# THE DERIVE - NEWSLETTER #7

## THE BULLETIN OF THE DERIVE USER GROUP

**Contents:**




revised reprint 2005 **September 1992**

**Proceedings of the International Spring School on the Didactics of Computer Algebra, Krems 1992**

The proceedings will be published from Chartwell-Bratt, UK and will be available in December 1992.

**The University of Birmingham**

Technology in Mathematics Teaching
A Bridge between Teaching and Learning

17 - 20 September 1993

The structure of the programme provides for those involved in the teaching of mathematics at every level. There will be a diversity of themes, both educational and technological, and opportunities for talks, workshops, research reports, symposia and discussion groups.

The British DUG-Meeting 1993 will be held in the frame of this conference (K. and D. Stoutemyer will be present.)

Further informations (Call for Papers and Presentations):

Pam Bishop (Vicepresident of the DUG)
CTICMS
Faculty of Education
The University of Birmingham
Edgbaston
Birmingham B15 2TT
UK

**International Conference on the Didactics of Computer Algebra in Krems, Austria, 21 - 25 September 1993**

You will find a Call for Papers in the next DNL.

Karen and David Stoutemyer (Soft Warehouse Hawaii) will honor this conference with their presence, too.

**The Leeds University**

British Congress of Mathematics Education
15 - 19 July 1993

The programme committee is looking for contributions, concerning working with DERIVE. (15 October 1992)

Informations: Rita Nolder
Department of Education
Loughborough University
Loughborough LE11 3TU
UK

Lieber Derive Anwender!

Ich hoffe, dass Sie den Sommer gut verbracht haben und dass Sie gut erholt in einen hoffentlich sehr schönen Herbst gehen. Ich hatte heuer endlich die Gelegenheit die USA kennen zu lernen und konnte mich davon überzeugen, dass dieses wunderbare Land neben dem Silicon Valley noch einiges zu bieten hat.

Die DUG besteht zur Zeit aus 450 Mitgliedern und ist mittlerweile zu einer weltumspannenden Gruppe geworden. Während der Sommerferien sind besonders viele neue Mitglieder aus den USA, Kanada, Australien und sogar Hongkong zu uns gestoßen. Wir heißen Sie alle recht herzlich willkommen und hoffen auf Ihre Mitarbeit. Ich freue mich, Ihnen für den nächsten DNL den ersten Beitrag aus Südamerika ankündigen zu können: Extended Fourier Series aus Petropolis, BRA.

Sie werden in diesem DNL vielleicht Mr Setif's Treasure Box vermissen. Keine Angst, die Schatzkiste ist noch lange nicht leer und schon in unserer nächsten Ausgabe werden wir wieder tief hineinlangen können. Dafür gibt es aber ab dieser Ausgabe eine Neuerung. Wie schon angekündigt, stellt Soft Warehouse Hawaii die Anfragen und Antworten aus dem Derive Bulletin Board Service (BBS) der DUG zur Verfügung. Ich habe die, mir als interessantest erscheinenden Teile aus dem Zeitraum Sept.91 bis Mai 92 ausgewählt und ab Seite 5 können Sie die Auszüge finden. Ab der nächsten Nummer kann ich Ihnen aktuellere Meldungen wiedergeben.

Abschließend möchte ich auf die allgemeinen Informationen auf der Innenseite des Umschlags hinweisen.

Mit den besten Grüßen

Dear Derive User,

I hope that you have had a fine summer so that you are now going into a beautiful autumn in a relaxed state of mind. Oh, sorry our friends from Australia and Brasil have just passed wintertime!. Fortunately I had the opportunity to visit the US for the first time and I could make sure myself that this wonderful country offers some more things beside Silicon Valley.

The DUG now has 450 members and has developed to a world wide group. During my holidays many new members have joined the DUG, especially from USA, Canada, Australia and Hongkong. We warmly welcome you all and are looking forward to your cooperation. So I'm glad to announce the first contribution from South America for the next DNL: Extended Fourier Series from Petropolis, BRA.

Maybe you will miss Mr. Setif's Treasure Box in this issue. Don't worry, the treasure box isn't empty and next time we will reach deep into it again. But there is a novelty in DNL#7, too. In DNL#6 I announced that Soft Warehouse Hawaii will provide the questions and answers from the DERIVE Bulletin Board Service (BBS) for the DUG. I have selected those parts of the 9/91 - 5/92 period which seemed to me to be most interesting for you all. You will find them on page 5.

Finally I want to call your attention to the informations on the inside cover.

With my best regards

Download all *DNL-DERIVE*- and TI-files from

http://www.austromath.ac.at/dug/

http://www.bk-teachware.com/main.asp?session=375059

The *Derive-News-Letter* is the Bulletin of the Derive-User Group. It is published at least three times a year with a contents of 30 pages minimum. The goals of the *D-N-L* are to enable the exchange of experiences made with Derive as well as to create a group to discuss the possibilities of new methodical and didactic manners in teaching Mathematics.

Editor: Mag. Josef Böhm
A-3042 Würmla
D´Lust 1
Austria
Phone/FAX:43-(0)2275/8207
e-mail: nojo.boehm@pgv.at

**Contributions:**

Please send all contributions to the above address. Non-English speakers are encouraged to write their contributions in English to reinforce the international touch of *D-N-L*. It must be said, though, that non-English articles will be warmly welcomed nonethe-less. 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 to the Editor, the more lively and richer in contents the Derive-News-Letter will be.

**Preview: Contributions waiting to be published**

Mrs Castelletti's worksheets;

Extended Fourier Series;

The Mechanics of Erkki Ahonen;

Rational Collocation;

(will be published December 92)

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

**Siemens Linz, Austria**

Here are two sample calculations under v. 2.10 when Manage Trigonometry reads

Direction (Auto)      Towards (Auto)
(which is now Options > Mode Settings > Trigonometry and Options > Mode Settings > TrigPowers)

The first is

2 COS(X) SIN(4X) – SIN(3X) – SIN(5X)

which (not surprisingly) does not Simplify to zero. The second is

TAYLOR ( TAN(X), X, 0, 5 )

which comes up with the quintic expansion on my 25 MHz 386/7 combination after only 0.2 seconds.

To get the trigonometric simplification working as I want it to, I called up Manage Trigonometry and changed settings to

Direction (Collect)   Towards (Auto)

Then

2 COS(X) SIN(4X) – SIN(3X) – SIN(5X)

simplifies to 0 as required; but

TAYLOR ( TAN(X), X, 0, 5)

now takes 56 seconds!! And TAYLOR (TAN(X), X, 0, 7) does not come up with anything at all in reasonable time. (On my 20 MHz 386/7SX combination at home, this command generates a Memory Full error message.)

(The 5th order expansion shows no problems with DERIVE6, the 7th order does!)

***D-N-L:*** The dramatic increase in time required to compute TAYLOR(TAN(x),x,0,5) when collecting trig expressions is due to the exponential growth of the derivatives of TAN(X) when in this mode. For example, compare the size of the 4th derivative of TAN(X) computed in "Auto" mode versus its size computed in "Collect" mode.

I recommend you compute this Taylor series in "Auto" mode and the resimplify the result in "Collect" mode to get it in the form you desire.

Note that we cannot always do Taylor series simplification in "Auto" mode because there may be other examples that work well only in "Collect" or in "Expand" mode. Trig simplification is very difficult! Until we know how to fully automate the process, users will have to learn to experiment.

Sincerely,
Albert D. Rich
Applied Logician

Trigonometry := Auto

$$\left(\frac{d}{dx}\right)^4 \text{TAN}(x) = \frac{24\cdot\text{SIN}(x)}{\text{COS}(x)^5} - \frac{8\cdot\text{SIN}(x)}{\text{COS}(x)^3}$$

Trigonometry := Collect

$$\left(\frac{d}{dx}\right)^4 \text{TAN}(x) = -\frac{\text{SIN}(14\cdot x) - 55\cdot\text{SIN}(10\cdot x) - 320\cdot\text{SIN}(8\cdot x) - 11\cdot(81\cdot\text{SIN}(6\cdot x) + 128\cdot\text{SIN}(4\cdot x) + 105\cdot\text{SIN}(2\cdot x))}{4\cdot(\text{COS}(2\cdot x) + 1)^8}$$

**Volker Neurath, Velbert**

To Mr. Appel's wishlist

I cannot follow Mr. Appel's argumentation when he says that pupils often want to test their teacher's competence. It's up to him to explain what he wants to say with that sentence. Does he mean, that pupils often try other ways of solution? Now, that is, in my opinion, an effect which is to be welcomed. Why? Now, because they surely would not have tried, if they have not had the possibility to use DERIVE. And using DERIVE they can try and immediately see, whether their way of thinking was correct or not. I'm not a teacher but I was a pupil for a very long time, and at the moment, I'm again. From my own experience I can say that since I am using DERIVE, I often try ways of solving a problem I surely never would if I had to do by hands. So I would not agree to Mr. Appel's wish for the ability to switch off menu-points « not needed » because with this ability he would hinder the pupils' creativity in regard to problems solution.
Regarding to the problem of steering DERIVE I agree: A mouse would be very useful, also the possibility of editing an expression after having pressed ENTER. In addition to this I think DERIVE should get own printer-drivers to get better printing results especially when printing plots from DERIVE. Also, DERIVE should get some drivers for (HP)-plotters to get first-class plots.
Question: Is there any DERIVE User who has programmed such a function by himself - eventually for the SHARP CE-516P Plotter?

**Adam Marlewski, Poznan, Poland**

Thank you very much for sending me all 4 issues of your DERIVE-Newsletter from 1991 and for the numbers 5 and 6 from this year. I'm finding your paper very interesting and let me wish you the progress for the future.

It is very nice to hear that you are interested to publish some of my works concerning to the applications of DERIVE. Today I'm mailing a 4-paged paper "Rational collocation" devoted to the problem which was first investigated almost 2 centuries ago, but the final answer was found just 100 years ago (so it is the "round" anniversary!)

I hope it will be good enough to be included in the DERIVE-Newsletter. Let me wish you all the best, sincerely yours

Adam Marlewski

***D-N-L:*** Many thanks for the good wishes, Mr. Marlewski. Your contribution will be published in our next issue. We are looking forward for your next papers.

**Erkki Ahonen, Salo, Finland**

.... What do you like my files? The research in mathematics and physics (mechanics) has been long time my work, although I am not a professional physicist. I am a writer (middle level studies in physics, too), and I have new ideas about mechanics. Are the files suitable for circulation? The mathematics in these files is very simple.

***D-N-L:*** Dear Mr. Ahonen, thanks for your letter and the files. We will publish the first part of your work in our next issue.

Message 1307 was entered on 9/28/91 at 9:14 PM.
**From DAN to PUBLIC about INTEGRAL**

I am having trouble computing the following integral using derive: int(1/sin(x)/sqrt(sin(x)^2 - a^2)) with respect to x. This appeared on another group, but going a little crazy. I did set for the domain, a>0 and a<1. Any suggestions?

Message 1322 was entered on 10/3/91 at 9:33 AM. (Read 5 times)
**From DON HARRIS to PUBLIC about ODE1.MTH**

I am trying to learn something about differential equations (at a very elementary level) using DERIVE. For the logistic equation: y' = ky - cy^2, y(0)=A what is the simplest method of solving the equation using ODE1.MTH?
Would appreciate any help. Thanks.

**From JERRY GLYNN to PUBLIC about BIG STORM**

We've been off the air for about 8 hours because of big storms in our area. It seems clear now. Jerry

Message 1348 was entered on 10/10/91 at 4:28 PM. (Read 3 times)
**From FLAMM to PUBLIC about GATHERING TERMS**

I would like to know what the easiest way (if there is one) to do the following in DERIVE is:
for an expression like x^a * (1/x)^b * 1/x^c * x^-d simplify it to x^(a-b-c-d)....
how do you do this when x is a variable?
how do you do this when x is an expression?
any hints greatly appreciated...

Message 1448 was entered on 12/8/91 at 10:34 AM. (Read 88 times)
**From MIKE TREUDEN to PUBLIC about 132 COLUMN VIDEO MODES**

To use DERIVE in a video mode not directly supported via the program menus, modify the DERIVE.INI file. By using a text video mode with more than 80 columns you can view long expressions more easily. (Many VGA and super-VGA cards have such modes.) With my ATI VGA Wonder+ card, the 132x44x16 (132 columns, 44 lines, 16 colors) text mode is mode 51 (decimal, not hex). To use this mode, I used a text editor to change a line in the DERIVE.INI file to

```
*PREVIOUS-MODE*  51
```

Pressing the F5 key from within DERIVE enables the new mode.
(F5 will then toggle between the modes given by *VIDEO-MODE* and *PREVIOUS-MODE* in the DERIVE.INI file.)

Notes:
1. Video cards made by different companies won't use 51 in general. Consult your video card manual for the proper mode number.

2. If you happen to change the video mode using the menus, DERIVE will "forget" the change made to the DERIVE.INI file. To revive your special video mode, re-load the DERIVE.INI file from within DERIVE (using TRANSFER/LOAD/STATE). Then F5 will get you back to the special mode.

3. I have not had much luck with using graphics-only video modes. You may want to give them a try on your video card.

Message 1505 was entered on 1/7/92 at 10:53 AM. (Read 77 times)
**From JEFFCOLE to PUBLIC about INTERPRET SIGN(0)**

I would like to know if anyone has figured out how Derive interprets SIGN(0)-it doesn't seem to be consistent. For example, to find the volume of the sphere of radius 5 centered at the origin (first octant only), I set up the following double integral:
Integrand=(25-x^2-y^2)^(1/2)
Y limits=0 to (25-x^2)^(1/2)
X limits=0 to 5
Upon simplification, Derive gives the answer 125pi SIGN(0)/6, which is correct if SIGN(0) is given the value of 1. I am using version 2.05-any help?

Message 1532 was entered on 1/25/92 at 7:41 AM. (Read 82 times) (Received)
**From JERRY GLYNN to GADIEL about #1530 / HOW DO I DO X <- F(X) ?**

ITERATES may be the command you need. ITERATES(2x^3,x,2,3) simplified will produce a list of 4 terms [2,16,8192,1099511627776] which is the result of starting at 2 and applying the 2x^3 three times. If you define the function first f(x):=2x^3 then ITERATES(f(x),x,2,3) will produce the same results. If you define f(x):= and then do ITERATES(f(x),x,2,3) you'll get [2,f(2),f(f(2)),f(f(f(2)))] and if you do ITERATES (f(x),x,x,3) you'll get [x,f(x),f(f(x)),f(f(f(x)))]. ITERATE (with no s at the end) will produce just the last term. Add terms slowly since the size of the results can blow-up quickly and try approX if Simplify produces results that are too big. Let me know whether or not this helps.
Good luck

Message 1586 was entered on 2/16/92 at 3:51 AM. (Read 61 times)
**From ANDROMEDA to PUBLIC about MATH**

I am faced with the problem of solving the integral of the standard normal density function f(z)=1/sqrt(2*pi)*e^(-z^2/2) with respect to z from 0 to c without using numerical approximation techniques on DERIVE. My calculus text tells me that the integral of the above function does not exist. My main interest is in finding a function or an equation which I can put in my spreadsheet program to calculate the z (c) values if the area under the curve is known without resorting to the table of z values. DERIVE returns some ERF function which is beyond my level at this point, and none of the books or the user unfriendly DERIVE manual is of any help. Is it possible at all to solve the above equation for c (even if it is of advanced level)? Can any one please help me?

Thanks

Message 1589 was entered on 2/16/92 at 7:12 PM. (Read 58 times) (Received)
**From BOOM-BOOM to ANDROMEDA about #1586 / MATH - NORMAL INFO**

Andromeda:
I have used the following in a spreadsheet before with pretty good results:

Let Q(x) be the area under the normal curve from x to infinity then Q(x) can be apprxoximated by the following:
Z(x)*(at+bt^2+ct^3+dt^4+et^5) where Z(x) is the standard normal curve (1/sqr(2pi))exp(-x^2/2)) and t=1/(1+p*abs(x))

p=.2316419
a=.319381530
b=-.356563782
c=1.781477937
d=-1.821244978
e=1.330274429

The reference for this is the Handbook of Mathematical Functions, Abramowitz and Stegun, National Bureau of Standards. The edition I have is 1964 which, most PROBABLY (sic) is older than you!
Good luck, Larry Gilligan

Message 1742 was entered on 4/2/92 at 8:07 PM. (Read 46 times)
**From KYLE to PUBLIC about PIECEWISE FUNCTIONS ON DERIVE**

TO THE PUBLIC:

I AM A NEW USER OF DERIVE AND AM HAVING TROUBLE WITH THE PIECEWISE FUNCTIONS CAPABILITIES OF THIS PROGRAM. I WOULD LIKE TO KNOW IF ANYONE ELSE HAS HAD THIS TROUBLE AND WOULD MIND HELPING ME OR AT LEAST POINT ME IN THE RIGHT DIRECTION. I AM USING THIS ON A LAPTOP IN MY CALCULUS CLASS AND HAVE NEEDED THIS QUITE OFTEN IN THE PAST FEW DAYS. IF THERE ARE ANY SUGGESTIONS PLEASE POST THEM TO KYLE. I WOULD APPRECIATE IT VERY MUCH.

KYLE BROWN

Message 1745 was entered on 4/4/92 at 3:38 PM. (Read 42 times)
**From DELAWARE to KYLE about #1742 / PIECEWISE FUNCTIONS ON DERIVE**

Kyle: I too use the piecewise function capability in calculus classes but you'll have to be more specific about your problems. One problem I remember encountering as a beginner was how to enter the 'test clause' in the IF statement; using logical connectors turned out to be the key, i.e., OR and AND. For instance, to enter the interval [1,2) I enter:
x>=1 and x<2.

A second example in full might be entering the function f(x) =x^2 on the interval [-1,1], yielding: IF(x>=-1 and x<=1, x^2, ?). The final "?" tells Derive not to graph the function anywhere but on the specified interval.

A final example, as I shoot here into the dark, is graphing f(x)=x^2 once again on (-inf,0) and f(x)=sin x on [0,inf), a function in two pieces: IF(x<0,x^2,sin x). Functions in three or more pieces just require nested-IF statements. I have no idea whether any of these reflect your difficulties, but I hope they help. My user name here is DELAWARE. Good luck.

<<< Reply - see message #1747 >>>

Message 1747 was entered on 4/5/92 at 8:25 PM. (Read 42 times)
**From BOOM-BOOM to KYLE about #1742 / PIECEWISE FUNCTIONS REPLY**

Try this:
IF(x<=5,x^2-1,4-7x)
Plot this expression. It is the piecewise function defined to be y=x^2-1 if x<=5 and the function y=4-7x if x>5.
If you need more "branches" of the piecewise function, you can imbed if commands. Good luck, Larry Gilligan

<<< Last reply to message #1742 >>>

Message 1780 was entered on 4/14/92 at 1:30 PM. (Read 37 times)
**From DAVID to PUBLIC about FOURIER TRANSFORM**

I NEED SOME HELP IN ENTERING PROBLEMS TO DO FOURIER TRANSFORMS. I CAN SEE HOW TO ENTER A SINGLE FUNCTION BETWEEN TWO LIMITS BUT HOW DO I ENTER A PROBLEM THAT HAS TWO SEPERATE LIMITS?

AS AN EXAMPLE: I(H) = 4COSh FROM H=0 TO H=2PI AND I(H) INCREASES AT A CONSTANT RATE FROM 0 TO 4 BETWEEN H=PI/2 AND H=2PI. A PROBLEM LIKE V(H)=SIN(H/2) FROM H=0 TO H=2PI IS EASY ANS WORKS GREAT.

ANOTHER EXAMPLE I NEED HELP ENTERING:
V(H)= 2H^2 FROM H=0 TO H=PI AND V(H) = ... FROM PI TO H=2PI.

I AM GETTING SO MANY ERRORS AT THIS END I HOPE YOU CAN READ ANY OF THIS ..
THANK YOU... DAVID BEARD

Message 1826 was entered on 4/27/92 at 2:41 PM. (Read 22 times)
**From BMI to PUBLIC about POSSIBLE BUG IN FRESNEL INTEGR**

Gerry, I was using the expression VECTOR(FRESNEL_SIN(V),V,5,10,0.01) to generate data for graphing Cornu's spiral. DERIVE generated correct va-lues for all V except at V=8.65 and V=8.83. The values DERIVE generated were 1.00942 and 1.34605, respectively (clearly impossible) whereas the correct values should be (approximately) V=0.51 and V=0.53, respectively. WHAT HAPPENED! The similar FRESNEL_COS(V) expression did not have this pro-blem. (I am using DERIVE 2). Thanks for any help. Barry

(Problem is solved, Josef)

Message 1851 was entered on 5/9/92 at 6:45 PM. (Read 8 times)
**From HEINZ to PUBLIC about PIECEWISE FUNCTIONS**

Jerry Glynn:
Thank you for solving the piecewise function problem and for disecting or rather trisecting it so that I was able to understand the solution. But now I discovered to mt horror that I did not copy the problem correctly from Leinbach's book. The correct problem is:

f(x) = -1 if x <= 0
f(x) = x + 2 if 0<x<1
f(x) = 1 if x>=1

I can, of course author and plot each part separately but I can't get it all into one function. I am afraid this problem is an order of magnitude more difficult than the one I asked you before.

**Answer: IF(x ≤ 0, −1, IF(x < 1, x + 2, 1))**

# A Computer-Aided Approach to Differential Equations

Diana Mackie, Napier University, Edinburgh

For the past three years the author has been using *DERIVE* with second year undergraduates in a Science with Management Studies course at Napier University in Edinburgh. The calculus studied during this course includes methods of integration, solution of first-order differential equations with applications, approximation of functions by power series and partial differentiation. *DERIVE* was selected as the most suitable computer algebra package as it is very easy to use, due to its menu-driven format, and it offers flexible function plotting.

Computer laboratory sessions are used to reinforce, illustrate and extend topics covered in lectures, to motivate students and to challenge the more able ones. One of the topics for which *DERIVE* has proved useful is the solution of first-order differential equations. The first technique studied is the method of separation of variables. Students learn to recognize equations which can be solved by this method, to separate the variables and then use *DERIVE* to assist in the subsequent integration and algebraic manipulation.

In a similar way, useful practice is gained by following through the steps required to solve a linear first order equation using an integrating factor. Again the computer is used to assist with the integration and manipulation. Once these basic techniques have been understood the students use the supplied functions, `SEPARABLE` and `LINEAR1`, contained in the utility file `ODE1.MTH`, to solve further problems and investigate the behaviour of models. An example of a model which is suitable for investigation in this way is given in Appendix 1.

An important advantage of solving differential equations analytically is that general results can be obtained for a class of problems before particular models are studied. One such example is the logistic growth equation

$$y' = k\,y\,(r - y)\ ;\ k, r > 0.$$

As a result the students begin to recognize certain types of problems and to predict the behaviour of the solution. When numerical values are given for a particular model, the solution can be plotted as a graph. With the aid of DERIVE, he can readily investigate the effect on the solution of changing various parameters of the problem. In this way the student develops an understanding of the behaviour and properties of a model.

At a later stage, computer packages which offer a variety of numerical methods for the solution of differential equations can be used to study problems which require more advanced techniques or for which there is no analytical solution.

*DERIVE* allows students to explore concepts algebraically, graphically and numerically, free from the hindrance of difficult algebra or tedious arithmetic. It can prove to be a powerful motivating factor for students from science and engineering disciplines.

**Appendix 1**

A, B and C are chemical compounds. In a certain reaction, chemical A is converted to B and then to C. The rates of conversion from one compound to the next are as follows:

$$A \xrightarrow[p]{} B \xrightarrow[q]{} C$$

Let α = initial concentration of A, and *a*, *b*, *c* be the concentration of A, B and C, respectively, at time *t*. The rate of change of concentration of the three compounds can be modelled by the equations:

$$\frac{da}{dt} = -p \cdot a \qquad (1)$$

$$\frac{db}{dt} = p \cdot a - q \cdot b \qquad (2)$$

$$\frac{dc}{dt} = q \cdot b \qquad (3)$$

with $a(0) = \alpha, b(0) = c(0) = 0.$

Using the *DERIVE* function `SEPARABLE`, the solution to equation (1) can be found by the sequence:

#1: SEPARABLE(−p, a, t, a, 0, α)

#2: $\mathrm{LN}(a) - \mathrm{LN}(\alpha) = -p \cdot t$

#3: $a = \alpha \cdot e^{-p \cdot t}$

The solution to the equation (2), $b' + q \cdot b = p \cdot a$, is obtained by the sequence:

#4: LINEAR1(q, p·a, t, b, 0, 0)

Substitute for a the expression above

#5: $\mathrm{LINEAR1}(q, p \cdot (\alpha \cdot e^{-p \cdot t}), t, b, 0, 0)$

#6: $b = \frac{\alpha \cdot p \cdot e^{-p \cdot t - q \cdot t} \cdot (e^{p \cdot t} - e^{q \cdot t})}{p - q}$

Simplify > Expand

#7: $b = \frac{\alpha \cdot p \cdot e^{-q \cdot t} - \alpha \cdot p \cdot e^{-p \cdot t}}{p - q}$

Using this result, equation (3) can be solved as follows:

#8: SEPARABLE(q·b, 1, t, c, 0, 0)

Substitute for b the result from above

#9: $\text{SEPARABLE}\left(q\cdot\frac{\alpha\cdot p\cdot e^{-q\cdot t}-\alpha\cdot p\cdot e^{-p\cdot t}}{p-q}, 1, t, c, 0, 0\right)$

#10: $c=\frac{\alpha\cdot q\cdot e^{-p\cdot t}}{p-q}+\frac{\alpha\cdot p\cdot e^{-q\cdot t}}{q-p}+\alpha$

Values can now be substituted for α, *p* and *q* and the resulting solutions can be investigated graphically. Figure 1 shows the solution for the case $\alpha = 5$, $p = 0.2$ and $q = 0.3$. The answer to questions such as:

**"At what time is *a* = *c*?"**

**"When is the concentration of *a* = 1?"**

**"What is the maximum concentration of *b*?"**

can all be obtained at this stage using *DERIVE*. For example, to find the maximum concentration of *b*, differentiate the expression for *b* and solve the equation $b' = 0$. For the example shown, the result obtained is $t = 4.05465$ (using approx mode). Substitute this value into the equation for *b* and then Simplify to show that the maximum concentration of $b = 1.48$.

It is useful, also, to explore the cases where $p < q$, $p > q$ and $p = q$. Note that the solutions obtained for *b* and *c* break down when $p = q$. In this case, substituting $q = p$ in the LINEAR1-function produces the amended solution

Special case p = q

#16: $\text{LINEAR1}(p, p\cdot(\alpha\cdot e^{-p\cdot t}), t, b, 0, 0) = \left(b = p\cdot\alpha\cdot e^{-p\cdot t}\cdot\int_0^t 1\, d@\right)$

#17: $\text{LINEAR1}(p, p\cdot(\alpha\cdot e^{-p\cdot t}), t, b, 0, 0) = (b = p\cdot t\cdot\alpha\cdot e^{-p\cdot t})$

#18: $\text{SEPARABLE}(p\cdot p\cdot t\cdot\alpha\cdot e^{-p\cdot t}, 1, t, c, 0, 0) = (c = \alpha - \alpha\cdot e^{-p\cdot t}\cdot(p\cdot t + 1))$

or calculating the respective limits for b and c from above:

#19: $\lim_{q\to p}\left(b=\frac{\alpha\cdot p\cdot e^{-p\cdot t-q\cdot t}\cdot(e^{p\cdot t}-e^{q\cdot t})}{p-q}\right) = (b = p\cdot t\cdot\alpha\cdot e^{-p\cdot t})$

#20: $\lim_{q\to p}\left(c = c = \frac{\alpha\cdot q\cdot e^{-p\cdot t}}{p-q}+\frac{\alpha\cdot p\cdot e^{-q\cdot t}}{q-p}+\alpha\right) = (c = \alpha - \alpha\cdot e^{-p\cdot t}\cdot(p\cdot t + 1))$

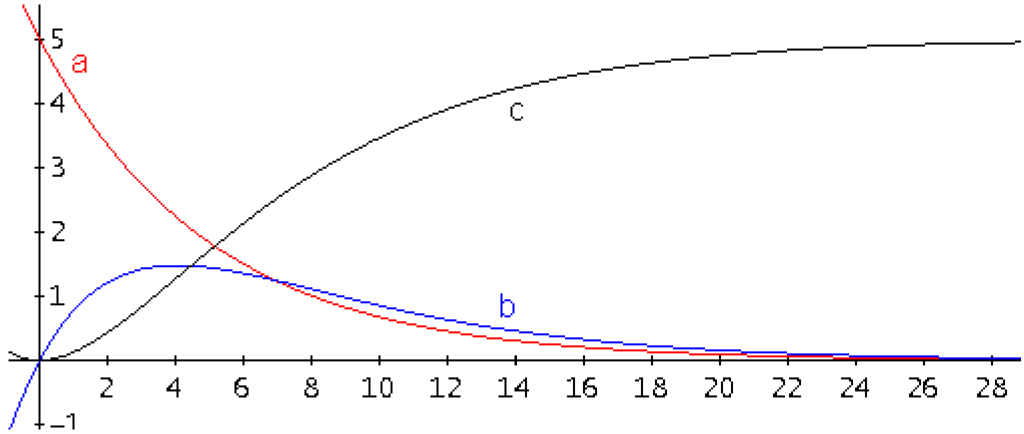
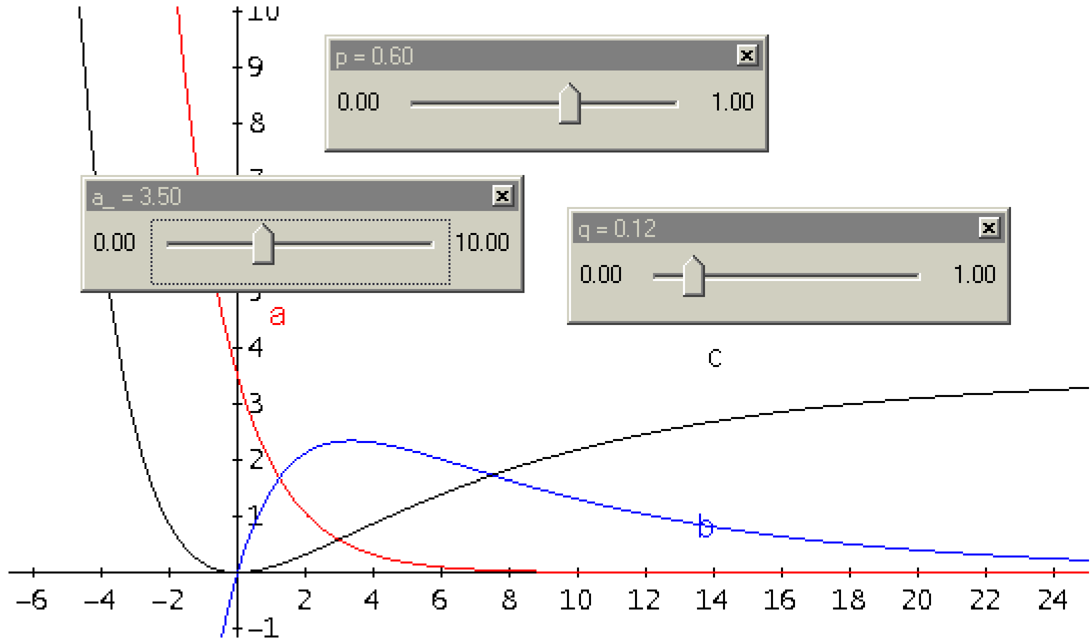


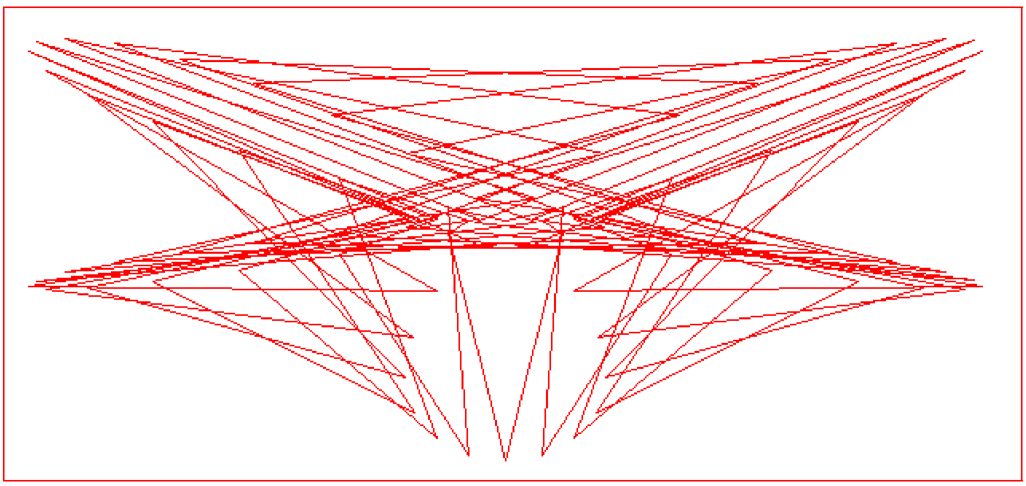

Figure 1

Concentration of chemicals A, B and C

DERIVE 6 allows introduction of slider bars of study the influence of p and q on the behaviour of the solution curves:

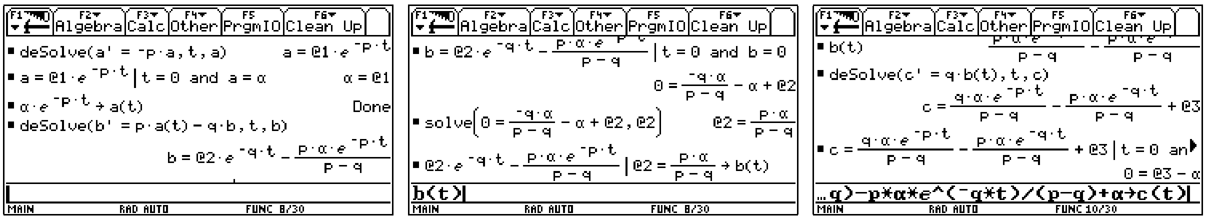



In 1992 there was no idea of having a CAS on handheld calculators. The CAS-TIs starting with the TI92 PLUS are powerful tools to treat DEs and systems of DEs as you can see on the following page.

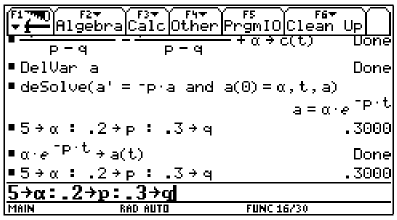


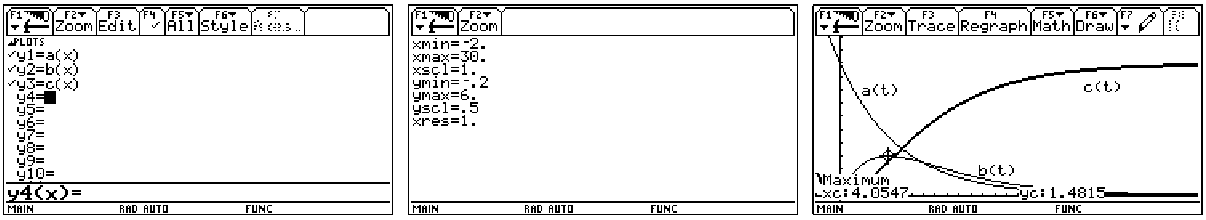
Maurer 9

The first four screen shots show the general solution of the system.

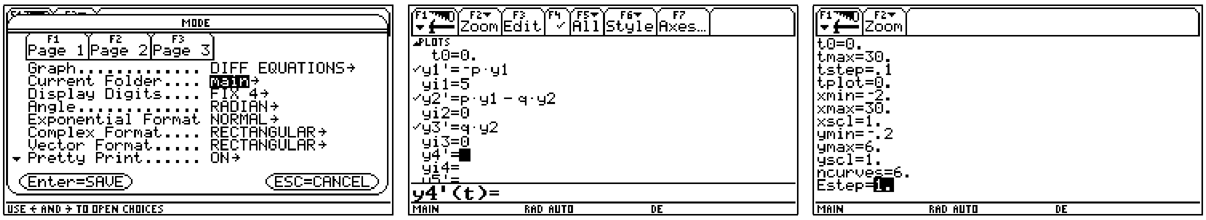


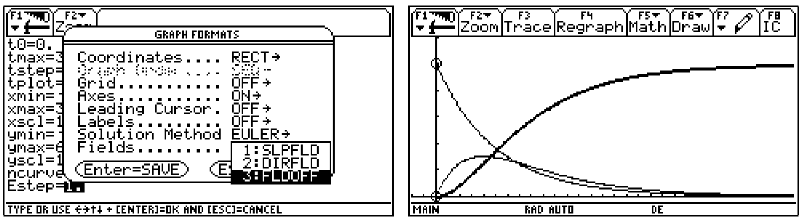

In the next step we define the special values for the parameters, prepare the function editor and the windows variables to obtain the graphic representation.



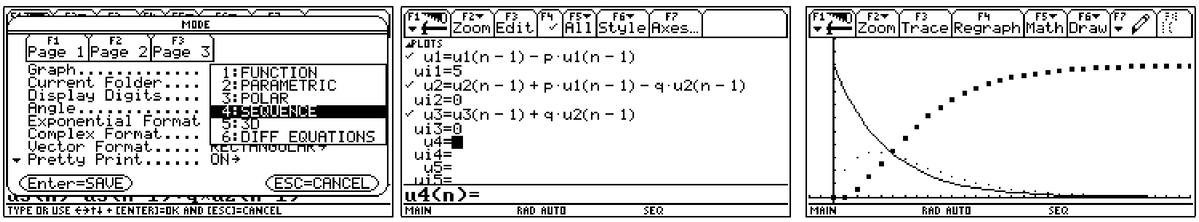




In the DIFF EQUATIONS Mode the function editor looks quite different and we can enter the equations immediately to obtain the graphic solution.




Finally I´ll show treating the problem as a system of difference equations using the sequence mode which is implemented in the first issue of the TI-generations.



I'd like to leave this contribution as it was in 1992. I find it charming working without subscripts (SUB), built-in truth tables and some other improvements which have been done since then. I´ll add possible changes in the DERIVE file. They are red coloured. Finally I try to transfer the LOGIC to the TI. Josef

# Logic with DERIVE

Dr. Karl Fuchs, Salzburg

(Dr. Fuchs uses DERIVE's built in Logical Operators and the IF-Function to define Conjunction, Disjunction and Negation as functions of two truth values. The truth tables are constructed, Implication and Equivalence will be defined. Some important rules and laws can be proofed. Using these functions truth tables of statements are easy to obtain.)

**PROTOKOLL (Protocol):**

a) Beachte: Kontrollstruktur BEDINGTE VERZWEIGUNG in DERIVE
Controlstructure: CONDITIONAL DECISION in DERIVE.

```
IF(Bedingung,W-Zweig,F-Zweig)
IF(Boolean expression, true-branch, false-branch)

disj(a,b):=if((a=1 or b=1),1,0)

conj(a,b):=if((a=1) and (b=1),1,0)

neg(a):=if(a=1,0,1)
```

```
       DISJ(a, b) :=
         If a = 1 ∨ b = 1
#3:         1
            0

       KONJ(a, b) :=
         If a = 1 ∧ b = 1
#4:         1
            0

       NEG(a) :=
         If a = 1
#5:         0
            1
```

b) Überprüfen der Wertebelegung: (Checking the function values)

```
#6:   DISJ(1, 1) = 1

#7:   CONJ(1, 0) = 1

#8:   NEG(1) = 0

#9:   etc.
```

c) Erstellen einer Wertetabelle: (Creating a value table)

c1) Erstellen einer Belegungsmatrix (Matrix of the domain)

`a:=[[0,0],[0,1],[1,0],[1,1]]`

c2) Erarbeitung einer Wertetabelle: (Generating the table)

$$\left[\begin{array}{c|c} A & \begin{array}{c} \text{disj(a,b)} \\ \text{conj(a,b)} \\ \text{neg(a)} \end{array} \end{array}\right]$$

wobei:
```
a:=element(element(A,i),1)
b:=element(element(A,i),2)
```
mit $1 \le i \le 4$

Zur Vereinfachung der Eingabe definieren wir eine Funktion:

`f(i,j):=element(element(A,i),j)`

Erzeugung der Wertetabelle (truth table) mit VECTOR:

`VECTOR([f(i,1),f(i,2),disj(f(i,1),f(i,2))],i,4)`

#23: $\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$

Ergibt die Wahrheitstafel der Disjunktion.

which gives the truth table of the disjunction.

**ÜBUNGEN (Exercises):**

1) Ersetzen wir die Funktion `disj` durch `conj` und `neg`, so erhalten wir die übrigen Wertetabellen.
   Replacing `disj` by `conj` and `neg` leads to other value tables.

2) Generieren weiterer Verknüpfungen als Module (Funktionen) und Erstellung der entsprechenden Wertetafeln:
   Creating other operations as modules (functions) and generating the respective value tables:

   `impl(a,b):=disj(neg(a),b))` (Implication)

   `equi(a,b):=conj(impl(a,b),impl(b,a))` (Equivalence)

3) Überprüfung der Morganschen Regeln:
   Double check the Laws of Morgan:

```
neg(disj(a,b))=conj(neg(a),neg(b))
neg(conj(a,b))=disj(neg(a),neg(b))
```

   Variante A: Getrennte Erstellung der Wertetafeln für die linke und rechte Seite mit anschließendem Vergleich.
   Distinct value tables for right and left side followed by comparing the results.

   Variante B: Zeige, dass beide Aussagen äquivalent sind.
   Show the equivalence of both sides.

   zB.: `equi(neg(disj(a,b)), conj(neg(a),neg(b)))`

4) Weitere Übungen:
   More exercises:

   - $a \vee (\neg a \wedge b) \leftrightarrow a \vee b$
   - $a \wedge (\neg a \vee b) \leftrightarrow a \wedge b$
   - $(a \vee b) \wedge (a \vee \neg b) \leftrightarrow a$

The editor tried to make handling of the formulas more comfortable and to build up truth tables for more than two variables. So I continue with two more exercises:

5) Give a proof the distributive law: $a \wedge (b \vee c) \leftrightarrow (a \wedge b) \vee (a \wedge c)$

6) Set up the truth table for $(a \rightarrow b) \wedge (b \rightarrow c)$

See the DERIVE session:

#1: Logic with DERIVE

#2: Dr. K. Fuchs, Salzburg

#11: [KONJ(1, 1), KONJ(1, 0), KONJ(0, 1), KONJ(0, 0)]

#12: [1, 0, 0, 0]

#13: [1 ∧ 1, 1 ∧ 0, 0 ∧ 1, 0 ∧ 0] = [1, 0, 0, 0]

#14: [NEG(1), NEG(0)]

#15: [0, 1]

See the difference applying NOT on 0, 1 and on true, false!!

#16: [¬ 1, ¬ 0] = [-2, -1]

#17: [¬ true, ¬ false] = [false, true]

#18: (¬ -1) = 0

#19: $v := \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}$

#20: F(i, j) := ELEMENT(ELEMENT(v, i), j)

#21: the truth table of the disjunction:

#22: VECTOR([F(i, 1), F(i, 2), DISJ(F(i, 1), F(i, 2))], i, 4)

#23: $\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$

#24: $\text{VECTOR}([a_1, a_2, \text{DISJ}(a_1, a_2)], a, v) = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$

#25: $\text{TRUTH\_TABLE}(a, b, a \vee b) = \begin{bmatrix} a & b & a \vee b \\ \text{true} & \text{true} & \text{true} \\ \text{true} & \text{false} & \text{true} \\ \text{false} & \text{true} & \text{true} \\ \text{false} & \text{false} & \text{false} \end{bmatrix}$

DERIVE operates with truth-values, whereas Karl´s functions are working with 1 and 0!

#26: Implication and equivalence:

#27: IMPL(a, b) := DISJ(NEG(a), b)

#28: VECTOR([F(i, 1), F(i, 2), IMPL(F(i, 1), F(i, 2))], i, 4)

#29: $\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$

#30: EQUI(a, b) := KONJ(IMPL(a, b), IMPL(b, a))

#31: VECTOR([F(i, 1), F(i, 2), EQUI(F(i, 1), F(i, 2))], i, 4)

#32: $\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Logic implication must be written as IMP (appears as → in #34) and equivalence must be entered as IFF – if and only if – and appears as ↔ in #36).

#33: VECTOR($[a_1, a_2, \text{IMPL}(a_1, a_2)]$, a, v) = $\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$

#34: TRUTH_TABLE(a, b, a → b) =

| a | b | a → b |
|---|---|---|
| true | true | true |
| true | false | false |
| false | true | true |
| false | false | true |

#35: VECTOR($[a_1, a_2, \text{EQUI}(a_1, a_2)]$, a, v) = $\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

#36: TRUTH_TABLE(a, b, a ↔ b) =

| a | b | a ↔ b |
|---|---|---|
| true | true | true |
| true | false | false |
| false | true | false |
| false | false | true |

#37: [a1 := F(i, 1), a2 := F(i, 2), a3 := F(i, 3)]

#38: Rule of Morgan

#39: VECTOR([a1, a2, NEG(DISJ(a1, a2))], i, 4) = $\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

#40: VECTOR([a1, a2, KONJ(NEG(a1), NEG(a2))], i, 4) = $\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

```
#41:  or do it this way:

#42:  VECTOR([a1, a2, EQUI(NEG(KONJ(a1, a2)), DISJ(NEG(a1), NEG(a2)))], i, 4)

       [ 1  1  1 ]
       [ 1  0  1 ]
#43:   [ 0  1  1 ]
       [ 0  0  1 ]

#44:  the third possibility to proof the equivalence: Compare col 3 and 4!

#45:  VECTOR([a1, a2, NEG(KONJ(a1, a2)), DISJ(NEG(a1), NEG(a2))], i, 4)

       [ 1  1  0  0 ]
       [ 1  0  1  1 ]
#46:   [ 0  1  1  1 ]
       [ 0  0  1  1 ]

                                                    [ a      b      ¬ (a ∨ b) ↔ ¬ a ∧ ¬ b ]
                                                    [ true   true          true           ]
#47:  TRUTH_TABLE(a, b, ¬ (a ∨ b) ↔ ¬ a ∧ ¬ b) =    [ true   false         true           ]
                                                    [ false  true          true           ]
                                                    [ false  false         true           ]

#48:  to 4) more exercises:

#49:  VECTOR([a1, a2, DISJ(a1, KONJ(NEG(a1), a2)), DISJ(a1, a2)], i, 4)

       [ 1  1  1  1 ]
       [ 1  0  1  1 ]
#50:   [ 0  1  1  1 ]
       [ 0  0  0  0 ]

#51:  qued, the last two columns are identical!

                                                    [ a      b      a ∨ (¬ a ∧ b) ↔ a ∨ b ]
                                                    [ true   true          true           ]
#52:  TRUTH_TABLE(a, b, a ∨ (¬ a ∧ b) ↔ a ∨ b) =    [ true   false         true           ]
                                                    [ false  true          true           ]
                                                    [ false  false         true           ]

                                                    [ a      b      a ∧ (¬ a ∨ b) ↔ a ∧ b ]
                                                    [ true   true          true           ]
#53:  TRUTH_TABLE(a, b, a ∧ (¬ a ∨ b) ↔ a ∧ b) =    [ true   false         true           ]
                                                    [ false  true          true           ]
                                                    [ false  false         true           ]

                                                      [ a     b     (a ∨ b)∧(a ∨ ¬ b) ↔ a ]
                                                      [ true  true        true            ]
#54:  TRUTH_TABLE(a, b, (a ∨ b) ∧ (a ∨ ¬ b) ↔ a) =    [ true  false       true            ]
                                                      [ false true        true            ]
                                                      [ false  false      true            ]
```

For didactical reasons it might be useful to present intermediate results, so split up the truth table. For task5) and 6) we have to extend matrix v to 3 columns for truth values for a, b and c.

```
#55:  TRUTH_TABLE(a, b, a ∨ b, a ∨ ¬ b, (a ∨ b) ∧ (a ∨ ¬ b), a)
```

#56:

| a | b | a ∨ b | a ∨ ¬ b | (a ∨ b) ∧ (a ∨ ¬ b) | a |
|---|---|---|---|---|---|
| true | true | true | true | true | true |
| true | false | true | true | true | true |
| false | true | true | false | false | false |
| false | false | false | true | false | false |

#57: v :=

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

```
#58:  VECTOR([a1, a2, a3, KONJ(a1, DISJ(a2, a3)), DISJ(KONJ(a1, a2), KONJ(a1, a3))],
       i, 8)
```

#59:

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

```
#60:  to 6), the truth table - first with 1 and 0, then with true and false:
#61:  VECTOR([a1, a2, a3, KONJ(IMPL(a1, a2), IMPL(a2, a3))], i, 8)
```

#62:

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

#63: TRUTH_TABLE(a, b, c, (a → b) ∧ (b → c))

#64:

| a | b | c | (a → b) ∧ (b → c) |
|---|---|---|---|
| true | true | true | true |
| true | true | false | false |
| true | false | true | false |
| true | false | false | false |
| false | true | true | true |
| false | true | false | false |
| false | false | true | true |
| false | false | false | true |

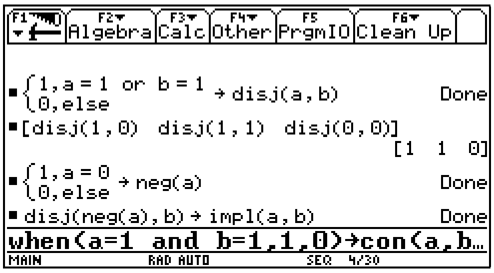
## How can we transfer this to the CAS-TI devices?

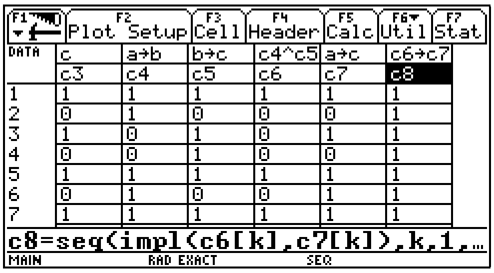
Josef Böhm

If we would like to work with 0 and 1 instead of false and true, we have to define the Boolean functions. (conj is not allowed, because `conj` is an implemented function for the conjugate complex).

Let us prove that a → b ∧ b → c implies a → c.

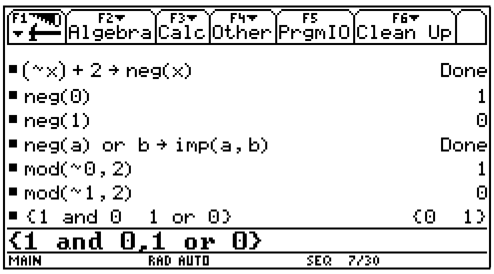
We define disjunction and negation, which are subsequently used to define implication:

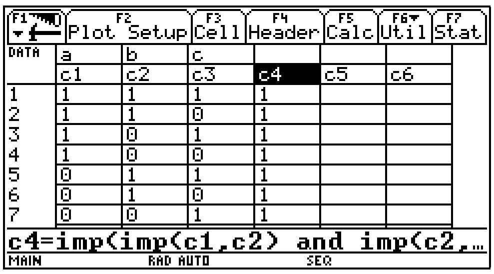



```
c4=seq(impl(c1[k],c2[k],k,1,dim(c1))
c5=seq(impl(c2[k],c3[k],k,1,dim(c1))
c6=seq(con(c4[k],c5[k],k,1,dim(c1))
c7=seq(impl(c1[k],c3[k],k,1,dim(c1))
c8=seq(impl(c6[k],c7[k],k,1,dim(c1))
```

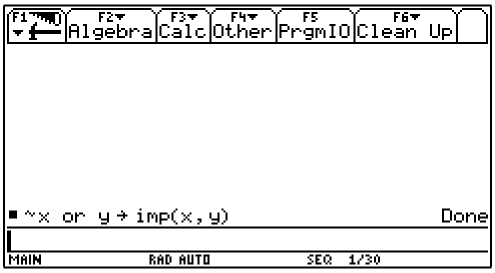
Because we have a when-construct it is necessary to address each single cell using sequences. It is not possible to simply enter `c4=(c1,c2)`! The next pair of screens shows another way to prove the statement.

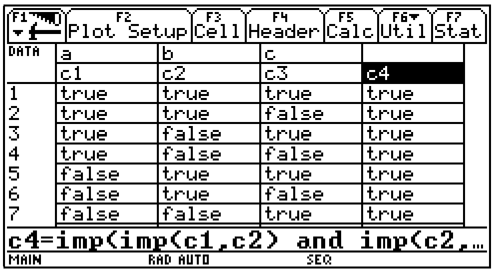


`not x` appears on the screen as `~x`. `not Ø` and `not 1` result in –1 and –2. So I define a function `neg(x)` using truth values 0 and 1 for false and true. Then I define again a function for the implication as `imp(x,y)`. Now it is easy work to open the `Data/Matrix-Editor`, fill columns `c1`, `c2` and `c3` to obtain all possible combinations for the truth values. In column `c4` I enter the header as `imp(imp(c1,c2) and imp(c2,c3),imp(c1,c3))` to find `true` in all cells.

If you prefer working with truth values `true` and `false`, the you don´t need the neg-function:



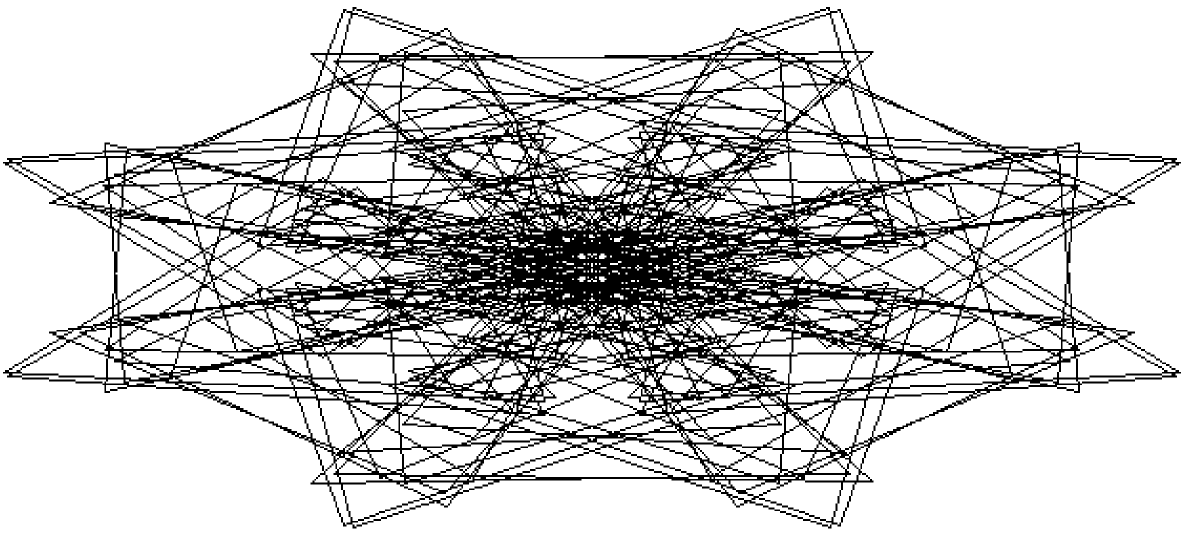

I don´t know how you think, but I find it boring to fill in the ones and zeros, the trues and falses in columns c1 to c3. It it a nice question to ask the students creating sequences which do this job.

Proposal for the solutions:

```
c1=seq(mod(ceiling(k/4),2),k,1,8)
c2=seq(mod(ceiling(k/2),2),k,1,8)
c3=seq(mod(k,2),k,1,8)
```

And for showing truth values true and false:

```
c1=seq(mod(ceiling(k/4),2)=1,k,1,8),
c2=seq(mod(ceiling(k/2),2)=1,k,1,8)
c3=seq(mod(k,2)=1,k,1,8)
```



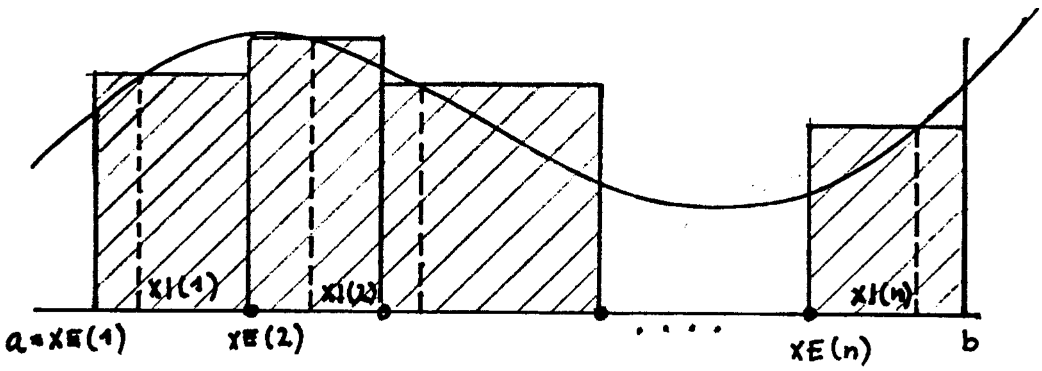
Maurer 11

# Riemann at Random with *DERIVE*

Josef Böhm, Würmla

In the frame of the Spring School in Krems I had the honour and pleasure to give a lecture on the visualization of the defined integral. I worked with various kkinds of decompositions using *DERIVE*`s graphic and calculating abilities. One idea was to use a kind of random decomposition into stripes, but the then latest version of *DERIVE* was not installed on the network, so I couldn't show this part of my work. One of the most interesting parts of version 2.10 or higher is the existence of some random functions.

Well, this is my idea: I want to calculate approximatively the area under the graph of f($x$) from the left border $x = a$ to the right border $x = b$.




Original sketch from 1992

I define an arbitrary vector `ints` = [$xe_1$, $xe_2$, ....., $xe_{n+1}$] with $xe_1 = a$, $xe_{n+1} = b$ and $xe_k < xe_{k+1}$ for $k$ = 1,..,$n$. This vector (list) can be entered by hand or automatically generated as a list of random values using `dec(a,b,n)`. Function `pts(ints)` creates list `intpts` of random values $xi_k$ with $xe_k < xi_k < xe_{k+1}$.
`sum_plot` is a vector consisting of the vertices of the rectangles $(xe_{k+1} - xe_k) \times f(xi_k)$. You can plot the rectangles together with the function graph and then calculate the sum of their areas using `sum_val` as a first approximation of the area. As another approach I can use `int_halves` in order to split the intervals into equal halves to obtain the next decomposition -with the double number of rectangles.

For plotting disable "`Automatically change color of new plots`" in Option Display Color and fix your colour. Set the point Mode to Connected and size Small. Simplifying `dsum` set Option Notation Style Decimal and Digits 3 to save both time and memory space.

RIEMRAND.MTH is possible with *DERIVE* version 2.10 or higher, because of the need of random numbers. I think the file and my sketch are explaining the procedure.

```
#1:   RANDOM(0)

#2:   "First simplify expression #1"

#3:   [PrecisionDigits := 6, Notation := Decimal, NotationDigits := 6]
```

#4: $f(x) := \frac{x^2}{2} + \frac{1}{2}$

```
      dec(a, b, n) :=
        Prog
#5:       ints := SORT(APPEND([a, b], VECTOR(a + RANDOM(1)·(b - a), i, n - 1)))
          RETURN "Intervals in stored in ints"
```

#6: dec(-1, 2.5, 10) = Intervals in stored in ints

#7: ints = [-1, -0.772119, -0.626015, -0.303355, 0.350448, 0.435068, 0.693916, 1.14083, 2.02746, 2.46438, 2.5]

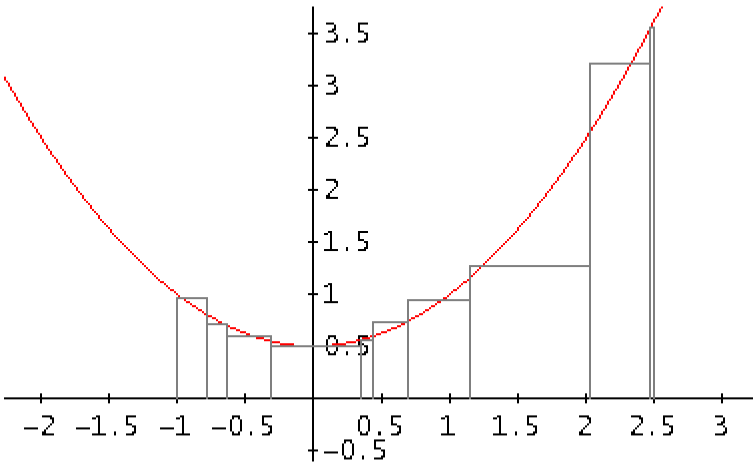
```
      pts(v) :=
        Prog
#8:       intpts := VECTOR(v↓k + (v↓(k + 1) - v↓k)·RANDOM(1), k, DIM(v) - 1)
          "points in intpts"
```

#9: pts(ints) = points in intpts

#10: intpts = [-0.95545, -0.641938, -0.450961, -0.0251856, 0.353458, 0.688806, 0.945258, 1.23555, 2.33067, 2.47064]

#11: $\text{sum\_plot} := \text{VECTOR}\left(\begin{bmatrix} \text{ints}_k & 0 \\ \text{ints}_k & f(\text{intpts}_k) \\ \text{ints}_{k+1} & f(\text{intpts}_k) \\ \text{ints}_{k+1} & 0 \end{bmatrix}, k, \text{DIM(ints)} - 1\right)$

#12: sum_plot




#13: $\text{sum\_val} := \sum_{k=1}^{\text{DIM(intpts)}} (\text{ints}_{k+1} - \text{ints}_k)\cdot f(\text{intpts}_k)$

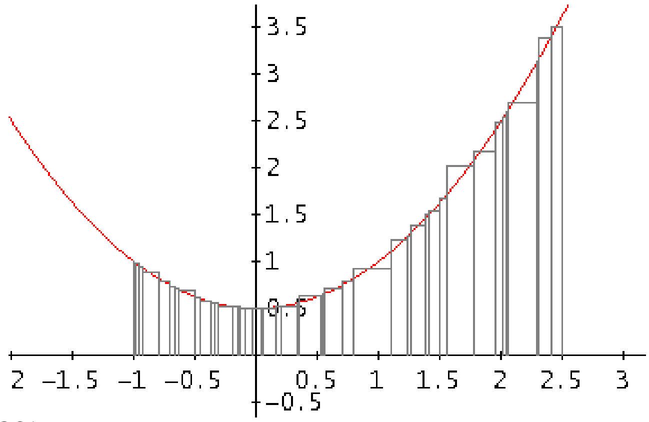
#14: sum_val = 4.15562

All functions needed are prepared now and it is easy to start another run with the same number of random rectangles or – as we are doing now – rising their number up to 50 followed by 200.

#15: dec(-1, 2.5, 50) = Intervals in stored in ints

#16: pts(ints) = points in intpts

#17: sum_plot

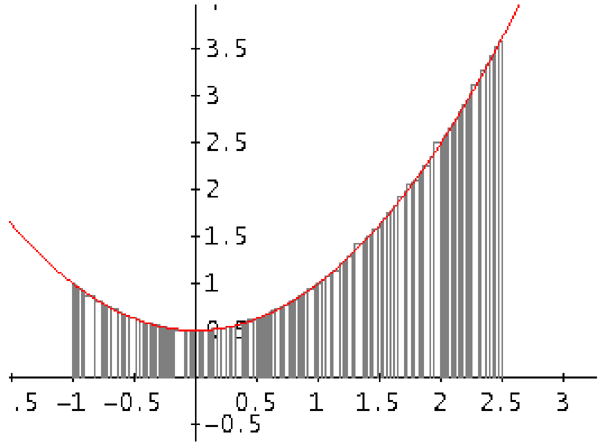


#18:   sum_val = 4.50361

#19:   dec(−1, 2.5, 200) = Intervals in stored in ints

#20:   pts(ints) = points in intpts

#21:   sum_plot



Let´s first compare with the exact value and then have 1000 rectangles:

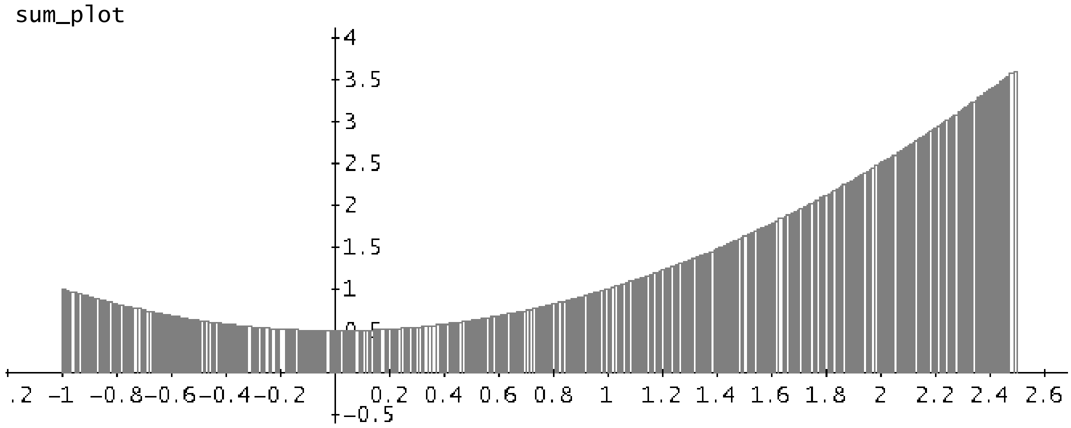
#22:   sum_val = 4.52379

#23:   $\int_{-1}^{2.5} f(x)\,dx = 4.52083$

#24:   dec(−1, 2.5, 1000) = Intervals in stored in ints

#25:   pts(ints) = points in intpts

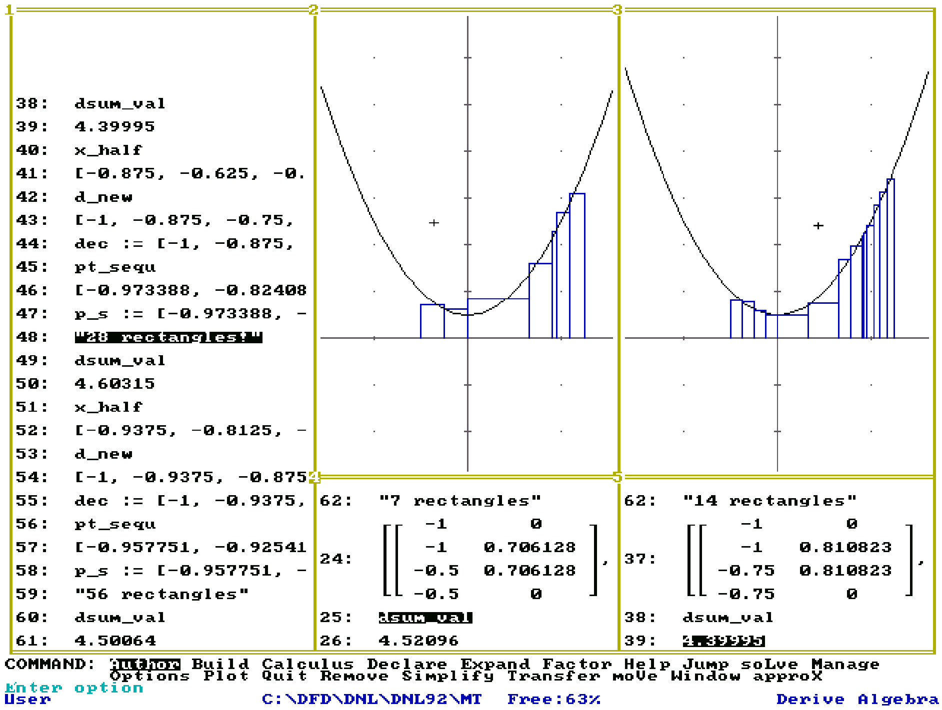
#26:   sum_plot



#27:   sum_val = 4.52096

This is the other approach by entering the first self defined decomposition followed by splitting them:

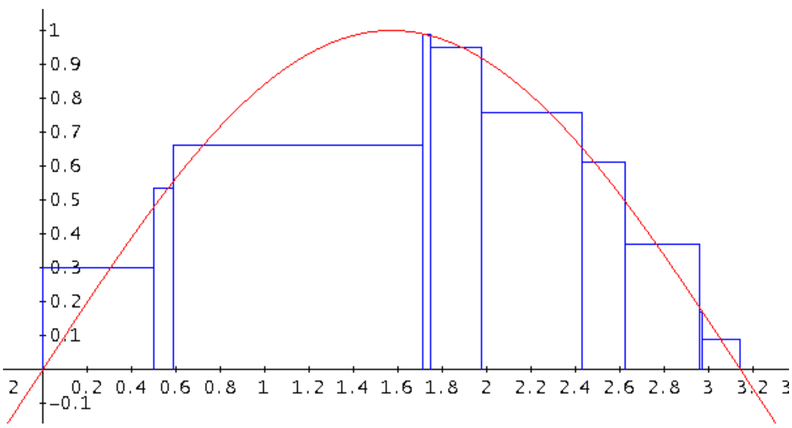
```
      int_halves(v, aux) :=
        Prog
#28:      aux := VECTOR((v↓k + v↓(k + 1))/2, k, DIM(v) - 1)
          ints := SORT(APPEND(v, aux))
          RETURN "Intervals stored in ints"

#29:  Precision := Approximate

#30:  Notation := Decimal
```




A screen shot of the original version from the 1992 RIEMANN.MTH using this approach.

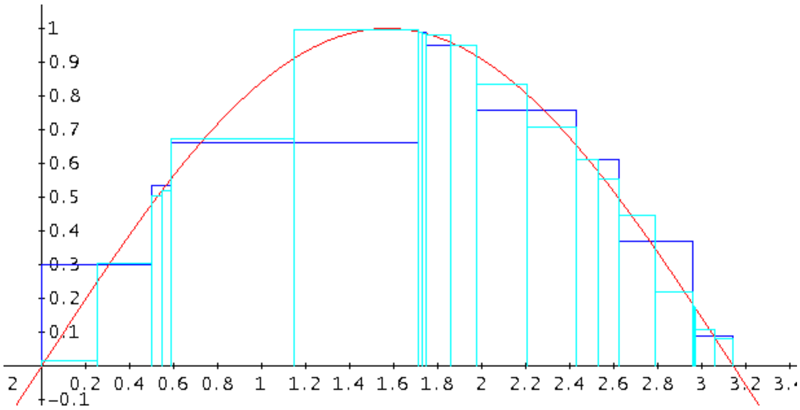
I'd like to demonstrate this using another example. `ints` is a free chosen decomposition of the interval [0, π].

```
#31:  f(x) := SIN(x)

#32:  ints := [0, 0.502221, 0.587468, 1.71114, 1.74736, 1.97597, 2.43179, 2.62161,
        2.95654, 2.97323, 3.14159]

#33:  pts(ints) = points in intpts

#34:  sum_plot
```

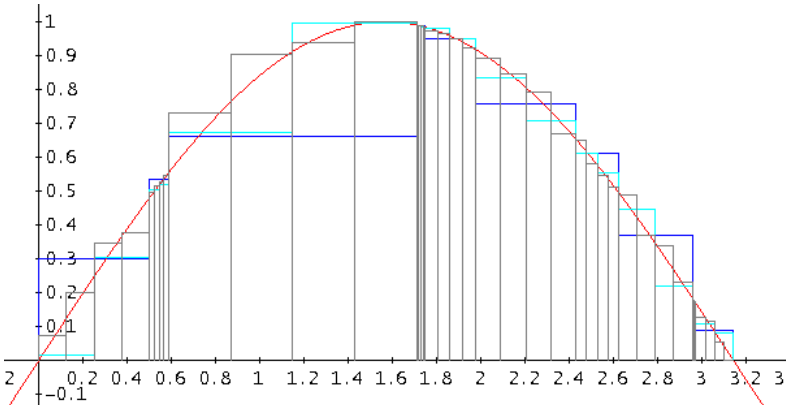


```
#35:  sum_val = 1.79542
```

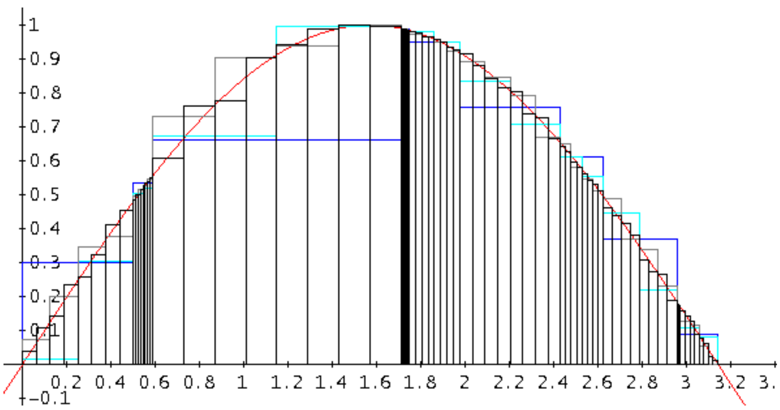
```
#36:  int_halves(ints) = Intervals stored in ints

#37:  ints = [0, 0.25111, 0.502221, 0.544844, 0.587468, 1.1493, 1.71114, 1.72925,
        1.74736, 1.86167, 1.97597, 2.20388, 2.43179, 2.5267, 2.62161, 2.78908,
        2.95654, 2.96489, 2.97323, 3.05741, 3.14159]

#38:  pts(ints) = points in intpts

#39:  sum_plot
```



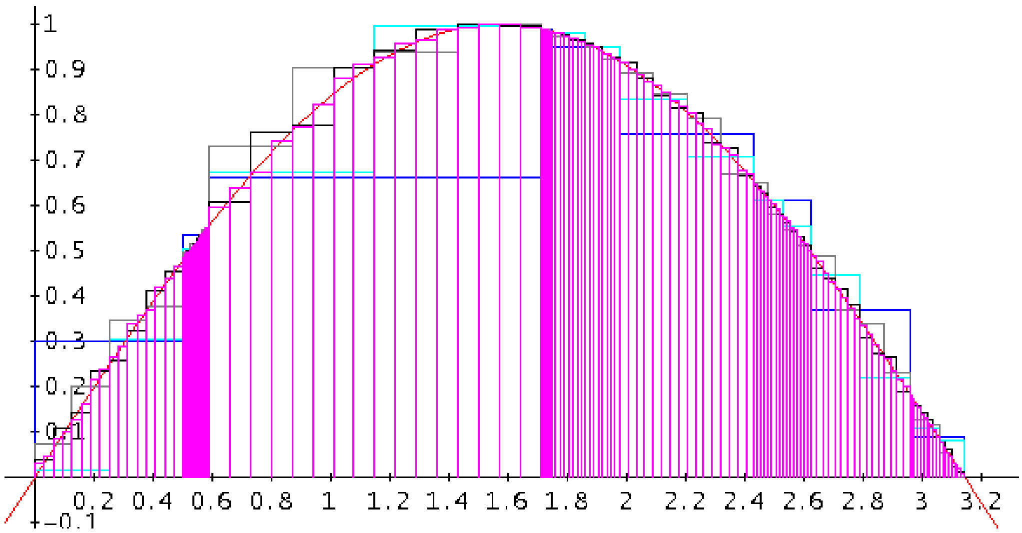
```
#40:  sum_val = 1.91111

#41:  int_halves(ints) = Intervals stored in ints

#42:  pts(ints) = points in intpts

#43:  sum_plot
```



```
#44:  sum_val = 2.03421

#45:  int_halves(ints) = Intervals stored in ints

#46:  pts(ints) = points in intpts

#47:  sum_plot
```



```
#48:   sum_val = 2.00724

#49:   int_halves(ints) = Intervals stored in ints

#50:   pts(ints) = points in intpts

#51:   DIM(intpts) = 160

#52:   sum_plot
```



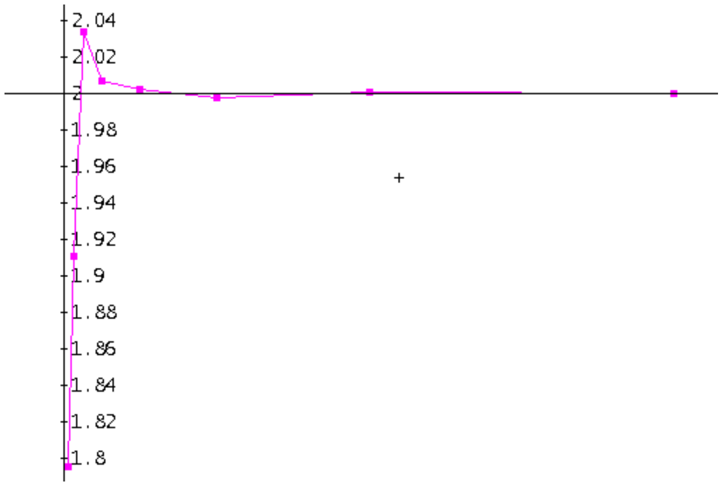

```
#53:   sum_val = 2.00247

#54:   int_halves(ints) = Intervals stored in ints

#55:   pts(ints) = points in intpts

#56:   sum_val = 1.99784

#57:   int_halves(ints) = Intervals stored in ints

#58:   pts(ints) = points in intpts

#59:   sum_val = 2.00046

#60:   int_halves(ints) = Intervals stored in ints

#61:   pts(ints) = points in intpts

#62:   sum_val = 2.00031
```

I collect the values for the areas depending on the number of rectangles to provide a graphic demonstration of the convergence of the process (together with the theoretical value for the area, which is 2).

#63:

| | |
|---|---|
| 10 | 1.79542 |
| 20 | 1.91111 |
| 40 | 2.03421 |
| 80 | 2.00724 |
| 160 | 2.00247 |
| 320 | 1.99784 |
| 640 | 2.00046 |
| 1280 | 2.00031 |

#64:   2

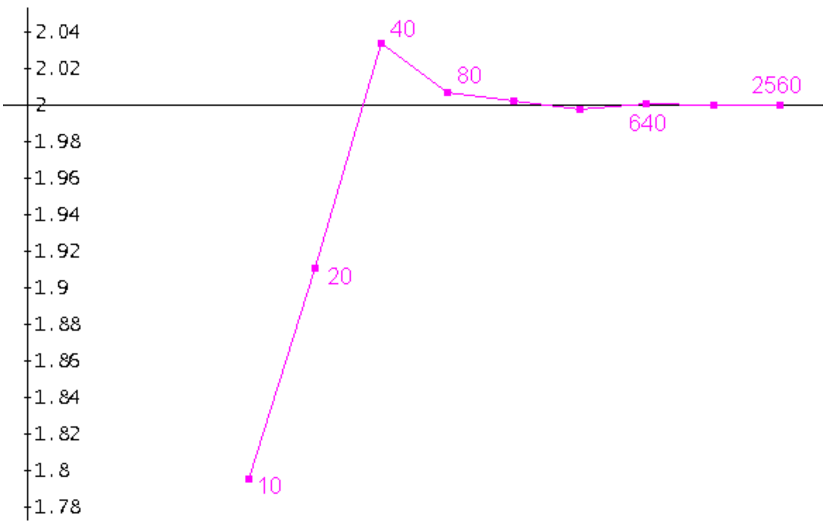



The large values for n make it difficult to show the process nicely. I add another step and take the opportunity to demonstrate the advantage of a log-scale.

```
#65:  int_halves(ints) = Intervals stored in ints

#66:  pts(ints) = points in intpts

#67:  DIM(intpts) = 2560

#68:  sum_val = 2
```

#69:

| | |
|---|---|
| LOG(10) | 1.79542 |
| LOG(20) | 1.91111 |
| LOG(40) | 2.03421 |
| LOG(80) | 2.00724 |
| LOG(160) | 2.00247 |
| LOG(320) | 1.99784 |
| LOG(640) | 2.00046 |
| LOG(1280) | 2.00031 |
| LOG(2560) | 2 |



# Goldbach´s Conjecture

Günter Scheu, Pforzheim, Germany

In the 18th century a mathematician named Charles Goldbach conjectured that *every even integer greater than 4 can be expressed as the sum of two odd primes*. Although many mathematicians attempted to prove or to disprove Goldbach´s conjecture, no one has been successful up to now.

In 1992 Günter had to preload the utility file SET.MTH for manipulating sets of expressions represented as vectors. Günter needs INTERSECTION of lists. `u INTERSECTION v` is now an implemented command – but only for sets `u, v` of expressions given between braces `{}`.

For using *DERIVE* 5 & 6 function `inters(vector_1,vector_2)` must be defined. Günter´s verification of the conjecture might be much easier by means of *DERIVE* 5 & 6 but it is charming to follow the process how it was done with *DERIVE* 2. (Josef)

The file is the printed MTH-file in 1992 style but can be used with the recent *DERIVE* versions.

```
"File GOLDBACH.MTH, G. Scheu"

"Test of Goldbach's conjecture"

PZ(x):=NEXT_PRIME(x)

"Function to create pairs of primes"

e(z,i):=IF(PZ(i-1)=i AND PZ(z-i-1)=z-i,[i,z-i],IF(i>1,e(z,i-1),"none"))

VECTOR(e(12,k),k,5,3,-2)=[[5,7],"none"]

VECTOR(e(52,k),k,25,3,-2)=[[23,29],[23,29],[11,41],[11,41],[11,41],[11,41],
          [11,41],[11,41],[5,47],[5,47],[5,47],"none"]

"Function to create primes"

f(z,i):=IF(PZ(i-1)=i AND PZ(z-i-1)=z-i,i,IF(i>1,f(z,i-1),0))

"vector of all primes but with repetitions"

VECTOR(f(12,k),k,5,3,-2)=[5,0]

VECTOR(f(52,k),k,25,3,-2)=[23,23,11,11,11,11,11,11,5,5,5,0]

v:=[23,23,11,11,11,11,11,11,5,5,5,0]

"Vector with X for multiple primes"

w=[23,"X",11,"X","X","X","X","X",5,"X","X","X"]

w:=VECTOR(IF(i>1 AND i<DIM(v) AND NOT(v SUB i=v SUB (i-1)) AND v SUB i/=0,
      v SUB i,IF(i>1 OR v SUB i=0,"X",v SUB i)),i,DIM(v))

"some auxiliary functions:"

empty(v_):=DIM(v_)=0

find(e_,v_):=IF(empty(v_) OR e_=FIRST(v_),v_,find(e_,REST(v_)),
             find(e_,REST(v_)))

found(e_,v_):=DIM(find(e_,v_))/=0

inters(u_,v_):=IF(empty(u_),u_,IF(found(FIRST(u_),v_),ADJOIN(FIRST(u_),
               inters(REST(u_),v_)),inters(REST(u_),v_)))
```

```
inters(v,w)=[23,23,11,11,11,11,11,11,5,5,5]

inters(w,v)=[23,11,5]

"as you can see inters is not commutative!"

u:=[23,11,5]

"Calculating the pairs of primes:"

goldpairs(u,z):=VECTOR([u SUB i,z-u SUB i],i,DIM(u))

goldpairs(u,52)=[[23,29],[11,41],[5,47]]

"These are the pairs of primes which add to 52"

"Let's take another odd number, say 100."

v:=VECTOR(f(100,k),k,49,3,-2)

v=[47,47,41,41,41,29,29,29,29,29,29,17,17,17,17,17,17,11,11,11,3,3,3,3]

w=[47,"X",41,"X","X",29,"X","X","X","X","X",17,"X","X","X","X","X",11,"X",
   "X",3,"X","X","X"]

inters(w,v)=[47,41,29,17,11,3]

goldpairs([47,41,29,17,11,3],100)=[[47,53],[41,59],[29,71],[17,83],
          [11,89],[3,97]]

v:=VECTOR(f(222,k),k,111,3,-2)

goldpairs(inters(w,v),222)=[[109,113],[83,139],[73,149],[71,151],[59,163]
                  ,[43,179],[41,181],[31,191],[29,193],[23,199],[11,211]]

"Try to compress the whole procedure into one function"

"or write a short program with the recent DERIVE version!"
```

$$\text{goldbach(200)} = \begin{bmatrix} 3 & 197 \\ 7 & 193 \\ 19 & 181 \\ 37 & 163 \\ 43 & 157 \\ 61 & 139 \\ 73 & 127 \\ 97 & 103 \end{bmatrix}$$

This might be a good example for beginners in programming with DERIVE.

I´ll give a little hint: `PRIME?(u)` proves to be a useful command – which was not available in 1992!

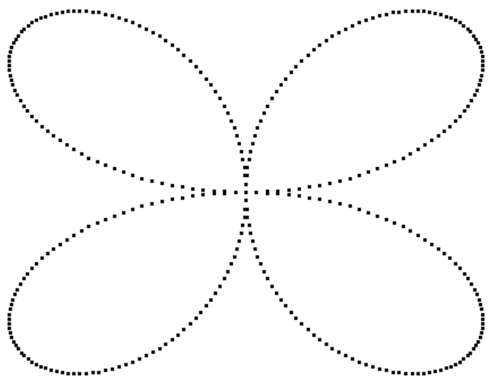
# MAURER Roses in my *DERIVE* Garden

Josef Böhm, Würmla

In the last DNL I showed some examples of so called "Maurer Roses". I first met these pretty flowers in Stan Wagon's book "Mathematica in Action"[1]. It is very easy to plant and to grow them.

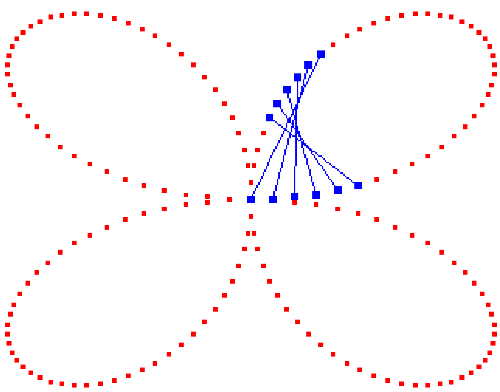
Take any *n*-leafed rose with its polar equation $r = \sin(n\ t)$. If *n* is odd then you obtain an *n*-leafed rose with $0 \le t \le \pi$ and you get an 2*n*-leafed rose if *n* is even with $0 \le t \le 2\pi$.

We take a walk along the rose in steps of *d*° for the parameter *t*. If GCD(360,*d*) = 1 then we will return to the starting point after 360 steps.

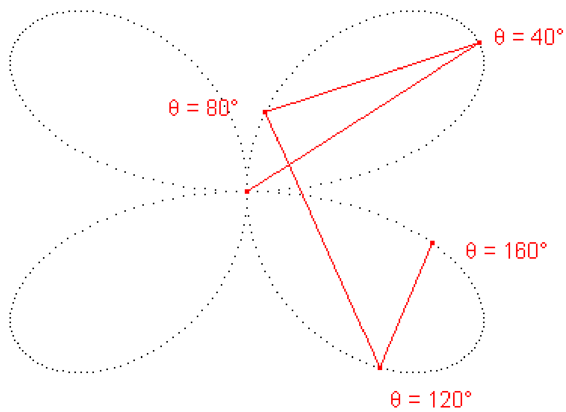


Maurer 1

In this case we will visit every point belonging to an integer value for t°. But if GCD(360,*d*) ≠ 1 then several points will remain unvisited. See the first steps in Maurer 2 ....



Maurer 2




Maurer 3

But there is another possibility to create this family of lines. Use a loop for *t* (step *z*) and calculate the connecting lines of the points [*r*(*t*),*t*)] and [*r*(*t*+*d*),*t*+*d*]. See the first steps in Maurer 3.

In the 1992 DERIVE-print the functions are called `ROSE_WALK_E` or `ROSE_WALK_O` for the walk for *n* = even or *n* = odd and `ROSE_EVEN` and `ROSE_ODD` if using pairs of points.
(I learned a lot during my DERIVE-life and can now use one function for both cases even and odd.)

It is very interesting to compare the generating process of the plots of the simplified expressions `rose(6,4,71,0,360,1)` and `rose_walk(6,4,71,360)`. The results are equal pictures. If there are problems with "Memory full" then split `rose(6,4,71,0,360,1)` into two expressions: `rose(6,4,71,0,180,1)` and `rose(6,4,71,180,360,1)`. (This will not happen with recent versions of DERIVE.)

#1: [DisplayFormat:=Compressed, PrecisionDigits:=6, Notation:=Decimal, NotationDigits:=6]

#5: $\text{leaf}(r,n,\text{step}\theta):=\text{VECTOR}\left(\left[r\cdot\text{SIN}\left(\frac{n\cdot t°}{2-\text{MOD}(n,2)}\right),t°\right],t,0,180\cdot(2-\text{MOD}(n,2)),\text{step}\theta\right)$

See how to include the even- and odd-case in one function. Try `leaf(6,5,4)` and `leaf(6,8,4)`

Maurer 1 and Maurer 2:

#8: leaf(8,4,1)

#9: leaf(6,4,2)

#10: $\text{rose}(r,n,d,\text{start},\text{end},\text{step}\theta):=\text{VECTOR}\left(\begin{bmatrix} r\cdot\text{SIN}\left(\frac{n\cdot t°}{2-\text{MOD}(n,2)}\right) & t° \\ r\cdot\text{SIN}\left(\frac{n\cdot(t+d)°}{2-\text{MOD}(n,2)}\right) & (t+d)° \end{bmatrix}, t,\text{start},\text{end},\text{step}\theta\right)$

#11: rose(6,4,70,0,10,2)

Maurer 4 and Maurer 5

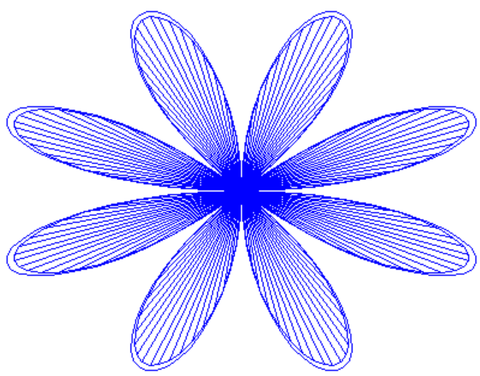
#12: [rose(6,8,180,0,180,2),leaf(6,8,2),leaf(6.2,8,2)]

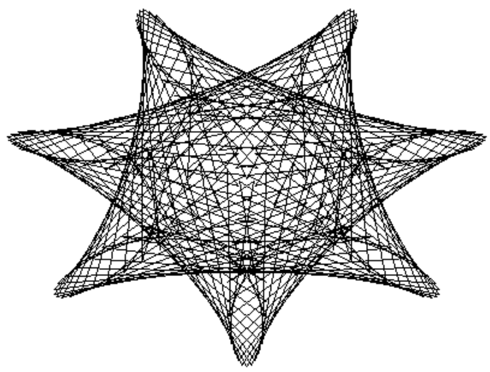
#13: rose(6,7,111,0,180,1)

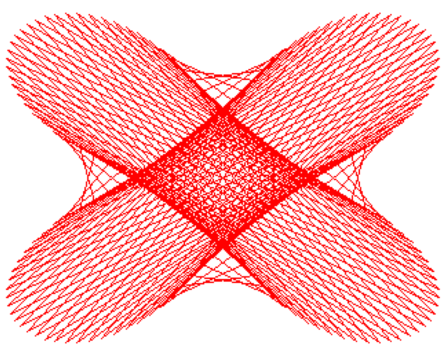
Maurer 6a and Maurer 6b

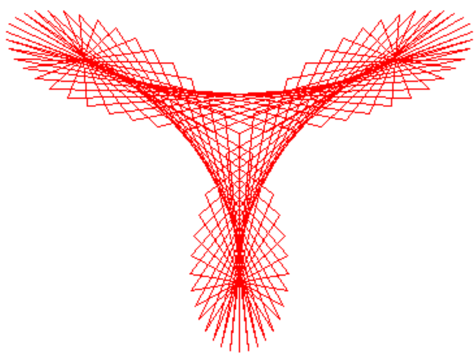
#14: rose(6,4,150,0,360,2)

#15: rose(6,3,150,0,360,2)









#16: $\text{rose\_walk}(r,n,d,steps):=\text{VECTOR}\left(\left[r\cdot \text{SIN}\left(\frac{n\cdot k\cdot d°}{2-\text{MOD}(n,2)}\right), k\cdot d°\right], k, 0, steps\right)$

Mauer 3

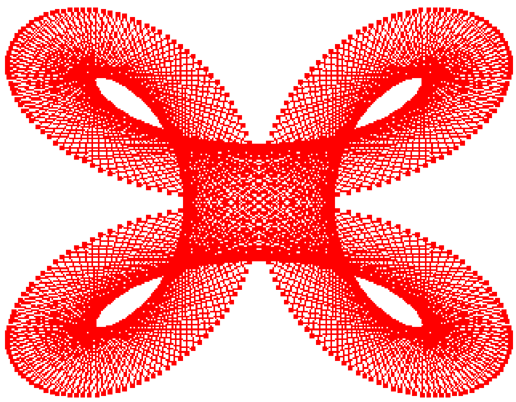
#17: leaf(6,4,1)

#18: rose_walk(6,4,40,4)

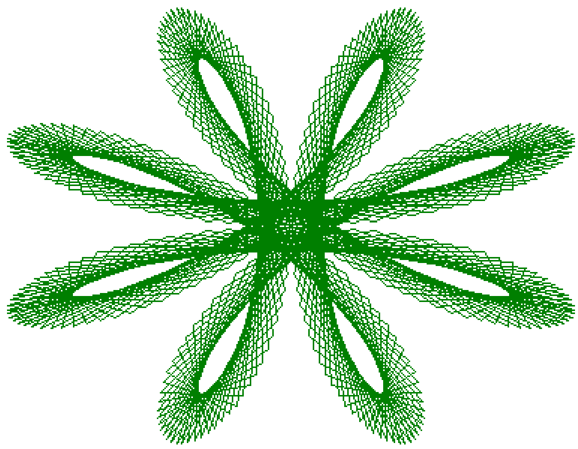
Try rose_walk(6,4,40,9) and rose_walk(6,4,39,360)

Try rose_walk(6,4,42,360) and rose_walk(6,4,37,360)

#21: rose_walk(6,4,41,360)



#24: rose_walk(6,8,19,360)



#25: rose_walk(6,4,39,4)

Experiment with slider bars! Which parameters can "slide" and which can not!

#26: rose(6,8,d_,0,180,2)

#27: rose(6,8,20,0,180,2)

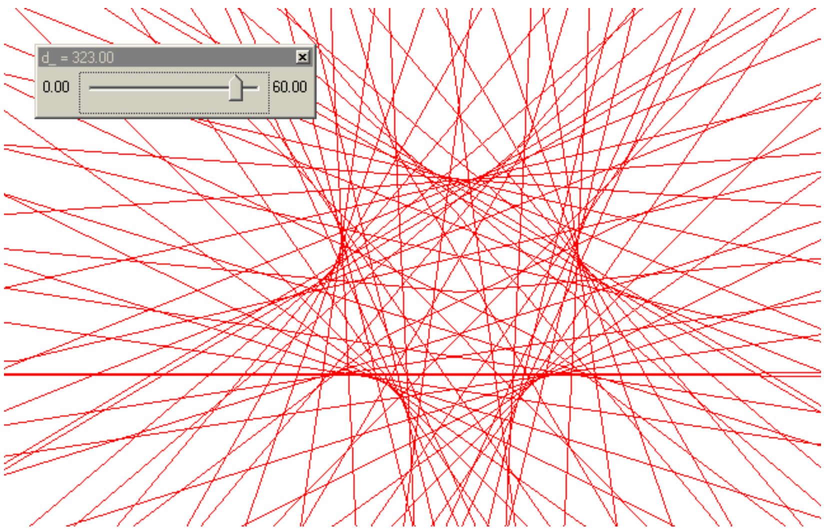
#28: rose(6,8,5,0,180,d_)

#30: rose(6,8,5,0,d_,2)

#32: rose(6,d_,180,0,180,2)

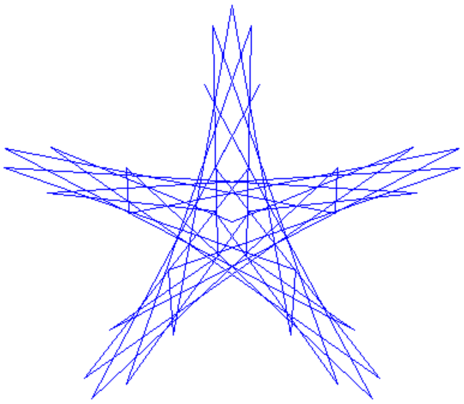
#35: rose(6,d_,6,0,180,2)

A deep look into the interior of a Maurer Rose – on the title page is a true rose from my garden.



Maurer 7

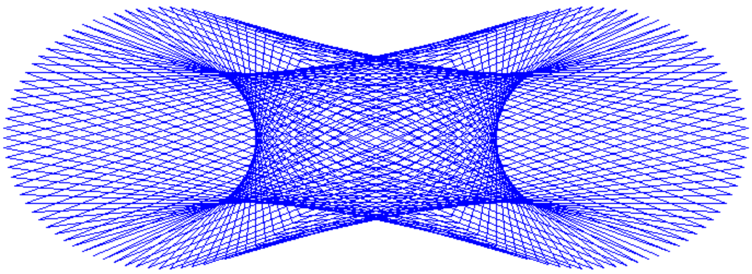
```
#36:  rose_walk(6,5,25,72)
```



```
#37:  rose_var(r,a,b,d,start,end,stepθ):=VECTOR([[r·SIN(a·t°), COS(b·t°)], [r·SIN(a·(t+d)°), COS(b·(t+d)°)]],
          t,start,end,stepθ)
```

Maurer 8

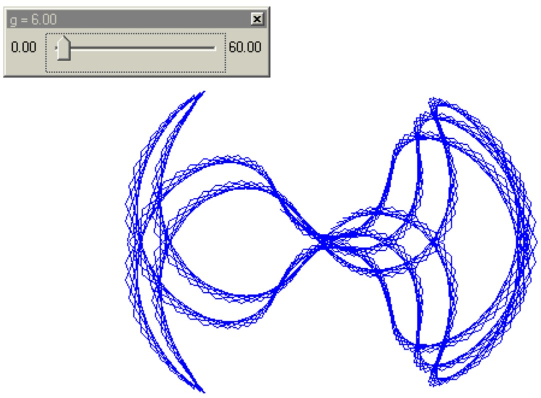
```
#38:  rose_var(5,1,1,120,0,360,2)
```

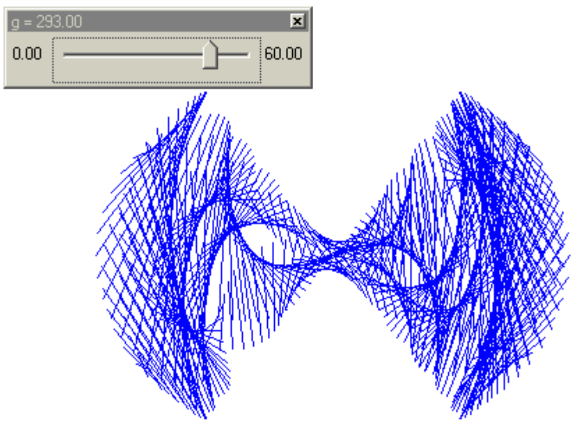


```
#39:  rose_var(5,2,4,103,0,360,1)

#40:  rose_var(5,2,4,g,0,360,1)
```

Use a slider bar for parameter g with 0 ≤ g ≤ 360 to perform a wonderful metamorphosis:

```
#41:  rose_var(5,2.5,6.5,g,0,360,1)
```

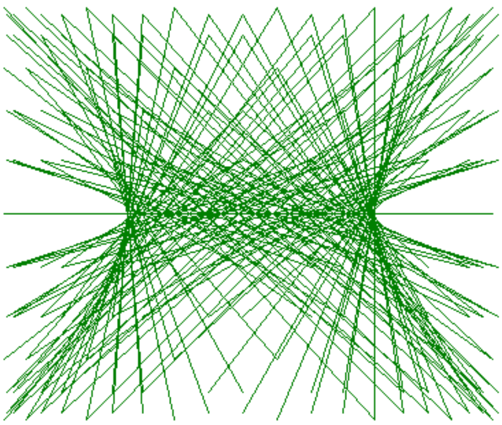


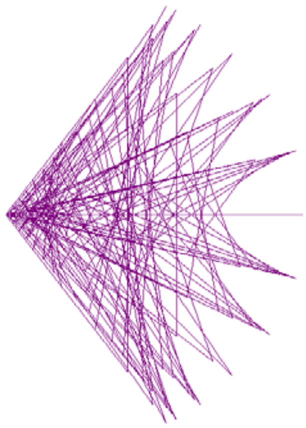
#42: rose_var(5,1.7,3,120,60,0,-2)

#43: $$\text{rose\_var2(r,a,b,d,start,end,step}\theta\text{):=VECTOR}\left(\begin{bmatrix} r\cdot \text{SIN}(a\cdot t°)^2 & \text{COS}(b\cdot t°) \\ r\cdot \text{SIN}(a\cdot (t+d)°)^2 & \text{COS}(b\cdot (t+d)°) \end{bmatrix}, t, start, end, step\theta\right)$$

#44: rose_var2(5,2,7.5,120,0,360,2)



Rectangular coordinates



Polar Coordinates

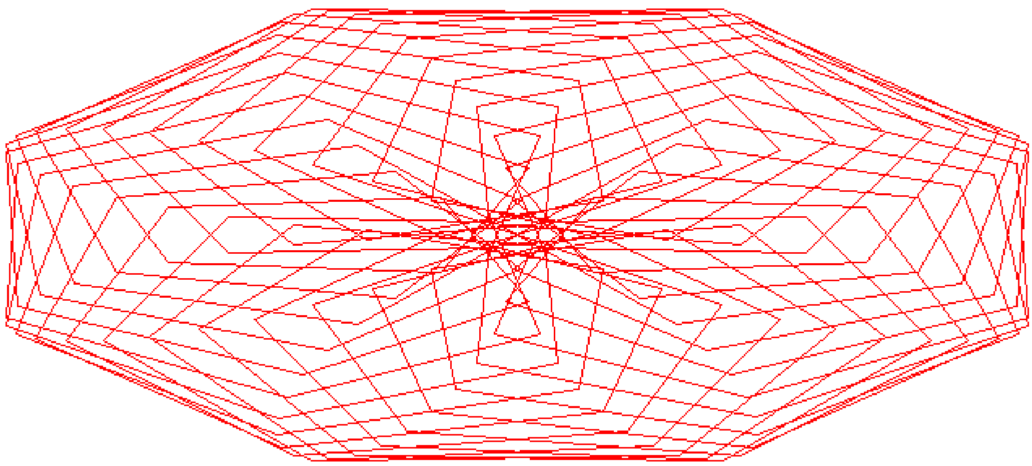
#45: rose_simp(r,a,b,n,d,start:=0,end:=360,stepθ:=1):=VECTOR([SIN(a·n·k·d°)·COS(b·n·k·d°)~ ,k·d°],k,start,end,stepθ)

Maurer 9

#46: rose_simp(5,0.5,3,2,23,0,360,2)

Maurer 10

#47: rose_simp(5,0.5,0.5,4,71,0,360,2)



Maurer 11

#48: rose_simp(3,0.5,3,8,121,0,360,1)

[1] Stan Wagon, *Mathematica in Action*, Freeman & Company, New York 1991

# Esperimenti didattici con un programma di Computer Algebra

Rosamaria Castelletti, Segrate, Italy

I want to thank Mrs Castelletti for sending a copy of an article on teaching mathematics with computers in Italian schools. The original is written in Italian, so I had to translate it into German and I am trying to give here a short summary. I have two reasons for not translating the worksheets Mrs Castelletti has sent: I think that the originals can be understood, and the Italian language is so beautiful.

Mille grazie, editor

The PC has to be established in all higher schools in Italy not for vocational training but as a subject "Informatica". Many questions are still to be answered: how to organize the work in school, how to plan the time tables, how to overcome the lack of technical personnel and how to change the traditional examinations. This was the experience at the Liceo Scientifico Statale "N. Machiavelli" in Pioltello, too. We are looking for a practicable way for our school-classes on the basis of existing facilities. The problems connected with the installation of the programs, the struggle against viruses or malfunctions of the hardware could be solved because of our good cooperation within the school. The teachers think that groups of 3-4 students should work with one PC.

The question, which Computer Algebra (CA) program should be used is still open (1991). We use short BASIC-programs (for exercises, revisions and small simulations). Here we don't see any chance for further development. The teachers have devoted their attention to algebra programs which are able to work with variables and formulas. The first important experiments were made with muMATH-83. Together with a BASIC-program we could represent curves in both Cartesian and Polar forms. DERIVE, a successor of muMATH - muSIMP, seems to come up to our expectations much more. The menu makes working with the program easier and graphical representation is implemented. Last but not least DERIVE provides a driver for our equipment (Olivetti M24). It is easy to switch from CGA (320 x 200, 4 colours) to Olivetti Mode (640 x 200, 2 colours).

At the end of this paper you will find some worksheets. In many cases it is too much work for the teacher to explain the requirements of the program and the mathematical contents at the same time. For that reason we have oroduced worksheets to be shown to groups or full classes on overhead projectors. Afterwards we can ask the students for further experiments. Thus the students are forced into self discipline and it seems to be a good compromise between group work on the one side and the necessity to assess the students' performance.

(to be continued)