



This is the polar form. See the construction.

Mit $r = \sqrt{x^2 + y^2}$ berechnet man (mit DERIVE) sofort:
$a^2(x^2 + y^2) - (x^2 + y^2 - b \cdot x)^2 = 0$
Damit sind Pascalsche Schnecken symmetrische algebraische Kurven vierter Ordnung.

Für $a = b$ erhält man aus der Kurvenschar wie in der nebenstehenden Abbildung die Kardioide (vgl. Folge 7).



We substitute for $r = \sqrt{x^2 + y^2}$ then we can see that these curves are symmetric algebraic curves of order 4. For *a* = *b* we obtain the Cardioid. Compare with Lexicon #7.

Weitere Erzeugungsweisen

Es sei noch erwähnt, dass Pascalsche Schnecken sich auch durch andere Konstruktionen erhalten lassen:

- als *Rollkurve*

Rollt ein Kreis außen auf einem festen Kreis ab, so beschreibt ein markierter Punkt auf dem rollenden Kreis eine Pascalsche Schnecke.

- *als Inversionskurve*

Bildet man einen Kegelschnitt durch eine Inversion an einem Kreis ab, dessen Mittelpunkt mit dem Brennpunkt des Kegelschnitts zusammenfällt, so erhält man Pascalsche Schnecken.

- *als Ortslinie merkwürdiger Dreieckspunkte*

Betrachtet man zu einem gegebenen Kreis Sehnendreiecke, deren eine Ecke A festliegt und bei dem der zugehörige Winkel α konstant ist, so ist der Ort der In- und Ankreismittelpunkte aller derartigen Dreiecke eine Pascalsche Schnecke.

- *als Ortslinie eines Winkelscheitels*

Bewegt sich ein konstanter Winkel so, dass seine Schenkel zwei feste Kreise ständig berühren, so beschreibt sein Scheitelpunkt eine Pascalsche Schnecke.

Other ways to obtain a Cardioid

It is worth to be mentioned that we can obtain a Snail of Pascal in some other ways:

- as *Trochoid*

A point on a circle rolling outside on a fixed other circle describes a Snail of Pascal

- *as a Curve of Inversion*

If we map a conic by an inversion at a circle with its centre lying in a focal point of the conic, then we again receive a Snail of Pascal.

- *as locus of remarkable points of a triangle*

Observing triangles inscribed in a given circle with a fixed vertex A and the accompanying constant angle α, then the locus of the centres of all the incircles and excircles results in a Snail of Pascal.

- *as locus of a vertex of an angle*

A constant angle moving with its sides touching two fixed circles gives a Snail of Pascal as the locus of its vertex.

Dreiteilung des Winkels mit Pascalschen Schnecken - Angle trisection using S. of P.

Gegeben sei eine spezielle Pascalsche Schnecke mit $a = b/2$.
Den zu drittelnden Winkel[1] $\omega = \angle SMP$ trägt man wie in nebenstehender Zeichnung an. P sei der Schnittpunkt des einen Schenkels von ω mit der Pascalschen Schnecke. Dann gilt:

Da das Dreieck QOM gleichschenklig ist: $\angle MOQ = \angle MQO =: \varphi$.

Der Winkel $\angle QMT$ ist Außenwinkel zum Dreieck QOM, hat also die Größe 2φ. Im gleichschenkligen Dreieck PMQ gilt für die Basiswinkel: $\angle QPM = \angle QMP = \frac{180° - \varphi}{2} = 90^0 - \frac{\varphi}{2}$..

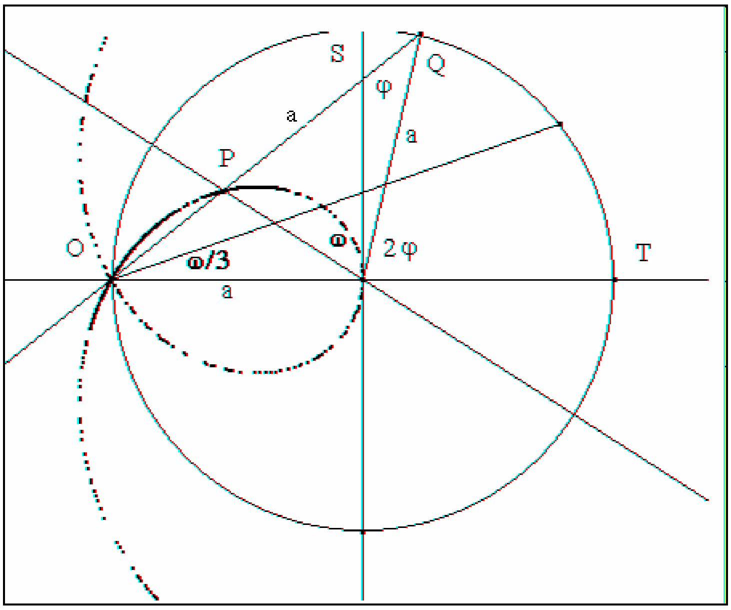
Für den Winkel $\angle TMP$ gilt also: $\angle TMP = \omega + 90^0$ und weiters

$$\angle TMP = \angle QMT + \angle QMP = 2\varphi + \left(90^0 - \frac{\varphi}{2}\right) = 90^0 + \frac{3\varphi}{2}.$$

Daraus folgt: $\omega = \frac{3\varphi}{2}$ bzw: $\frac{1}{3}\omega = \frac{1}{2}\varphi$..

[1] Im folgenden wird nicht unterschieden zwischen Winkel und Winkelmaß. Die jeweilige Bedeutung ergibt sich aus dem Kontext. In the following we will not differ between angle and its measure.

Halbiert man also den Winkel $\angle MOQ = \varphi$, so erhält man ein Drittel des gegebenen Winkels ω.



Let´s take a special Snail of Pascal with $a = b/2$.

$\omega = \angle SMP$ is the angle to be trisected. P is the intersection point of one of its leg with the curve. The following can be deduced:

As ΔQOM is isosceles we find $\angle MOQ = \angle MQO = \varphi$. $\angle QMT$ is an exterior angle of the triangle QOM, hence equals 2φ. ΔPMQ is another isosceles triangle with $\angle QMP = \angle QPM = = 90° - \frac{\varphi}{2}$.

$\angle TMP = \omega + 90°$ and $\angle TMP = \angle QMT + \angle QMP = 2\varphi + \left(90° - \frac{\varphi}{2}\right) = 90° + \frac{3\varphi}{2}$.

This leads obviously to $\omega = \frac{3\varphi}{2}$, $\omega = \frac{3\varphi}{2}$, and to $\frac{\omega}{3} = \frac{\varphi}{2}$.

So if we bisect the angle $\angle MOQ = \varphi$ we obtain a third of the given angle ω.

Anmerkung zum WorldWideWeb – Interesting WWW - pages

Highly recommended:

The MacTutor History of Mathematics archive

http://turnbull.mcs.st-and.ac.uk/~history/

Index of Biographies: http://turnbull.mcs.st-and.ac.uk/history/BiogIndex.html

History Topic Index: http://turnbull.mcs.st-and.ac.uk/history/Indexes/HistoryTopics.html

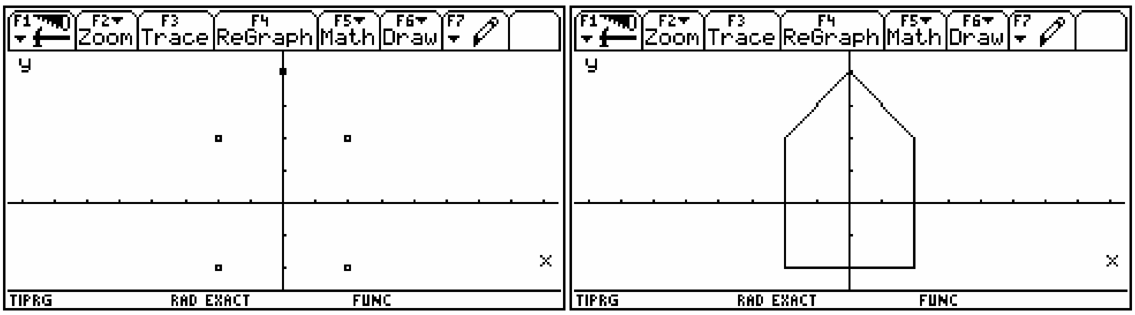
Famous Curves Index: http://turnbull.mcs.st-and.ac.uk/history/Curves/Curves.html

Mathmeticians of the Day: http://turnbull.mcs.st-and.ac.uk/history/Day_files/Now.html

Additional Matrials: http://turnbull.mcs.st-and.ac.uk/history/Indexes/Extras_index.html

****

Two screen shots of PLOMAT(house) from the TI-92 to plot matrices of points.
(left side: Points discrete and large; right side: Points Connected)

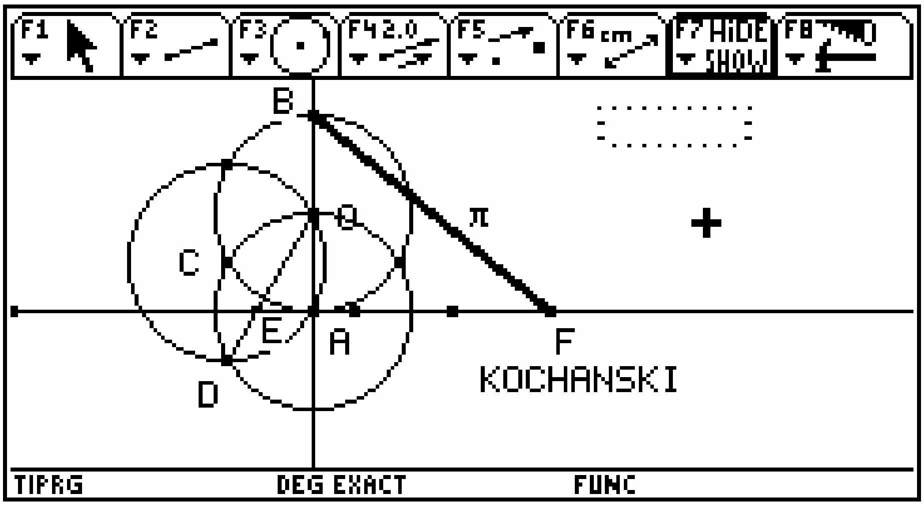


Two other approximations to square the circle

Alfonso J. Población Sáez, Valladolid, Spain

3. A method from Poland

This third method is due to the Polish Jesuit Reverend Adam Kochanski, in 1685. He was the first to use a steel spring in the suspension of a clock's pendulum.



Kochanski´s construction with a circle of radius r = 1.

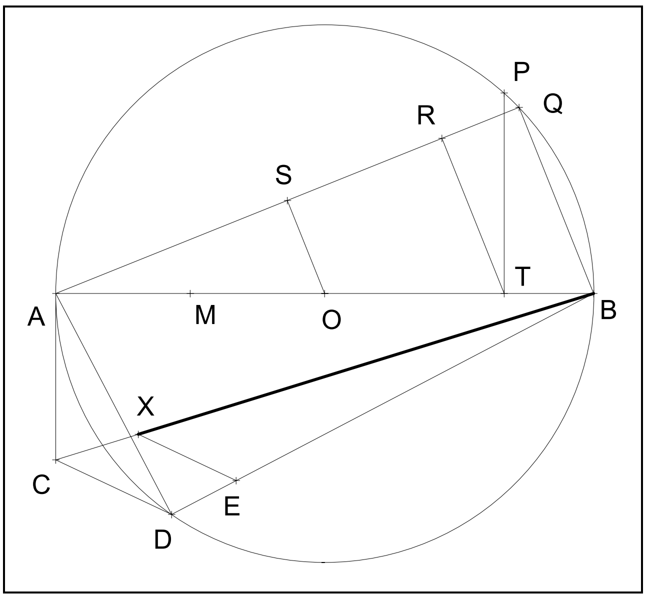
This reminded me on my school time - on the screen of a TI-92. Josef

Start with a circle (centre = O and radius r). Proceed with a circle of radius OA and centre A (on the diameter BOA in order to find C. With centre in C, draw one more circle having the same radius and obtain D. Consider the segment OD and its intersection with the tangent to the initial circle passing through A. This gives you point E. Then F is on this tangent and verifies that EF = 3 * OA. The length BF is approximately OA*π. What is here the value of π?

4. Ramanujan´s contribution

Finally, I chose the construction of a great Hindu mathematician. Srinivasa Ramanujan. He gave us a lot of interesting formulae, one of which is implemented in *DERIVE* to approximate π. (see [5]).

From a given circle, consider M, the midpoint of OA and T such that OT = $\frac{2}{3}$ OB. Then P is on the circumference such that TP is perpendicular to AB and Q is such that BQ = TP. Take S as the midpoint of AQ and D satisfies AD = AS. The segments TR, BQ and OS must be parallel. Now draw AC being tangent to the circumference and length equal to RS. Finally, BE = BM and X such that EX is parallel to CD. What is the relation between BX and π?



REFERENCES

[1] Boyer, C.B. Historia de la matemática. Alianza Editorial. Madrid, 1987
(English version: A History of Mathematics. John Wiley and Sons, Inc.)

[2] Gardner, M. Nuevos Pasatiempos Matemáticos. Alianza Editorial. Madrid, 1982
(English version: New Mathematical Diversions. Scientific American, N.Y.)

[3] Kline, M. El pensamiento matemático desde la Antigüedad a nuestros dias. Alianza Editorial. Madrid, 1992
(English version: Mathematical thought from ancient to modern times. Oxford University Press, Inc.)

[4] Proceedings of TEMU 95. Servicio de Publicaciones de la Universitat Politécnica de Catalunya. Barcelona, 1994

[5] Rich, A.D. and Stoutemyer, D.R. Inside the DERIVE Computer Algebra System.
The International DERIVE Journal, vol 1 number 1. April 1994

[6] Steinhaus, H. Instantáneas Matemáticas. Salvat Ediciones. Barcelona, 1989
(English version: Mathematical Snapshots. Oxford University Press, Inc. N.Y.)

Probability Distributions
Proof and Computations (1)

Peter Mitic, Medstead, UK

ABSTRACT

The use of *DERIVE* to perform relevant algebraic computations in the context of probability distributions is discussed, with particular reference to some common distributions. Problems with using *DERIVE* in this way are noted, and some partial solutions are suggested.

INTRODUCTION

Certain components of courses in probability and elementary statistics can benefit from the algebraic manipulation facilities offered by *DERIVE*. The way in which the *DERIVE* interface is used (i.e. select an expression and operate on it) makes it particularly suitable for computations which involve sequential operations. Consequently, the user can concentrate on wider aspects of problem solving and is free to appreciate the overall strategy used. In this paper we use *DERIVE* to prove certain results in probability theory and show how some shortcomings of *DERIVE* may be overcome.

We consider how *DERIVE* may be used to obtain some standard results in probability theory, and assess its efficacy in doing so. The initial discussion centres on the ability of *DERIVE* to calculate means and variances, given some standard probability distributions. This involves summing series and evaluating integrals. It is an advantage, from a didactic point of view, to be able to do such computations directly. Looking up standard results is not meaningful unless it is accompanied by a good conceptual understanding. Routine computations can help to provide this.

MEANS AND VARIANCES OF DISCRETE RANDOM VARIABLES

We consider the cases of the binomial and geometric distributions, because the principal pedagogic and technical points are covered by these distributions.

For a discrete random variable, *X*, defined in a domain, S, the definitions below may be used to calculate the mean and variance, μ and σ^2 respectively, of *X*.

$\mu = \sum_{s} x\, P(X = x)$ and $\sigma^2 = \sum_{s} x^2\, P(X = x) - \mu^2$, where $x \in$ S.

Applying these results to a Binomial(*n,p*) random variable, we use the *DERIVE* construct COMB(n,x) p^x $(1–p)^{n–x}$ for P(*X* = *x*) and sum over *x* from 0 to *n*. The random variable, *X*, might represent the sum of the scores obtained in *n* independent tosses of a die which has the probability *p* of landing *heads* on any one toss. In the discussion by Etchells (Etchells 1992), these computations are done by considering *n* = 1, then *n* = 2 and then *n* = 3 with results

$\mu_1 = p$, $\sigma_1^2 = p\,(1 – p)$,

$\mu_2 = 2p$, $\sigma_2^2 = 2p\,(1 – p)$ and

$\mu_1 = 3p$, $\sigma_3^2 = 3p\,(1 – p)$ respectively.

DERIVE has no problems in evaluating these sums and the general results $\mu = n\,p$ and $\sigma^2 = n\,p\,(1–p)$ may then be *conjectured*. However, *DERIVE* cannot perform the summations when a limit for the summation is non-numeric, so that a general proof is not possible. This is a problem because omission of a formal proof gives the impression that a conjecture based on a few numerical results constitutes a proof in its own right. We suggest that if a general proof is not be given, there should be, as a minimum, a statement that the proof of the general case is missing.

This proof may be approached by either by attempting to calculate the moment generating function, $M(t) = \sum_{x \in s} e^{xt}\, P(X = x)$, (where the range of *t* is such that the series is convergent) or the probability generating function for a Binomial(*n,p*) random variable. The motivation for introducing the moment generating function is that it is required to prove the Central Limit Theorem, which we aim to discuss in a later paper. The same problem is encountered in computing a sum in which there is a symbolic limit. In the case of the moment generating function, the relevant sum is

$$M(p,n,t) = \sum_{x=0}^{n} e^{xt}\, p^x (1-p)^{n-x}\, C_x^n \quad .$$

Unfortunately, *DERIVE* cannot simplify it to obtain the result $M(p,n,t) = \left(p\,e^t + (1-p)\right)^n$.

This illustrates a general inability of *DERIVE* to handle such series, which have proved to be troublesome in other computer algebra packages as well. However, once it has been determined by hand, *DERIVE* can very easily perform the necessary calculus by calculating $\mu = M'(0)$ and $\sigma^2 = M''(0) - \mu^2$. A particularly compact way of doing this is illustrated in the *DERIVE* session below.

```
#1:  "PR_DIST1.MTH"                                          User
                           t          n
#2:  M(p, n, t) := (p·ê  + (1 - p))                         User
                      d
#3:  mean := lim     ── M(p, n, t)                           User
             t->0    dt

#4:  n·p                                                     Simp(#3)
                        ┌d┐2                    2
#5:  variance := [lim   │──│  M(p, n, t)] - mean             User
                  t->0  └dt┘

#6:  n·p·(1 - p)                                             Simp(#5)
```

In the case of a Geometric random variable the same problems are encountered. There is an additional problem in that, given a particular case, *DERIVE* is only able to obtain a result for the moment generating function, *M(t)*, of such a random variable, if the range of the parameter *t* is restricted so that the series is convergent. This problem is not unique to the Geometric distribution, but it is easy to see what the restriction should be in this case.

In the example below, in which the moment generating function for a Geometric(0.25) distribution is calculated, we must restrict the range of t so that $0 < 0.75\ e^t < 1$ (i.e. $0 < t < \ln(4/3)$). Unfortunately, if **D**eclare **V**ariable is used with the upper limit ln(4/3), *DERIVE* substitutes the rational form 1817/6316 for ln(4/3). This approximation is slightly larger than ln(4/3) and the series will not converge. A smaller approximation, such as $\ln(4/3) \approx 0.28768$, allows convergence to the correct result, $1/(4 - 3e^t)$. *DERIVE* changes this decimal to the rational form 899/3125. The payback for *DERIVE*´s inability to sum this series unaided is therefore some very positive work on the geometric series.

```
                    ∞                x   x·t
#7:  M(p, n, t) :=  Σ   0.25·0.75 ·ê                            User
                   x=0

#8:  ?                                                       Simp(#7)

#9:  t :ε Real [0, 899/3125]                                     User

            1
      ────────────
#10:           t                                             Simp(#7)
       4 - 3·ê
```

The general case of computing the moment generating function for a Geometric(p) random variable also cannot be done at all. The result

$$M(t) = \frac{p\,e^t}{1 - (1-p)\,e^t}$$

has to be obtained on paper, but, as with the Binomial(n,p) distribution, *DERIVE* can easily calculate and use $M'(0)$ and $M''(0)$ to find the mean, $1/p$, and the variance $(1 - p) / p^2$.

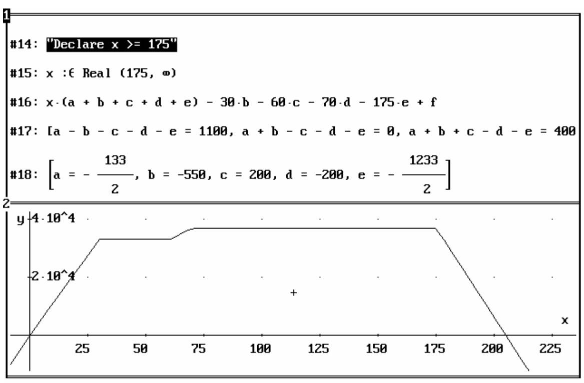
If, instead of using the moment generating function, we use the probability generating function of the distribution, $G(t) = \sum_{r \in s} t^r\, p_r$, the same general problems occur. In the binomial case, the series cannot be summed unless the limits for the sum are explicitly numeric. However, the general result $G(t) = (p\ t + 1 - p)^n$ is relatively easy to obtain by considering binomial expansions. In the geometric case, the series cannot be summed unless the probability that an event happens is explicitly numeric. Given that $G(t)$ has been determined, use of the formulae

$$\mu = G'(1) \quad\text{and}\quad \sigma^2 = G''(1) + G'(1) - [G'(1)]^2$$

for the mean and the variance of a random variable proceeds smoothly.

The figure shows the DERIVE realization of Ed Laughbaum's TI 92 paper (page 54), how to combine piecewise defined functions in a sum of absolute values function.

Until now I've always investigated these functions from the other direction: decompose such a function in its linear components. I find it very challenging to reverse that investigation. Josef



Comment of the Editor: I leave this contribution in its original form from 1996. Many functions which Johann developed have been implemented since then. You can find the DERIVE 6 – suitable file at the end of Titbits (9). titbits9_new.mth is the respective file.

Titbits from Algebra and Number Theory (9)

by Johann Wiesenbauer, Vienna

Time is flying, isn't it? It doesn't seem so long ago that I wrote the first column in this series using DERIVE 2.56 then. In the meantime numerous new versions of DERIVE have come out taking account of the various users' wishes. The present culmination of this development is DERIVE for Windows 4.0 (called DfW in the following for short) which has become available these days.

What I think about DfW? Well, despite a lot of apparent improvements such as the lightning-fast graphics, the new file management system or the enhanced scrolling capabilities, it isn't exactly true that it made me feel all warm inside from the very beginning - you can't teach an old dog new tricks, as the saying goes. What I found most irritating was the input window that pops up every few seconds - why on earth didn't they use the main window for this purpose like every other CAS I know of ? -, but after minimizing it, i.e. increasing the screen resolution and maximizing the main window at the same time, I finally got used to it. Currently I am discovering new nice features of DfW every day, and it is no longer inconceivable that I end up in saying one day: "I like it".

One of those nice features I was talking about is the option to include the main variable as a third parameter in the functions REMAINDER and QUOTIENT. (Up to now, the main variable could only be adjusted by using "Manage Order" from the main menu.) To the best of my knowledge, this was suggested by Eugenio Roanes-Lozano, who had to apply in [3] a lot of trickery to circumvent this restriction when defining his function PREM(p,q,v):

```
POLY_COEFF(u,x,n):=1/n!*LIM(DIF(u,x,n),x,0)

POLY_DEGREE_AUX(u,x,n):=IF(u=0,n,POLY_DEGREE_AUX(DIF(u,x),x,n+1),
    POLY_DEGREE_AUX(DIF(u,x),x,n+1))

POLY_DEGREE(u,x):=POLY_DEGREE_AUX(u,x,-1)

LCOEFF(p,v):=POLY_COEFF(p,v,POLY_DEGREE(p,v))

arb_var:=(RHS(SOLVE(0=0,x))) SUB 1

SUBST(f,x,a):=LIM(f,x,a)

SWAP_AUX(f,x,t,y):=SUBST(SUBST(SUBST(f,x,t),y,x),t,y)

SWAP(f,x,y):=SWAP_AUX(f,x,arb_var,y)

REMAINDER_V_AUX(p,q,v,v0):=SWAP(REMAINDER(SWAP(p,v,v0),
    SWAP(q,v,v0)),v,v0)

REMAINDER_V(p,q,v):=REMAINDER_V_AUX(p,q,v,(VARIABLES([p,q])) SUB 1)

MUL_FAC(p,q,v):=LCOEFF(q,v)^(1+POLY_DEGREE(p,v)-POLY_DEGREE(q,v))

PREM(p,q,v):=REMAINDER_V(MUL_FAC(p,q,v)*p,q,v)
```

In DfW the same function can be defined without any auxiliary functions (!) as follows:

```
PREM(p,q,v):=REMAINDER(LIM((TERMS(EXPAND(q,v))) SUB 1,v,1)^
    (1+LIM(v*DIF(f_,v)/f_,f_,(TERMS(EXPAND(p,v))) SUB 1)-
    LIM(v*DIF(f_,v)/f_,f_,(TERMS(EXPAND(q,v))) SUB 1))*p,q,v)
```

On close inspection of this code you will find out that I used the following more efficient implementations

```
POLY_DEGREE(u,x):=LIM(x*DIF(f_,x)/f_,f_,(TERMS(EXPAND(u,x))) SUB 1
LCOEFF(u,x):= LIM((TERMS(EXPAND(u,x))) SUB 1,x,1)
```

Though I didn't need POLY_COEFF(u,x,n) I would like to point out the following interesting alternative to the definition given above (cp. DNL #21, p37):

```
POLY_COEFF(u,x,n):=QUOTIENT(REMAINDER(u,x^(n+1),x),x^n,x)
```

Furthermore, for those with older versions of DERIVE the following implementation of PREM might still be of interest (if @ is not available in your DERIVE version then use any other unused variable instead of it!):

```
SWAP(f,x,y):=ITERATE(LIM(f_,@,y),f_,LIM(f,[x,y],[@,x]),1)
REM(p,q,v):=ITERATE(SWAP(REMAINDER(SWAP(p,v,v_),SWAP(q,v,v_)),v,v_),v_,
    (VARIABLES([p,q])) SUB 1,1)
PREM(p,q,v):=REM(LIM((TERMS(EXPAND(q,v))) SUB 1,v,1)^(1+LIM(v*DIF(f_,v)/f_,f_,
    (TERMS(EXPAND(p,v))) SUB 1)-LIM(v*DIF(f_,v)/f_,f_,
    (TERMS(EXPAND(q,v))) SUB 1))*p,q,v)
```

You might wonder by now what this mysterious PREM-routine is all about. Sorry, I should have told you that earlier! Above all it is a very useful tool to solve a system of polynomial equations by transforming it into a solvable system of equations without "losing" any solutions of the original system. To be more precise, if f and g are polynomials in $x_1,...,x_n$, where we may assume that $\deg(f,x_1) \geq \deg(g,x_1)$ w.l.o.g., then either $\deg(g,x_1)) = 0$ or we can shift to polynomials q, r in $x_1,...,x_n$, which are defined by the following equation (reflecting the polynomial division of $m \cdot f$ by g):

$$m \cdot f = q \cdot g + r \text{ with } \deg(r,x_1) < \deg(f,x_1).$$

Here m is a polynomial in $x_1,...,x_n$, that is chosen in such a way that fractions are avoided (see the formula MUL_FAC(p,q,v) above for its exact definition) and the pseudo-remainder r is the output of PREM(f,g,x1). Again either $\deg(r,x_1) = 0$ or we can repeat this procedure with g and r until we finally arrive at polynomials f', g', s.t. g' doesn't contain x_1 any longer, and it is clear from the construction that every solution of $f = g = 0$ is also a solution of $f' = g' = 0$.

Given a system of polynomial equations this algorithm can be used to achieve a "triangulated system" in a way very similar to Gauß' elimination. I will try to give you an idea how this works by means of a simple example, which was also used by A.Perotti in [2]. We consider the system of polynomial equations

$$x\cdot y=1-z \qquad (1)$$
$$x\cdot z=1-y$$
$$y\cdot z=1-x$$

By applying PREM three times

PREM(x*y-1+z,x*z-1+y,z)=x^2*y-x-y+1 (2a)

PREM(x*y-1+z,y*z-1+x,z)=x*(y^2-1)-y+1 (2b)

PREM(x*(y^2-1)-y+1,x^2*y-x-y+1,y)=-x^5+x^4+2*x^3-3*x^2+x (3)

we get the triangulated systems made up of (1),(2a),(3) or (1),(2b),(3), respectively. Solving (3) w.r.t. x yields

SOLVE(-x^5+x^4+2*x^3-3*x^2+x,x)=
[x=0,x=1,x=SQRT(5)/2-1/2,x=-SQRT(5)/2-1/2]

Using (2a) and (2b) we get the corresponding values of y:

SOLVE(LIM(x^2*y-x-y+1,x,0),y)=[y=1]

SOLVE(LIM(x^2·y - x - y + 1, x, 1), y) = [y = @2]

SOLVE(LIM(x*(y^2-1)-y+1,x,1),y)=[y=0,y=1]

SOLVE(LIM(x^2*y-x-y+1,x,SQRT(5)/2-1/2),y)=[y=SQRT(5)/2-1/2]

SOLVE(LIM(x^2*y-x-y+1,x,-SQRT(5)/2-1/2),y)=[y=-SQRT(5)/2-1/2]

Now for reasons of symmetry $y = z$ must hold except for $x = 1$, where we only know that $y,z\in\{0,1\}$. By using (1) to decide this ambiguous case we finally arrive at the five triples

$$(0,1,1),(1,0,1),(1,1,0),\left(\frac{\sqrt{5}-1}{2},\frac{\sqrt{5}-1}{2},\frac{\sqrt{5}-1}{2}\right),\left(\frac{-\sqrt{5}-1}{2},\frac{-\sqrt{5}-1}{2},\frac{-\sqrt{5}-1}{2}\right).$$

All these triples are really solutions of the original system. Note that we have actually used (1),(2a),(2b),(3) to compute them. It is easy to see that both triangulated systems (1),(2a),(3) and (1),(2b),(3) on their own have more solutions than the original system. (Check it!).

One could solve this system of polynomial equations also by means of my SOLVE2-function from my last 'Titbits' but it is very tricky and cannot be generalized. I hate to admit it, but Eugenio's PREM-function is streets ahead of my RED-function (the kernel of SOLVE2), when it comes to solving systems of polynomial equations. (It must be said though that it was invented for quite a different purpose, cf. DNL #20, p38.) If you are interested in learning more about applications of the PREM-function I can recommend to you reading Eugenio's wonderful paper [3] where he used it extensively to prove theorems of elementary geometry by the so-called Wu's method. At any rate I don't want to leave this topic without thanking him for his cooperation and many fruitful discussions via email.

Back to DfW. What else could be done to enhance it? Well, a lot of little things come to my mind, e.g. that there is no calculation time shown after the instant simplification of expressions by "=" or the fact that there is no Cartesian product for lists and sets, and what not. But from a number theoretic point of view there is one thing that is on the top of my wish list and once more it concerns the notorious powermod-function: It should also work for negative exponents!

At present if you ask e.g. for the inverse of 3 mod 5, i.e. mod(3^(-1),5), you will get the answer

mod(3^(-1),5)=1/3,

which is not exactly what you want. It could be programmed along the following lines (cp. DNL #17, p35):

```
POWERMOD(a,n,m):=IF(n>=0,MOD(a^n,m),IF(GCD(a,m)=1,MOD(MOD((ITERATE(
    IF(MOD(a_,b_)=0,[a_,b_,c_,d_],[b_,MOD(a_,b_),d_,c_-FLOOR(a_,b_)*d_]),
    [a_,b_,c_,d_],[a,m,1,0])) SUB 4,m)^(-n),m)))
```

If the exponent n is negative and gcd(a,m)>1, the output will be a question mark as, of course, powermod(a,n,m) is undefined in this case.

I will use this function to implement another very important formula which is based on the so-called Chinese Remainder Theorem (CRT):

If $m_1, m_2, ..., m_r$ is a sequence of pairwise coprime integers, then for arbitrary integers $a_1, a_2, ..., a_r$ the system of congruencies

$$x \equiv a_1 \bmod m_1,$$
$$x \equiv a_2 \bmod m_2,$$
$$...$$
$$x \equiv a_r \bmod m_r,$$

has a solution, which is unique mod $m_1.m_2...m_r$.

Fortunately, there exists also an explicit formula for the solution which can be written in DERIVE-notation as follows (cf. [1], p37)

```
CRT(a,m):=ITERATE(MOD(a*VECTOR(p_/m_*POWERMOD(p_/m_,-1,m_),m_,m),p_),
    p_,PRODUCT(m_,m_,m),1)
```

Here a denotes the vector $[a_1, a_2, ..., a_r]$ and m the vector $[m_1, m_2, ..., m_r]$. Again a question mark will be output if the numbers $m_1, m_2, ..., m_r$ are not pairwise coprime. Number theory abounds with applications of this important routine, but space is running out. Therefore, bye for now! (email: j.wiesenbauer@tuwien.ac.at)

References

[1] Wilfried Nöbauer and Johann Wiesenbauer, Zahlentheorie, Prugg-Verlag, Eisenstadt, 1981.

[2] Alessandro Perotti, Gröbner bases with DERIVE, The International DERIVE Journal, 1996, Vol.3,No.2, 83-98.

[3] Eugenio Roanes-Lozano and Eugenio Roanes-Macias, Automatic theorem proving in elementary geometry with DERIVE, The International DERIVE Journal, 1996, Vol.3,No.2, 67-82.

#1: $lcoeff(p, v) := POLY\_COEFF(p, v, POLY\_DEGREE(p, v))$

#2: $arb\_var := (SOLUTIONS(0 = 0, x))_1$

#3: $swap\_aux(f, x, t, y) := SUBST(SUBST(SUBST(f, x, t), y, x), t, y)$

#4: $swap(f, x, y) := swap\_aux(f, x, arb\_var, y)$

#5: $remainder\_v\_aux(p, q, v, v0) := swap(REMAINDER(swap(p, v, v0), swap(q, v, v0)), v, v0)$

#6: $remainder\_v(p, q, v) := remainder\_v\_aux(p, q, v, (VARIABLES([p, q]))_1)$

#7: $mul\_fac(p, q, v) := lcoeff(q, v)^{1 + POLY\_DEGREE(p, v) - POLY\_DEGREE(q, v)}$

#8: $prem(p, q, v) := remainder\_v(mul\_fac(p, q, v) \cdot p, q, v)$

#9: $prem(x \cdot y - 1 + z, x \cdot z - 1 + y, z) = x^2 \cdot y - x - y + 1$

#10: $prem(x \cdot y - 1 + z, y \cdot z - 1 + x, z) = x \cdot (y^2 - 1) - y + 1$

#11: $prem(x^2 \cdot y - x - y + 1, x \cdot (y^2 - 1) - y + 1, y) = x^2 \cdot y - x - y + 1$

#12: $prem(x \cdot (y^2 - 1) - y + 1, x^2 \cdot y - x - y + 1, y) = - x^5 + x^4 + 2 \cdot x^3 - 3 \cdot x^2 + x$

#13: $SOLUTIONS(- x^5 + x^4 + 2 \cdot x^3 - 3 \cdot x^2 + x = 0, x) = \left[0, 1, \frac{\sqrt{5}}{2} - \frac{1}{2}, - \frac{\sqrt{5}}{2} - \frac{1}{2}\right]$

#14: $SOLVE(SUBST(x^2 \cdot y - x - y + 1, x, 0), y) = (y = 1)$

#15: $SOLVE(SUBST(x^2 \cdot y - x - y + 1, x, 1), y) = true$

#16: $SOLVE(SUBST(x \cdot (y^2 - 1) - y + 1, x, 1), y) = (y = 1 \vee y = 0)$

#17: $SOLVE\left(SUBST\left(x^2 \cdot y - x - y + 1, x, \frac{\sqrt{5}}{2} - \frac{1}{2}\right), y\right) = \left(y = \frac{\sqrt{5}}{2} - \frac{1}{2}\right)$

#18: $SOLVE\left(SUBST\left(x^2 \cdot y - x - y + 1, x, - \frac{\sqrt{5}}{2} - \frac{1}{2}\right), y\right) = \left(y = - \frac{\sqrt{5}}{2} - \frac{1}{2}\right)$

#20: $POWERMOD(3, -1, 5) = 2$

#21: $POWER\_MOD(3, -1, 5) = 2$

#22: $CRT\_(a, m) := ITERATE\left(MOD\left(a \cdot VECTOR\left(\frac{p\_}{m\_} \cdot POWERMOD\left(\frac{p\_}{m\_}, -1, m\_\right), m\_, m\right), p\_\right), p\_, \Pi(m\_, m\_, m), 1\right)$

#23: $CRT\_([1, 2, 3], [4, 5, 11]) = 157$

#24: $CRT\_([MOD(83, 5), MOD(83, 7), MOD(83, 16)], [5, 7, 16]) = 83$

#25: $CRT([1, 2, 3], [4, 5, 11]) = 157$

#26: $CRT([MOD(83, 5), MOD(83, 7), MOD(83, 16)], [5, 7, 16]) = 83$

POWERMOD from page 51 vs implemented POWER_MOD and CRT_ from page 51 (now CRT_) vs the implemented CRT-function.

157 is the solution of the congruencies $x = 1 \bmod 4$, $x = 2 \bmod 5$, and $x = 3 \bmod 11$.

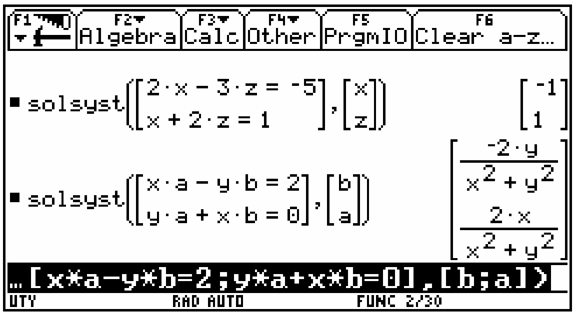
As I don't want to print all the TI-programs and -functions which I have received for the DNL´s TI - Corner - many thanks to you all -I include the files in a special TI-directory on the Diskette of the Year. I am sure that not all TI users have TI-Graph LINK at their disposal so they will not be able to use the programs. I tried to find another way: I include a document (in Word for WINDOWS 6 format) containing many programs and functions in readable form. (User Forum, Terence Etchells) You will find W. Pröpper's wonderful DIRA - for discussing curves - in a German and in an English version (DNL#23). I also have the pleasure to enclose D. Stoutemyer's Program Library together with a printable documentation. Many thanks David. DIRA and the library are not on a Word document because they are too big. At last I want to point out that there is lot of useful TI-stuff to be downloaded from the Internet. (Web pages of TI, SWH, ...)

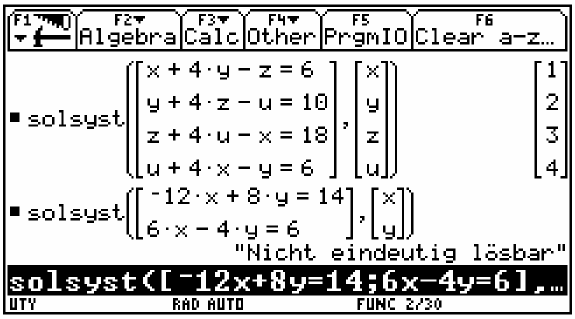
At last I´m glad to announce a lot of TI-contributions for the next year. Most of them will be interesting for DERIVE Users, too. (So I think that E.Laughbaum´s contribution for the TI on the next pages can easily be transferred on PC-DERIVE, see page 47.) There will be among others:
Dynamic System, Permutations, Numbers of Bernoulli, Continued Fractions, R.Schorn; Mortgage Tables, Tania Koller; The Simplex Method on the TI, Bruce Chaffee; Diophantine Equations & The Gauss-Seidel Method, L.Tortosa & J.Santacruz; Endpoint vs. Interior Extrema, White & Leinbach; Investigations on GCD and LCM with the TI, Griebel; First experiences with a TI in the class room, Böhm....

SOLSYST - A FUNCTION TO SOLVE SIMULTANEOUS EQUATIONS

BY W. PRÖPPER, NÜRNBERG, GERMANY

```
Solsyst(sys,var)
Func
Local i,j,n,mtx,dtm
rowDim(sys)→n
newMat(n,n+1)→mtx
For i,1,n
 For j,1,n
  d(left(sys[i,1]),var[j,1])→mtx[i,j]
  right(sys[i,1])→mtx[i,n+1]
 EndFor
EndFor
det(subMat(mtx,1,1,n,n))→dtm
If getType(dtm)="NUM" and dtm≠Ø or getType(dtm)="EXPR" Then
 rref(mtx)→mtx
 Return subMat(mtx,1,n+1,n,n+1)
Else
 Return "Nicht eindeutig lòsbar"
EndIf
EndFunc
```





With this nice function you can solve systems of linear equations in a *DERIVE* like way. Non German speaking users have to change one string in the third line from the bottom: "Nicht eindeutig lösbar" to the equivalent of "No unique solution".

Sums of Absolute Value Functions: An Application

Edward D. Laughbaum, Columbus, Ohio
elaughba@math.ohio-state.edu

Consider the following:

a) The level of the drug Imipramine in the blood of a patient rises at a constant rate (for example 60 nanograms per week) until the patient is at the prescribed level and then the rate of change remains at 0% until the patient is taken off the drug at a constant rate (for example 90 nanograms per week).

b) For users of electricity who also use a heat pump, many power companies charge, for example, \$0.08 per kWh used for the first 1000 kWh and \$0.05 for the next 1000 and finally, \$0.03 for any consumption over 2000 kWh.

c) Sliding scale commissions are business examples of multi-constant rates of change and are used to provide incentive for higher employee productivity. For example, a business may pay the sales staff 3% commission on sales from \$0 to \$10,000, 5% on sales from \$10,001 to \$15,000, and 8% commission on sales \$15,001 and over.

d) The pay scale for piece-work by a telemarketing company is \$0.35 for the first 300 calls in the week, \$0.42 for the next 200 calls, and \$0.65 for any call over 500 calls.

e) A new long-distance phone company has the following rate schedule:
12¢ per minute for the first 15 minutes
9¢ per minute for the next 10 minutes, and
6¢ per minute for any time over 25 minutes.

f) The 1994 (or any year) United States federal income tax form 1040 schedule (for single filers) had rates of taxation of 15% on the first \$22,750 of taxable income, 28% on the next \$32,350, 31% on the next \$59,900 of taxable income, etc. That is, the taxable income brackets are at \$22,750, \$55,100, and \$115,000. (There were two more brackets that will be ignored for the sake of brevity.)

g) A commercial airline flight from Reno, Nevada to St. Louis, Missouri ascends at a constant rate (after initial take-off) of 1100 feet per minute for 30 minutes (until it reaches a cruising altitude of 33,000 feet). It then levels off until it is 60 minutes into the flight. At 60 minutes into the flight, it has burned off enough fuel to ascend to 37,000 feet at a rate of 400 feet per minute. This takes 10 minutes. The plane remains at 37,000 feet until 175 minutes of flight time when it descends at a rate of 1233 feet per minute and then lands in St. Louis.

The Algorithm for Finding the Model.

The algorithm assumes a knowledge of behavior of sums of absolute value functions (Laughbaum, 1996).

Recognize that the structure of the model for all of the above situations is

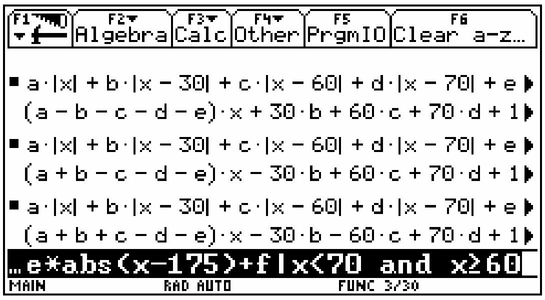
$M = a|x+e_1| + b|x+e_2| + \ldots d|x+e_n| + f$, where there are *n* corners.

1. Find the corners (e_1, e_2, …) [corners are normally known]
2. Simplify the model for each rate interval using the TI-92
3. Set the coefficients of *x* equal to each rate of change. [rates of change are normally known]
4. Solve the system.
5. Find *f* using a geometric transformation of a vertical shift.

An Example:

The on-board computer on a commercial airline flight from Reno, Nevada to St. Louis, Missouri commands the plane to ascend at a constant rate (after initial take-off) of 1100 feet per minute for 30 minutes (until it reaches a cruising altitude of 33,000 feet). It then levels off until it is 60 minutes into the flight. At 60 minutes into the flight, it has burned off enough fuel to ascend to 37,000 feet at a rate of 400 feet per minute. This takes 10 minutes. The plane remains at 37,000 feet until 175 minutes of flight time when it descends at a rate of 1233 feet per minute and then lands in St. Louis.

1. Recognize that the structure of the model is
$h(x) = a|x+e_1| + b|x+e_2| + c|x+e_3| + d|x+e_4| + e|x+e_5| + f$, where there are 5 corners.
2. Find the corner parameters (*e*1, *e*2, …). They are given as 0, -30, -60. -70, and -175. Thus, the initial model is $h(x) = a|x| + b|x-30| + c|x-60| + d|x-70| + e|x-175| + f$, where x is time in minutes.
3. Simplify the model for each rate interval using the TI-92

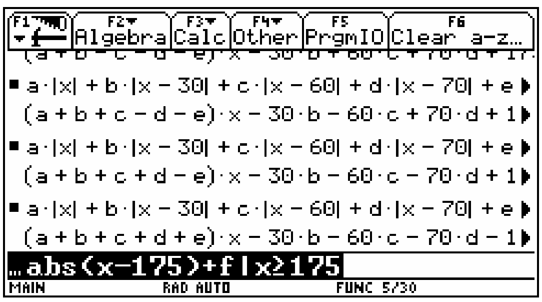


Conditions:

$x \geq 0$ and $x < 30$

$x \geq 30$ and $x < 60$

$x \geq 60$ and $x < 70$



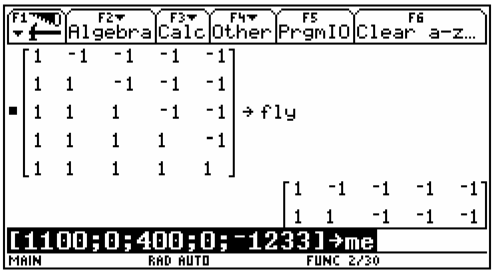
$x \geq 70$ and $x < 175$

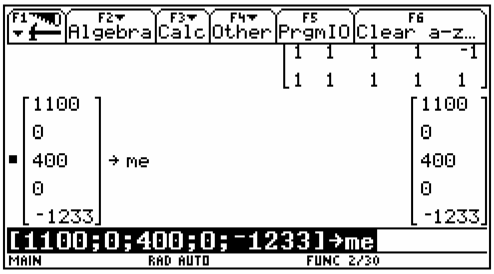
$x \geq 175$

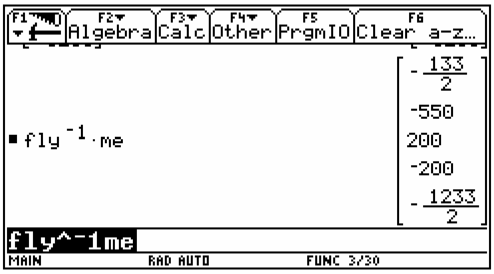
4. Set the coefficients of x equal to each rate of change; that is, each piece of the model simplifies to a linear function as shown on the TI-92 screens. Thus, the coefficient of x is the rate of change for each piece.

$$\begin{aligned} a-b-c-d-e &= 1100 \\ a+b-c-d-e &= 0 \\ a+b+c-d-e &= 400 \\ a+b+c+d-e &= 0 \\ a+b+c+d+e &= -1233 \end{aligned}$$

5. Solve the system.



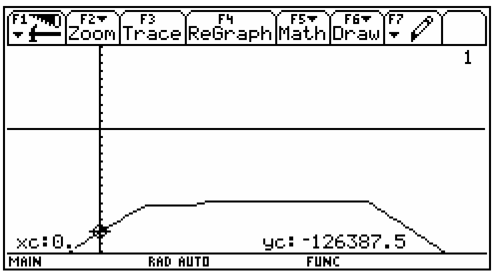


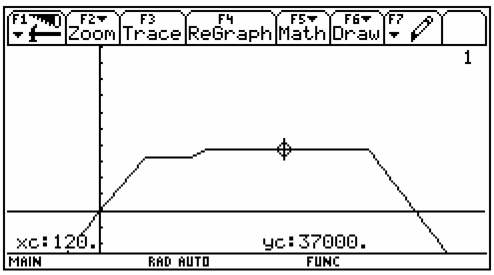


The model for the height control function is

$$h(x) = -\frac{133}{2}|x| - 550|x-30| + 200|x-60| - 200|x-70| - \frac{1233}{2}|x-175| + f\ .$$

6. Find f using a geometric transformation of a vertical shift.
When x is 0, for example, the function is -126387.5. Add 126387.5 as the parameter f.





The final model needed by the computer is

$$h(x) = -\frac{133}{2}|x| - 550|x-30| + 200|x-60| - 200|x-70| - \frac{1233}{2}|x-175| + 126387.5\ .$$

References

Laughbaum, E. D. (1996). Modeling data exhibiting multi-constant rates of change.
The AMATYC Review, 17(2), 27-34.

(*You can find the equivalent DERIVE application as SUM_ABS.MTH. Josef*)