

THE BULLETIN OF THE



USER GROUP

+ CAS-TI

C o n t e n t s :

- | | |
|----|---|
| 1 | Letter of the Editor |
| 2 | Editorial - Preview |
| 3 | User Forum |
| | Roland Schröder |
| 5 | The Common Measure |
| | Guido Herweyers |
| 9 | Tutorials for the TI-Nspire - Distributions |
| | Josef Böhm |
| 15 | What's the Time, Grandie? |
| | Peter Lüke-Rosendahl |
| 19 | An Interesting Property of a Triangle |
| | Josef Böhm |
| 26 | Working with DEQME |
| | Johann Wiesenbauer |
| 32 | Titbits 37 |
| 38 | ACA09 - The DERIVE Session |

D-N-L#74	Information	D-N-L#74
----------	-------------	----------

Interesting websites:

<http://www.fachgruppe-computeralgebra.de/cms/tiki-index.php?page=Rundbrief>

You can download the „Rundbriefe“ (mostly in German) starting in December 1987. All documents are in pdf-format. This provides a very interesting history of development and spreading of CAS. The Rundbriefe contain plenty of information about the various computer algebra systems.

The next sites are recommended by Prof. de Villiers. Many thanks for his wonderful and inspiring Newsletter.

I'll give a selection of his recommendations. You can find much more starting with his personal website:

<http://mysite.mweb.co.za/residents/profmd/homepage4.html>

You can download 2 volumes (together nearly 600 pages!) containing the complete proceedings of the ICMI conference (Taiwan May 2009):

ICMI Study 19: Proof and Proving in Mathematics Education

from: <http://ocs.library.utoronto.ca/index.php/icmi/8>

Goto the website of the Mathematical Association of America and download among others an article on Teaching and Learning Differential Equations by Chris Rasmussen and Karen Whitehead:

http://www.maa.org/t_and_l/index.html

http://www.maa.org/t_and_l/sampler/rs_7.html

The next website recommended offers plenty of geometric models and how to create them with paper and scissors:

<http://www.korthalsaltes.com/>

http://www.korthalsaltes.com/three_pyramides_in_a_cube.htm

Download contributions published in the Journal of Mathematics Education (most of them submitted by Chinese researchers:

<http://educationforatoz.com/journalandmagazines.html>

Dear DUG Members,

I apologize for being late with DNL#74. At the end of June and begin of July I attended some interesting conferences (ACA09, CADGME and others) which needed some extra preparations. Then some clarifications were necessary to finalize the DEQME-contribution of this DNL. And as I wrote in my short info our webmaster Walter Wegscheider enjoys his very well deserved holidays. When he will be back from his cruise on the Mediterranean Sea he will upload this issue. I hope that it was worth to wait one month:

ACA09 (Applications of Computer Algebra 2009) was an excellent organized conference which was held in Montreal. Many thanks to Kathleen Pineau, Michel Beaudin and Gilles Picard who made the conference a full success. There were among others a rich Educational Session and a special DERIVE session. You can find the abstracts of the DERIVE-related contributions. DNL#75 will present the full papers.

By the way, don't forget to mark July 2010 in your agenda.

TIME 2010 will be held in Malaga, Spain from 6 - 10 July. We will have four extraordinary keynote speakers:

Bärbel Barzel, Germany

Michel Beaudin, Canada

Colette Laborde, France

Eugenio Roanes, Spain

The Conference website is www.time2010.uma.es.

This DNL contains a variety of articles. You can find another of R. Schröder's projects for the classroom, an interesting paper on the use of the TI-NspireCAS, one of my exercising programs (originally written for my granddaughter), a great triangle problem where you can use various CASs, the first part of a review of Nils Hahnfeld's Differential Equations' package for the handheld and last but not least informs Johann Wiesenbauer in his Titibits 37 (!!!) about other methods for factorizing integers.

So I do hope that everybody will find some useful, interesting or just delighting pages. I wish you all a pretty summer and I am looking forward to meeting you again in fall.



Download all DNL-DERIVE- and TI-files from

<http://www.austromath.at/dug/>

The *DERIVE-NEWSLETTER* is the Bulletin of the *DERIVE & CAS-TI 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 experiences made with *DERIVE*, *TI-CAS* and other CAS as well to create a group to discuss the possibilities of new methodical and didactical manners in teaching mathematics.

Editor: Mag. Josef Böhm
D'Lust 1, A-3042 Würmla
Austria
Phone: ++43-06604070480
e-mail: nojo.boehm@pgv.at

Contributions:

Please send all contributions to the Editor. Non-English speakers are encouraged to write their contributions in English to reinforce the international touch of the *DNL*. It must be said, though, that non-English articles will be warmly welcomed nonetheless. Your contributions will be edited but not assessed. By submitting articles the author gives his consent for reprinting it in the *DNL*. The more contributions you will send, the more lively and richer in contents the *DERIVE & CAS-TI Newsletter* will be.

Next issue: September 2009
Deadline 15 August 2009

Preview: Contributions waiting to be published

Some simulations of Random Experiments, J. Böhm, AUT, Lorenz Kopp, GER
Wonderful World of Pedal Curves, J. Böhm
Tools for 3D-Problems, P. Lüke-Rosendahl, GER
Financial Mathematics 4, M. R. Phillips
Hill-Encryption, J. Böhm
Simulating a Graphing Calculator in *DERIVE*, J. Böhm
Henon, Mira, Gumowski & Co, J. Böhm
Do you know this? Cabri & CAS on PC and Handheld, W. Wegscheider, AUT
Steiner Point, P. Lüke-Rosendahl, GER
Overcoming Branch & Bound by Simulation, J. Böhm, AUT
Diophantine Polynomials, D. E. McDougall, Canada
Graphics World, Currency Change, P. Charland, CAN
Cubics, Quartics – interesting features, T. Koller & J. Böhm
Logos of Companies as an Inspiration for Math Teaching
Exciting Surfaces in the FAZ / Pierre Charland's Graphics Gallery
BooleanPlots.mth, P. Schofield, UK
Old traditional examples for a CAS – what's new? J. Böhm, AUT
Truth Tables on the TI, M. R. Phillips
Advanced Regression Routines for the TIs, M. R. Phillips
Where oh Where is IT? (GPS with CAS), C. & P. Leinbach, USA
Embroidery Patterns, H. Ludwig, GER
Mandelbrot and Newton with *DERIVE*, Roman Hašek, CZ
Snail-shells, Piotr Trebisz, GER
A Conics-Explorer, J. Böhm, AUT
Coding Theory for the Classroom?, J. Böhm, AUT
Tutorials for the NSpireCAS, G. Herweyers, BEL
Some Projects with Students, R. Schröder, GER
Runge-Kutta Unveiled, J. Böhm, AUT
The Horror Octahedron, W. Alvermann, GER
and others

Impressum:

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

Tom Barbara, Oregon, USA

Hi Josef,

Thanks for adding me to the group.

I am a long time user of Derive (from Version 1). However, I did not use it that much in recent years, although I did have the first windows version of the software. I recently purchased the latest version that is available. Besides the general algebraic tools I used derive's matrix capabilities for problems in finite dimensional quantum theory as applied to Nuclear Magnetic Resonance. Back then I lobbied the Derive folks to include a function for taking the Kronecker Product of matrices,

And they were happy to comply. However, I now find that this feature is no longer supplied with the software. I believe this came as an additional operator that one could load from a library and I guess it fell off the list of those that were included.

Perhaps some of the users recall this function and still have a copy of the library it came in? Or perhaps someone has invented their own operator or knows how to use the operators that are supplied to write a Kronecker Product function. Back then, it did not appear to be possible with the current tools and that is why I suggested to Derive that they supply one.

Thanks,

Tom B

DNL:

Dear Tom.

I didn't find the Kronecker product among my many DERIVE files.

And I must admit that I didn't know about this special matrix product, but I could find its definition.

So I took the challenge trying to produce the respective function (see the attached file).

I hope that this works properly. Then I'll include a respective note in the next DERIVE Newsletter.

#1:
$$\text{kron_prod}(a, b) := \text{APPEND}\left(\text{VECTOR}\left(\text{APPEND_COLUMNS}\left(\left(\text{VECTOR}\left(\text{VECTOR}(a_{i,k} \cdot b, k, \text{DIM}(a')), i, \text{DIM}(a))\right), j, \text{DIM}(a)\right)\right), j, \text{DIM}(a)\right)\right)$$

#2:
$$\left[\text{kron_prod}\left(\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}, \begin{bmatrix} 5 & 6 & 7 \\ 8 & 9 & 10 \\ 11 & 12 & 13 \\ 1 & 2 & 3 \end{bmatrix}\right), \text{kron_prod}\left(\begin{bmatrix} 10 & 20 & 30 \end{bmatrix}, \begin{bmatrix} 3 \\ 4 \\ 5 \end{bmatrix}\right) \right]$$

#3:
$$\left[\begin{bmatrix} 5 & 6 & 7 & 10 & 12 & 14 \\ 8 & 9 & 10 & 16 & 18 & 20 \\ 11 & 12 & 13 & 22 & 24 & 26 \\ 1 & 2 & 3 & 2 & 4 & 6 \\ 15 & 18 & 21 & 20 & 24 & 28 \\ 24 & 27 & 30 & 32 & 36 & 40 \\ 33 & 36 & 39 & 44 & 48 & 52 \\ 3 & 6 & 9 & 4 & 8 & 12 \\ 25 & 30 & 35 & 30 & 36 & 42 \\ 40 & 45 & 50 & 48 & 54 & 60 \\ 55 & 60 & 65 & 66 & 72 & 78 \\ 5 & 10 & 15 & 6 & 12 & 18 \end{bmatrix}, \begin{bmatrix} 30 & 60 & 90 \\ 40 & 80 & 120 \\ 50 & 100 & 150 \end{bmatrix} \right]$$

Tom Barbara, Oregon, USA

Hi Josef,

Well your routine looks correct. Studying your example will be a good lesson in programming for me.

Thanks so much.

Fred J. Tydeman

I notice that in a function definition, one can have multiple statements on one line, with a comma as the separator between statements.

However, outside of a function, when I try either a comma or a semi-colon to separate multiple statements on one line, they are considered errors.

So, how does one do multiple statements on one line?

Danny Ross Lunsford

Use a PROG statement to group them, e.g.

```
PROG(x := 10, y := SIN(x), DISPLAY ("Hello world"))
```

This evaluates to whatever is returned by the last statement in the group.

-drl

Fred J. Tydeman

I created a file a.mth with the contents of:

```
PROG (x:=3, y:=4)
[a:=1, b:=5]
x+a
```

I did Simplify, Basic the x+a line and got

```
x + 1
```

Johann Wiesenbauer

Hi Fred,

Maybe I'm wrong, but it appears to me that you didn't understand the crucial difference between `[x:=3, y:=4]` and `prog(x:=3, y:=4)`. In the first case, x and y will have the corresponding values after INPUTTING that line, in the second case x and y will have those values only after SIMPLIFYING that line. After all, `prog(x:=3, y:=4)` is a program, as its name says, and programs will never be started automatically in input mode. Hope this helps,

Johann

Germain Labonté

[grmlabonte@yahoo.ca]

Hello,

I've come accross your contact information from the Derive User Group web site.

I've had some interest in muLISP (on which Derive was programmed). From what I can gather, the latest version of muLISP which was sold up to the first half of 2005 (along with Derive for DOS) included both the 16-bit and 32-bit kernels.

Is this correct? If so, what would be the version number. I'm hoping that one day a copy of this version of muLISP will come on the "used-software" market (e.g. eBay) and I would like to know what questions to ask the seller.

Sincere regards,

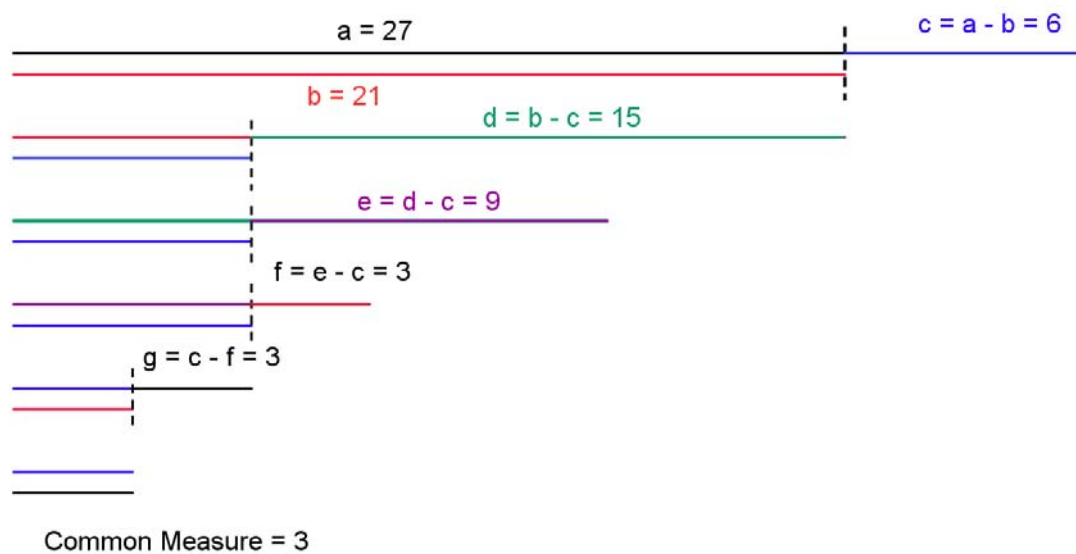
Germain Labonté, Mississauga, Ontario, Canada

Is there anybody, who can help?

The Common Measure

Roland Schröder, Celle, Germany

Two wooden sticks of lengths $a = 27$ cm and $b = 21$ cm shall be sawed up – without using any measuring instrument – to as many pieces of equal length as possible. We can do this by putting the sticks flush together and separate the overhang $c = a - b$ from the longer stick. Now we choose the two shortest remaining sticks and repeat the procedure. This describes a recursive algorithm which ends if we cannot choose the two shorter sticks because they all are of the same length (which is 3 in our case).



We see that 3 is the GCD (Greatest Common Divisor) of 27 and 21 and that this recursive procedure is nothing else than the Euclidean Algorithm. I'd recommend giving the pupils pairs of straws or paper stripes and letting them perform this activity. They should find out that this is a way to find the GCD of two numbers.

Maybe that some of you prefer using the computer instead of a saw, then you can write in DERIVE:

$$\text{ITERATES}([b, |a - b|], [a, b], [27, 21], 7) = \begin{bmatrix} 27 & 21 \\ 21 & 6 \\ 6 & 15 \\ 15 & 9 \\ 9 & 6 \\ 6 & 3 \\ 3 & 3 \\ 3 & 0 \end{bmatrix}$$

Calculation ends when comparing the two shortest sticks the overhang $o = 0$. The common measure is the minimum positive number appearing in the procedure.

Of course, we don't know that only 7 iteration steps are necessary. What happens if we do not enter the number of steps?

$$\text{ITERATES}([b, |a - b|], [a, b], [27, 21]) = \begin{bmatrix} 27 & 21 \\ 21 & 6 \\ 6 & 15 \\ 15 & 9 \\ 9 & 6 \\ 6 & 3 \\ 3 & 3 \\ 3 & 0 \\ 0 & 3 \\ 3 & 3 \end{bmatrix}$$

We see that ITERATES works until discovering the second occurrence of an element – which is [3,3] in our case. If our goal is only finding the common measure applying our function it will be sufficient to deliver the first element of the last row:

$$\underset{1}{(\text{FIRST}(\text{REVERSE}(\text{ITERATES}([b, |a - b|], [a, b], [27, 21]))))} = 3$$

$$\underset{1}{(\text{FIRST}(\text{REVERSE}(\text{ITERATES}([b, |a - b|], [a, b], [104, 74]))))} = 2$$

Our story does not end here. It is mathematically interesting to take sticks of irrational lengths. What about side and diagonal of the unit square ($a = \sqrt{2}$, $b = 1$).

$$\text{ITERATES}([b, |a - b|], [a, b], [\sqrt{2}, 1], 8) = \begin{bmatrix} \sqrt{2} & 1 \\ 1 & \sqrt{2} - 1 \\ \sqrt{2} - 1 & 2 - \sqrt{2} \\ 2 - \sqrt{2} & 3 - 2\sqrt{2} \\ 3 - 2\sqrt{2} & \sqrt{2} - 1 \\ \sqrt{2} - 1 & 3\sqrt{2} - 4 \\ 3\sqrt{2} - 4 & 3 - 2\sqrt{2} \\ 3 - 2\sqrt{2} & 5\sqrt{2} - 7 \\ 5\sqrt{2} - 7 & 10 - 7\sqrt{2} \end{bmatrix}$$

It is necessary to enter the number of iterations, otherwise the calculations does not stop (in exact mode). Working in approximate mode, we get after 50 iteration steps:

$$\left[1.501823965 \cdot 10^{-6}, 1.501823965 \cdot 10^{-6} \right]$$

Some expressions are occurring twice in the same column. We can conclude that they are cut off twice and we define – a little bit daring – the following recursion instruction.

$$a_{n+1} = a_{n-1} - 2 \cdot a_n; a_1 = 1; a_2 = \sqrt{2} - 1$$

We transfer this recursion to DERIVE:

$$\text{ITERATES}([b, a - 2 \cdot b], [a, b], [1, \sqrt{2} - 1], 8)$$

and we receive

$$\begin{bmatrix} 1 & \sqrt{2} - 1 \\ \sqrt{2} - 1 & 3 - 2 \cdot \sqrt{2} \\ 3 - 2 \cdot \sqrt{2} & 5 \cdot \sqrt{2} - 7 \\ 5 \cdot \sqrt{2} - 7 & 17 - 12 \cdot \sqrt{2} \\ 17 - 12 \cdot \sqrt{2} & 29 \cdot \sqrt{2} - 41 \\ 29 \cdot \sqrt{2} - 41 & 99 - 70 \cdot \sqrt{2} \\ 99 - 70 \cdot \sqrt{2} & 169 \cdot \sqrt{2} - 239 \\ 169 \cdot \sqrt{2} - 239 & 577 - 408 \cdot \sqrt{2} \\ 577 - 408 \cdot \sqrt{2} & 985 \cdot \sqrt{2} - 1393 \end{bmatrix}$$

No expression is appearing twice. Approximating the matrix shows that all expressions are positive. This confirms – at least for the first 9 steps – that each expression can be subtracted twice from its predecessor.

We would like to find the explicit representation of this sequence. For this purpose we use a method which can be taken as the standard one. We conjecture that it might be a geometric sequence $(a \cdot q^n)_{n \in \mathbb{N}}$. So we substitute for the respective elements of the sequence in the recursion formula:

$$a \cdot q^{n+1} = a \cdot q^{n-1} - 2a \cdot q^n.$$

After division by $a \cdot q^{n-1}$ we get $q^2 = 1 - 2q$ which is a quadratic equation with solutions $q_{1/2} = \pm \sqrt{2} - 1$. As all elements of the sequence are positive – and they shall remain positive – we accept only the positive solution. Using the positive quotient we can define a geometric sequence:

$$\text{VECTOR}\left(\left[n, (\sqrt{2} - 1)^n\right], n, 0, 8\right) = \begin{bmatrix} 0 & 1 \\ 1 & \sqrt{2} - 1 \\ 2 & 3 - 2 \cdot \sqrt{2} \\ 3 & 5 \cdot \sqrt{2} - 7 \\ 4 & 17 - 12 \cdot \sqrt{2} \\ 5 & 29 \cdot \sqrt{2} - 41 \\ 6 & 99 - 70 \cdot \sqrt{2} \\ 7 & 169 \cdot \sqrt{2} - 239 \\ 8 & 577 - 408 \cdot \sqrt{2} \end{bmatrix}$$

The right column of this table corresponds with the left column of the result of the recursion from above.

So it seems to be clear that alternative removal starting with two sticks of lengths 1 and $\sqrt{2}$ is leading to a geometric sequence of stick lengths. Because of $q < 1$ it will converge towards zero but never reach zero. The procedure for finding a common measure will never come to an end. There good reasons to suspect, that 1 and $\sqrt{2}$ don't have a common measure. We can name 1 and $\sqrt{2}$ as incommensurable.

The procedure can be used to create approximating representations as fractions together with a sound error estimation. DERIVE gives:

$$(\sqrt{2} - 1)^{20} \approx 0,0000000221$$

or:

$$22619537 - 15994428 \cdot \sqrt{2} \approx 0,0000000221.$$

Solving for $\sqrt{2}$ results in the difference of a fraction (approximation for $\sqrt{2}$ represented as a fraction) and a number less 10^{-14} (error estimation).

The following screen shots show the procedure performed on the Voyage 200 (in Sequence Mode):

F1	F2	F3	F4	F5	F6	F7
Zoom	Edit	✓	All	Style	Axes...	
PLOTS ✓ u1=u2(n-1) u11=27 ✓ u2= u1(n-1)-u2(n-1) u12=21 u3= u13= u4= u14= u5= u15= u3(n)=						
MAIN RAD EXACT SEQ						

F1	F2	F3	F4	F5	F6	F7
Setup	Cell	Header	Del	Row	Im	Row
n	u1	u2				
0.	27.	21.				
1.	21.	6.				
2.	6.	15.				
3.	15.	9.				
4.	9.	6.				
5.	6.	3.				
6.	3.	3.				
7.	3.	0.				
n=0.						
MAIN RAD EXACT SEQ						

F1	F2	F3	F4	F5	F6	F7
Zoom	Edit	✓	All	Style	Axes...	
PLOTS u1=u2(n-1) u11=27 u2= u1(n-1)-u2(n-1) u12=21 ✓ u3=u4(n-1) u13=1 ✓ u4=u3(n-1)-2·u4(n-1) u14= 2-1 u5= u5(n)=						
MAIN RAD EXACT SEQ						

F1	F2	F3	F4	F5	F6	F7
Setup	Cell	Header	Del	Row	Im	Row
n	u3	u4				
0.	1.	.414214				
1.	.414214	.171573				
2.	.171573	.071068				
3.	.071068	.029437				
4.	.029437	.012193				
5.	.012193	.005051				
6.	.005051	.002092				
7.	.002092	8.67E-4				
n=0.						
MAIN RAD EXACT SEQ						

Both examples from above performed with CellSheet Application of the TI handheld:

F1	F2	F3	F4	F5	F6	F7	F8
File	Plot	Edit	Links	\$	Funcs	Stat	ReCalc
mea	A	B	C		D		E
1	27	21		1	$f(2)-1$		
2	21	6		$f(2)-1$	$3-2*f(2)$		
3	6	15		$3-2*f(2)$	$5*f(2)-7$		
4	15	9		$5*f(2)-7$	$17-12*f(2)$		
5	9	6		$17-12*f(2)$	$29*f(2)-41$		
6	6	3		$29*f(2)-41$	$99-70*f(2)$		
7	3	3		$99-70*f(2)$	$169*f(2)-239$		
D7: =C6-2*D6							
MAIN RAD EXACT SEQ							

(Josef)

Tutorials for the NSpireCAS – Distributions

Guido Herweyers, T3-Vlaanderen & T3-Wallonie, Belgium

1. Sampling a normal distribution

Press to turn on the handheld.
Press for the home menu.
Press to open a new Lists and Spreadsheet page.



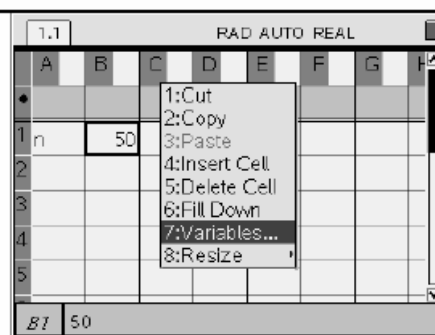
Consider a population that has a normal distribution with mean $\mu = 175$ and standard deviation $\sigma = 10$.

We want to take a sample of this population with sample size $n = 50$

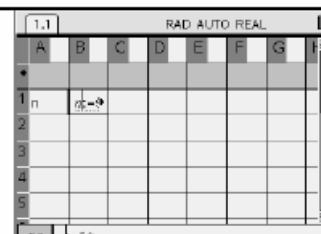
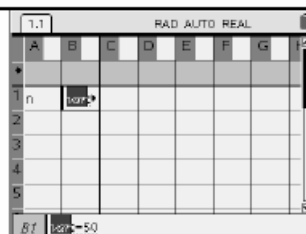
Enter n in cell A1 and 50 in B1.

Press for the context menu.

Choose 7: Variables
1: Store Var

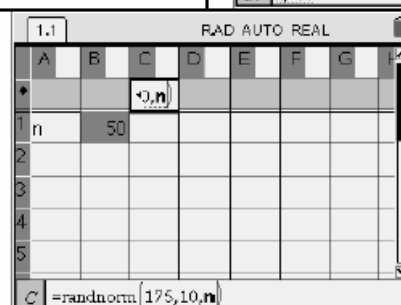


Type n in the place of var.
From now on, the value of this cell is linked to the variable n .



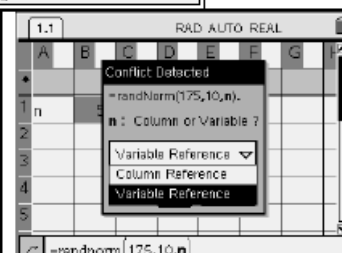
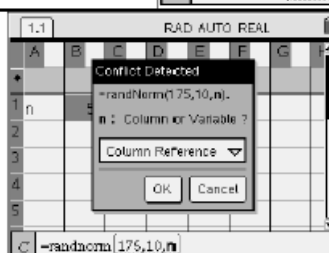
Go to the gray position of column C (second line) to define the column and type:
 $= \text{randnorm}(175, 10, n)$

This will generate the sample in column C

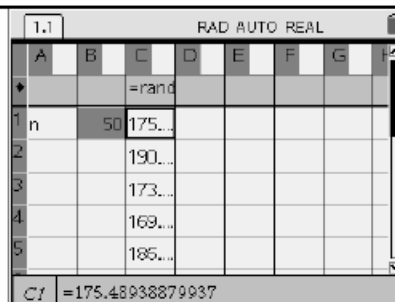


First you are asked to choose whether n is the name of a column or a variable.

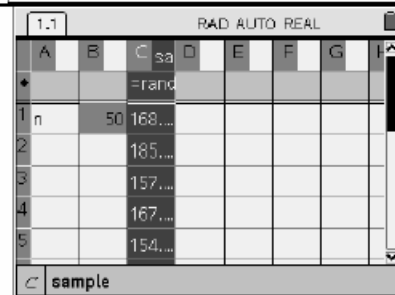
Press and to choose variable reference, confirm with



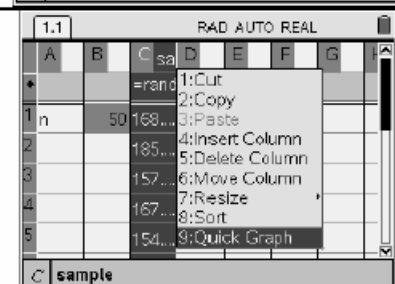
The sample appears in column C



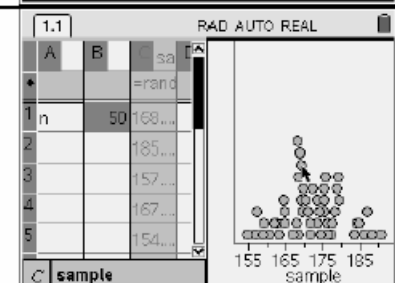
Move to the top position to enter a name for column C, type sample
Then press to select the whole column



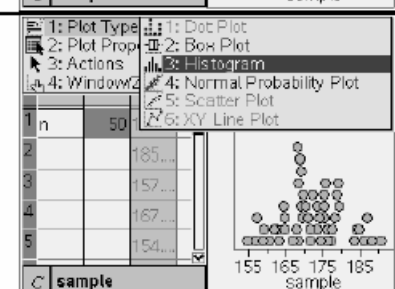
Press for the context menu.
Choose 9: Quick Graph



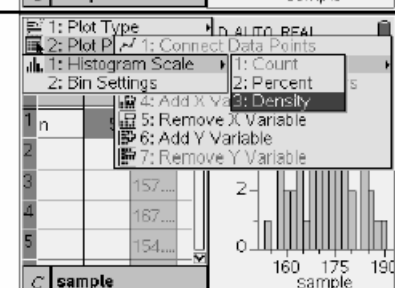
The window is split into two parts, by default a dot plot of the data appears in the data and statistics window.





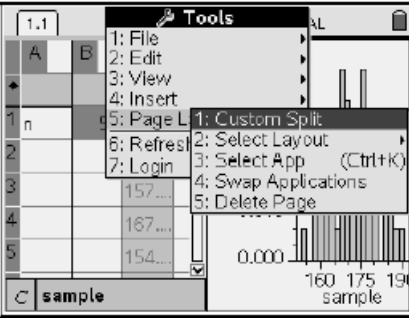


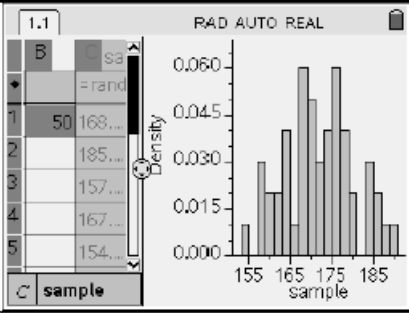

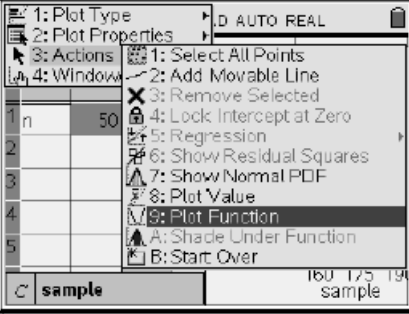

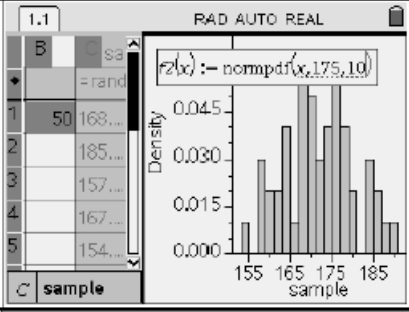


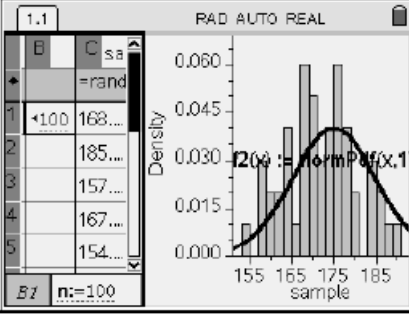
Press
then 1: Plot type
3: Histogram



Press
then 2: Plot Properties
2: Histogram properties
1: Histogram scale
3: Density

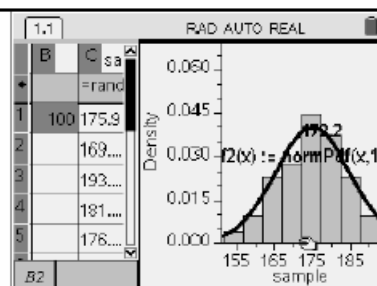


We now have a histogram with density scale (total area = 1)

<p>Press   for the tools menu, then</p> <p>5: Page 1: Custom Split</p>	
<p>Press  for a wider histogram window and confirm with </p>	
<p>Press , then</p> <p>3: Actions 9: Plot Function</p>	
<p>The syntax for the normal density function with mean $\mu = 175$ and standard deviation $\sigma = 10$ is <code>normpdf(x, 175, 10)</code></p> <p>Confirm with </p>	
<p>Select the spreadsheet window with  </p> <p>Change the sample size into 100.</p>	

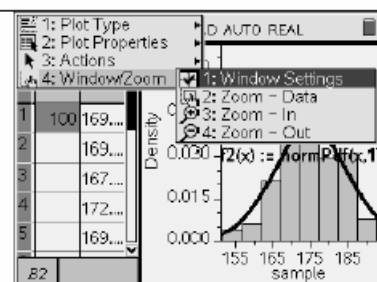
The histogram width is too small.
Go to the x-axis until \leftrightarrow appears,
press ctrl \leftarrow to get a closed hand \leftarrow .
Press \rightarrow or \leftarrow to change the histogram
width.

Observe the influence of the histogram
width.



To change the window settings press
 menu , then
4: Window/Zoom
1: Window settings

choose Xmin= 145 and Xmax= 205



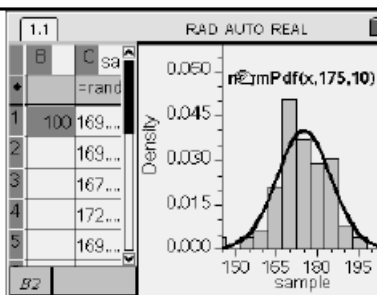
Displace the function label.

Select the spreadsheet window with

ctrl tab

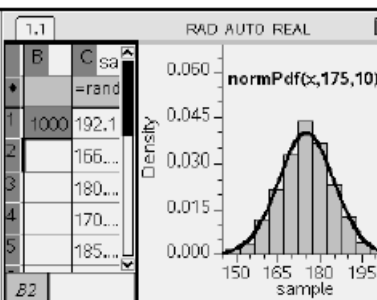
To take a new sample press ctrl R

Take a few samples, the changing
histograms illustrate the variability of the
data.



Change the sample size into 200, ...,
1000.

The histograms are getting closer and
closer to the density function.



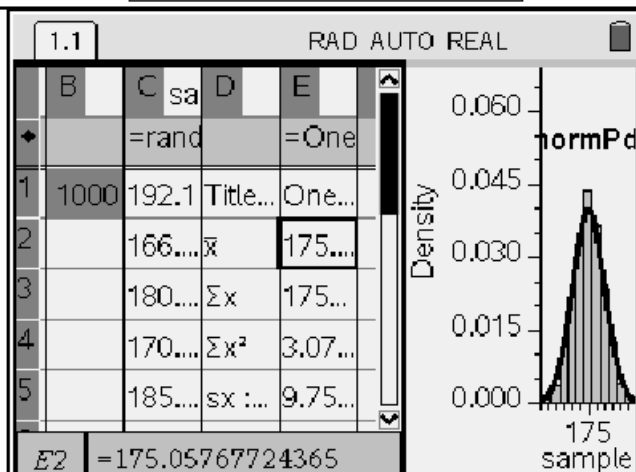
Task:

find out how you can produce the one
variable statistics of the sample data in
the spreadsheet window.

The sample mean \bar{x} and the sample
standard deviation s are estimations for
 $\mu = 175$ and $\sigma = 10$.

Observe:

1. The variability from sample to sample
with the same sample size.
2. The estimations are getting better
with increasing sample size.




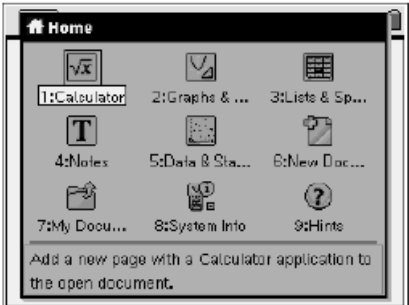

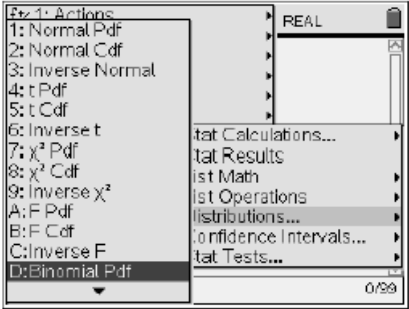
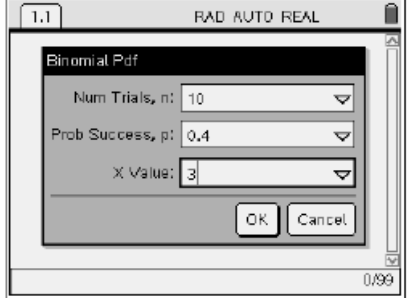
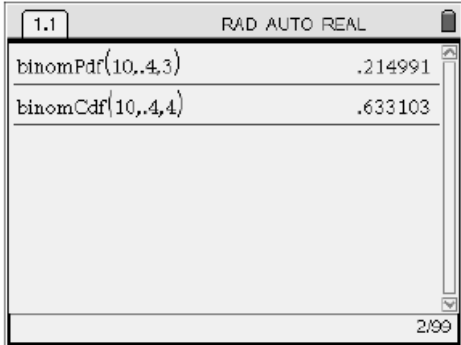


2 Binomial distributions

Forty per cent of a sweet assortment are 'soft centres' and the remainder 'hard centres'.

A handful of ten sweets may be regarded as a random sample from a huge pile of the sweets.

- What is the probability that there are three of ten 'soft centres' in a handful?
- What is the probability that there are at most four of ten 'soft centres' in a handful?




<p>Press  to turn on the handheld.</p> <p>Press  for the home menu.</p> <p>Press  to open a calculator page.</p>	
<p>Press , then</p> <p>7: Statistics</p> <p>5: Distributions</p> <p>D: Binomial Pdf</p>	
<p>The number of trials is $n=10$</p> <p>The success probability is $p=0.4$</p> <p>The X-value is 3</p>	
<p>If a random variable X has a binomial distribution with parameters n and p, then</p> <p>$P(X=x) = \text{binompdf}(n,p,x)$</p> <p>a) $P(X=3) = 21.5 \%$</p> <p>$P(X \leq x) = \text{binomcdf}(n,p,x)$ (the cumulative distribution function)</p> <p>b) $P(X \leq 4) = 63.3\%$</p>	

3 Poisson distributions


If a large grass lawn contains on average 1 weed per 600 cm², what will be the distribution of the total number X of weeds in an area of 400 cm² ?

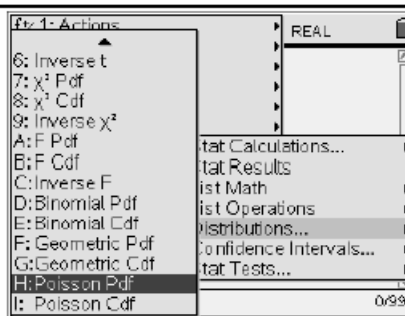
- a) Find $P(X=2)$
- b) Find $P(X \leq 3)$

A suitable model for this situation is a Poisson distribution whose mean is $\lambda = \frac{400}{600} = \frac{2}{3}$

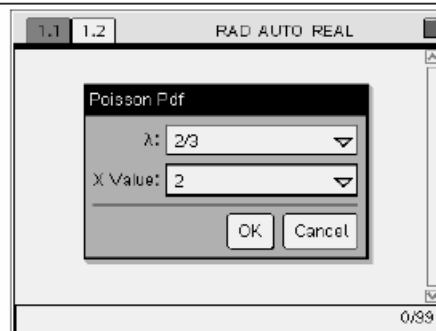
Press  to turn on the handheld.
Press  for the home menu.
Press  to open a calculator page.



Press , then
7: Statistics
5: Distributions
H: Poisson Pdf



$$\lambda = 2/3 \quad \text{and} \quad X = 2$$



If a random variable X has a Poisson distribution with mean λ , then

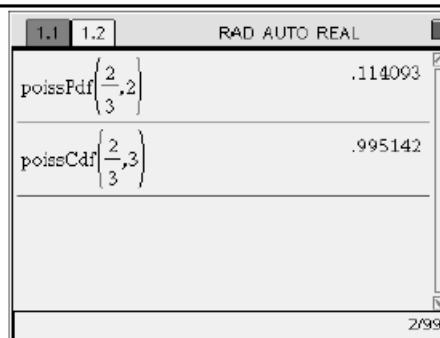
$$P(X=x) = \text{poisspdf}(\lambda, x)$$

$$\text{a) } P(X=2) = 11.4 \%$$

$$P(X \leq x) = \text{poisscdf}(\lambda, x)$$

(the cumulative distribution function)

$$\text{b) } P(X \leq 3) = 99.5\%$$



What's the Time, Grandie?

Josef Böhm, Würmla, Austria

Maybe that some of you will remember DNL#45 where I introduced proudly our first granddaughter Kim. Later I presented a picture with Kim and the TI-89 (with Grandpa's eyeglasses, of course). Now Kim is attending the grammar school in Tulln, Lower Austria, and sometimes – it is not too often – she asks her grandpa for support in mathematics.

Below is Kim together with her sister Yvonne sitting on my desk some years ago.

At the right you can see Kim and Yvonne (both left) together with their younger sisters and her mother Astrid (our daughter).



It was last year when Kim needed some exercising for calculations with times. I took the occasion to extent my collection of training programs and wrote a DERIVE utility for adding and subtracting times and for adding times to certain day times in the week.

```

start=

#1:  LOAD(D:\DOKUS\SCHULE\skills\time_US.mth)

Trainer for calculating times

ti(n)= gives n problems to add and subtract times and
time_ex_res= shows the correct answers.

time= gives a day of the week together with the time of the day
and a time to add or to subtract.
res= gives the correct answer.

time_ex(n)= offers n examples in form of a table and
time_ex_res= shows the correct answers.

```

All my training programs show the same structure (except one which treats exercising with Venn-diagrams for visualising set operations):

The files are MTH-files and should be loaded as Utility-files. Then simplify the command `start`. You will be presented the instructions how to use the file (see page 15). The “trick” is to use the `DISPLAY` command for this purpose (see the code of `start_()` on the next page). `start_()` contains the simplification of `dummy:=random(0)` which makes sure that we will be offered new problems at every run of the utility.

A session could – according to the instructions given above – start and run as follows:

```
#2: start =
```

```
#3: ti(3) = [ 5 hrs 53 min 18 sec + 13 hrs 37 min 18 sec
             17 hrs 26 min 11 sec - 4 hrs 35 min 20 sec
             20 hrs 9 min 39 sec + 8 hrs 17 min 32 sec ]
```

```
#4: time_ex_res = [ 19 hrs 30 min 36 sec
                   12 hrs 50 min 51 sec
                   28 hrs 27 min 11 sec ]
```

```
#5: time = Fri 4 hrs 30 min am - 7 hrs 7 min
```

```
#6: res = Thu 9 hrs 23 min pm
```

```
#7: time = Thu 4 hrs 46 min pm + 17 hrs 38 min
```

```
#8: res = Fri 10 hrs 24 min am
```

```
#9: time_ex(5) = [ Thu 12 hrs 23 min am + 16 hrs 0 min
                  Thu 8 hrs 35 min pm - 5 hrs 44 min
                  Mon 8 hrs 53 min am - 8 hrs 55 min
                  Wed 3 hrs 49 min am + 15 hrs 12 min
                  Mon 3 hrs 55 min am + 6 hrs 24 min ]
```

```
#10: time_ex_res = [ Fri 4 hrs 23 min am
                   Thu 2 hrs 51 min pm
                   Sun 11 hrs 58 min pm
                   Wed 7 hrs 1 min pm
                   Mon 10 hrs 19 min am ]
```

Kim learns English – and she likes English more than mathematics – so I could not resist to include the “am – pm” notation of the times of the day. Additionally it was nice to program this feature.

The next page shows the utility file:

```

#1:  [days := [Mon, Tue, Wed, Thu, Fri, Sat, Sun]]

#2:  [time_res :=, t_res :=, time_ex_res :=]

#3:  [t1 :=, h1 :=, m1 :=, h2 :=, m2 :=, op :=]

      start_(dummy) :=
        Prog
          dummy := RANDOM(0)
          DISPLAY("Trainer for calculating times")
          DISPLAY("")
          DISPLAY("ti(n)= gives n problems to add and subtract times and")
          DISPLAY("time_ex_res= shows the correct answers.")
#4:  DISPLAY("")
          DISPLAY("time= gives a day of the week together with the time of the day")
          DISPLAY("and a time to add or to subtract.")
          DISPLAY("res= gives the correct answer.")
          DISPLAY("")
          DISPLAY("time_ex(n)= offers n examples in form of a table and")
          DISPLAY("time_ex_res= shows the correct answers.")
          DISPLAY("")

#5:  start := start_()

      zeit_(days, time1, time2, h1_) :=
        Prog
          t1 := RANDOM(7) + 1
          h1 := RANDOM(24)
          h1_ := IF(h1 > 12, h1 - 12, h1)
          sp := IF(h1 > 12, "pm", "am")
#6:  m1 := RANDOM(60)
          time1 := APPEND(days↓t1, " ", STRING(h1_), " hrs ", STRING(m1), " min ", sp)
          h2 := RANDOM(24)
          m2 := RANDOM(60)
          time2 := APPEND(STRING(h2), " hrs ", STRING(m2), " min")
          op := [" + ", " - "]↓(RANDOM(2) + 1)
          APPEND(time1, op, time2)

#7:  time := zeit_()

      res_(st, d, erg, mr, hr, tr, sp) :=
        Prog
          st := t1*24*60 + h1*60 + m1
          d := h2*60 + m2
          erg := IF(op = " + ", st + d, st - d)
          mr := MOD(erg, 60)
#8:  erg := (erg - mr)/60
          hr := MOD(erg, 24)
          erg := MOD((erg - hr)/24, 7)
          sp := IF(hr > 12, "pm", "am")
          erg := IF(erg = 0, 7, erg)
          hr := IF(hr > 12, hr - 12, hr)
          APPEND(days↓erg, " ", STRING(hr), " hrs ", STRING(mr), " min ", sp)

#9:  res := res_()

      time_ex(n, n_, table) :=
        Prog
          n_ := 1
          table := []
          time_ex_res := []
#10:  Loop
          If n_ > n exit
          table := APPEND(table, [[time]])
          time_ex_res := APPEND(time_ex_res, [[res]])
          n_ := + 1
          RETURN table

```

The next expression is expression #11. I removed the expression number to have a better screen shot of the extended function.

```
time_(secs, min, hrs, tim1, tim2, tim3, signs, si, tsk, pr) :=
  Prog
    Loop
      pr := ""
      signs := VECTOR(1 - 2*RANDOM(2), k, RANDOM(2) + 1)
      tsk := VECTOR(RANDOM(60*60*24) + 1, k, DIM(signs) + 1)
      signs := APPEND([1], signs)
      secs := signs*tsk
      t_res := secs
      tim1 := MOD(secs, 60)
      t_res := (t_res - tim1)/60
      tim2 := MOD(t_res, 60)
      tim3 := (t_res - tim2)/60
      t_res := APPEND(STRING(tim3), " hrs ", STRING(tim2), " min ", STRING(tim1), " sec")
      If secs > 0
        Loop
          If DIM(tsk) = 0
            RETURN REST(REST(pr))
          secs := MOD(tsk↓1, 60)
          tsk↓1 := (tsk↓1 - secs)/60
          min := MOD(tsk↓1, 60)
          hrs := (tsk↓1 - min)/60
          si := IF(signs↓1 = 1, " + ", " - ")
          pr := APPEND(pr, si, STRING(hrs), " hrs ", STRING(min), " min ", STRING(secs), " sec ")
          tsk := REST(tsk)
          signs := REST(signs)

ti(n, n_, table) :=
  Prog
    n_ := 1
    table := []
    time_ex_res := []
#12:   Loop
      If n_ > n exit
      table := APPEND(table, [[time_()]])
      time_ex_res := APPEND(time_ex_res, [[t_res]])
      n_ := + 1
    RETURN table
```

Among the files accompanying this DNL you can find `zeit.mth`, which provides a German version without the “am-pm”-notation. It should be no problem for non-German users to adapt the file for their own language if necessary. (`tage := [Mo, Di, Mi, Do, Fr, Sa, So]` must be changed!)

```
#12:  zeitb(5) = [
    Do 8h 0min + 3h 0min
    So 2h 2min - 12h 59min
    Do 9h 5min + 15h 18min
    Fr 20h 43min + 12h 13min
    Do 10h 29min + 11h 2min
  ]

#13:  zeitb_res = [
    Do 11h 0min
    Sa 13h 3min
    Fr 0h 23min
    Sa 8h 56min
    Do 21h 31min
  ]
```

An Interesting Property of a Triangle

Peter Lüke-Rosendahl, Germany

In 1968 the following theorem was presented as a problem in the journal *American Mathematical Monthly*:

Given is a triangle ABC . We raise squares $CBED$, $ACFG$ and $BAHK$ over the sides of the triangle outwards. Then we draw the parallelograms $FCDQ$ and $EBKP$. Triangle PAQ seems to be right and isosceles as well. Prove this!

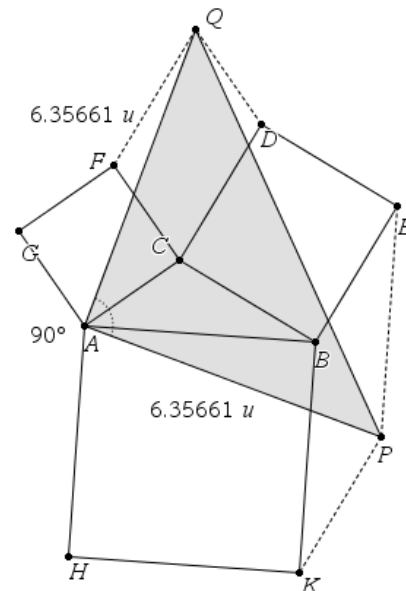
(It might be nice to pose the problem to the students changing the last sentence: What can you say about triangle PAQ ? Prove your conjecture? Josef)

The first step (finding a conjecture) is a typical task for working with a dynamic geometry program.

We can do it on the PC using TI-NspireCAS or Cabri or GeoGebra or any other DGS.

But we can do it on the handheld, too using either the TI-92 / Voyage 200 with their Cabri-Application or the TI-Nspire handheld device.

The next step could be verifying the conjecture using special data for three points and verifying the fact that the other two possibilities drawing the triangle will give the same result.



TI-Nspire – the Graphs & Geometry Application

I will proceed with DERIVE using the slider bars keeping the coordinates of the triangle vertices general and then continue proving the two properties.

```
#1: CaseMode := Sensitive
```

```
#2: InputMode := Word
```

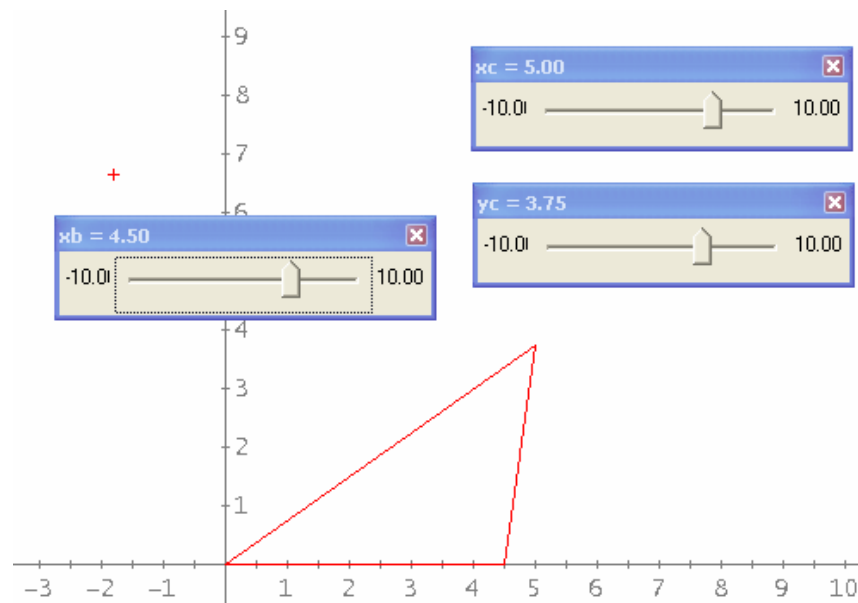
```
#3: triangle(U, V, W) := [U, V, W, U]
```

```
#4: [A := [0, 0], B := [xb, 0], C := [xc, yc]]
```

Introduce sliders for xb, xc and yc from -10 ... +10

```
#5: triangle(A, B, C)
```

Plot the triangle



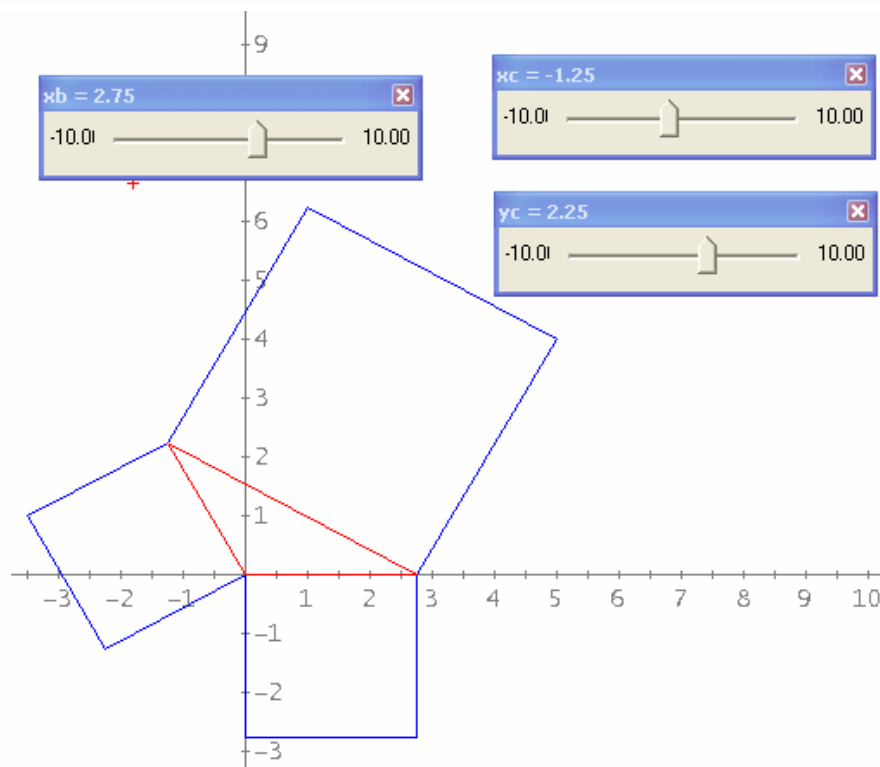
#6: $\text{perp}(U, V) := \begin{bmatrix} (V - U)_2 \\ -(V - U)_1 \end{bmatrix}$

$\text{square}(U, V)$ = square over UV outside of the triangle (neg. direction of rotation)

#7: $\text{square}(U, V) := [U, V, V + \text{perp}(V, U), U + \text{perp}(V, U), U]$

#8: $[\text{square}(A, C), \text{square}(C, B), \text{square}(B, A)]$

Plot the squares



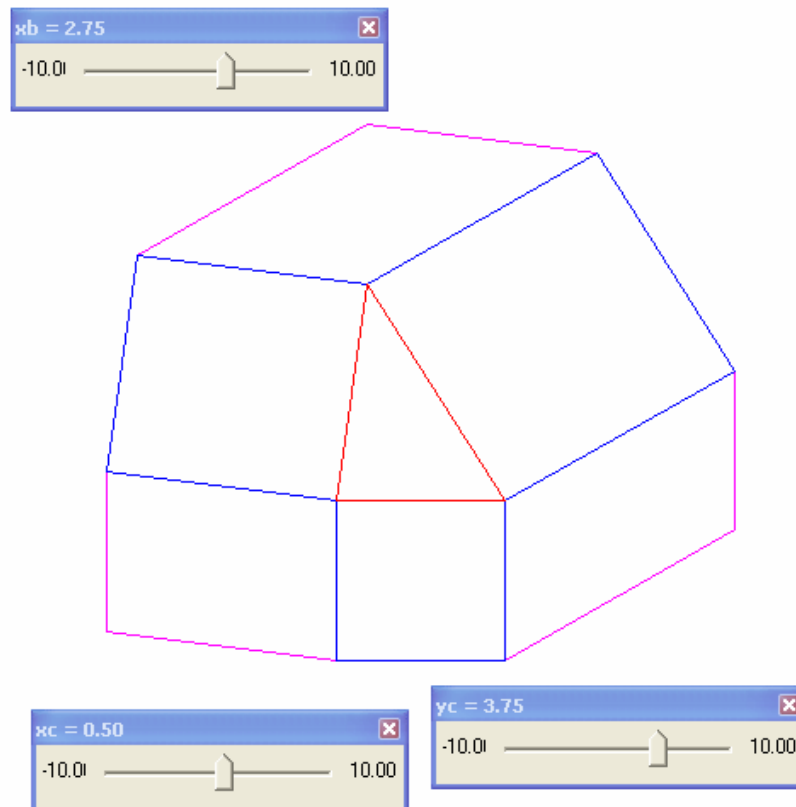
$\text{Ppt}(U,V,W)$ = point of parallelogram with point of triangle W as vertex

#9: $\text{Ppt}(U, V, W) := W + \text{perp}(W, U) + \text{perp}(V, W)$

$\text{par}(U,V,W)$ = parallelogram with vertex W

#10: $\text{par}(U, V, W) := [W, W + \text{perp}(V, W), \text{Ppt}(U, V, W), W + \text{perp}(W, U), W]$

Plot the parallelograms



#11: $[\text{par}(A, B, C), \text{par}(C, A, B), \text{par}(B, C, A)]$

#12: $\text{goaltr}(U, V, W) := [U, \text{Ppt}(U, V, W), \text{Ppt}(W, U, V), U]$

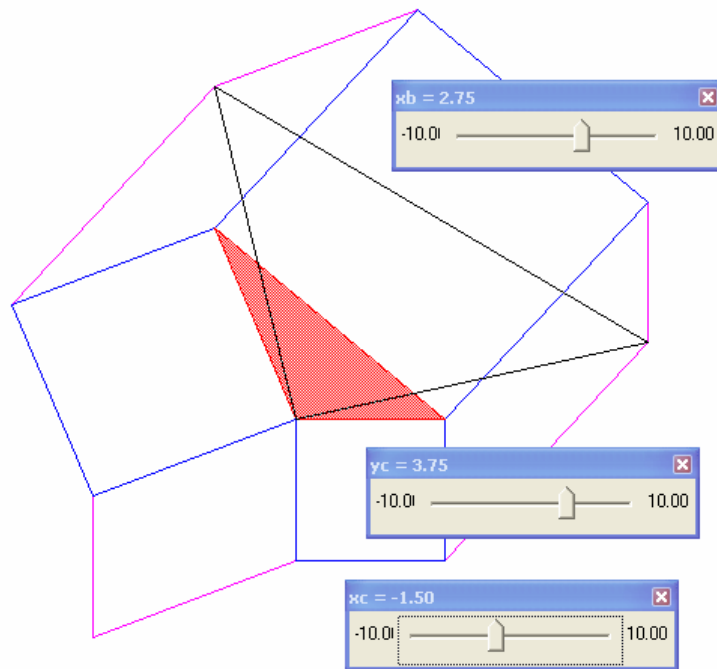
#13: $\text{goaltr}(A, B, C)$

Plot one of the final triangles

#14: $\text{POLYGON_FILL}([A, B, C, A])$

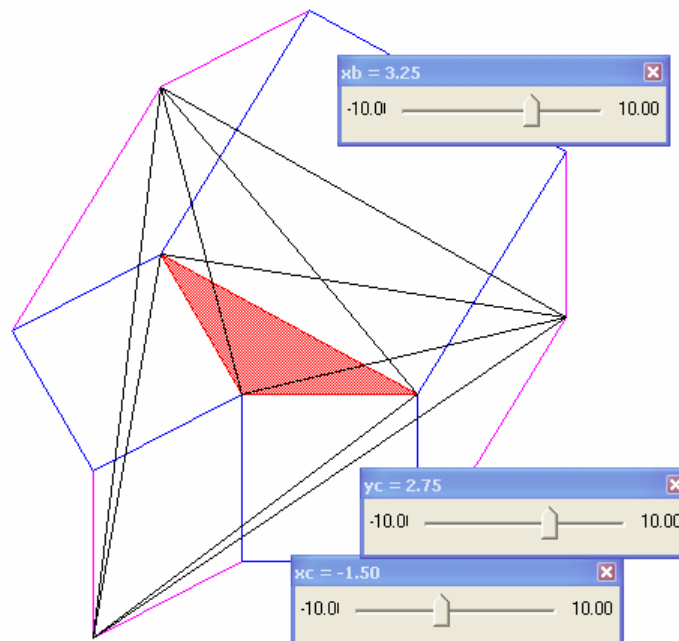
Plot to fill the initial triangle

#15: $[\text{goaltr}(B, C, A), \text{goaltr}(C, A, B)]$



#15: $[\text{goaltr}(B, C, A), \text{goaltr}(C, A, B)]$

Plot the other two final triangles



Prove the properties of the final triangles:

#16: $(\text{Ppt}(A, B, C) - A) \cdot (\text{Ppt}(C, A, B) - A) = 0$

#17: $|\text{Ppt}(A, B, C) - A| - |\text{Ppt}(C, A, B) - A| = 0$

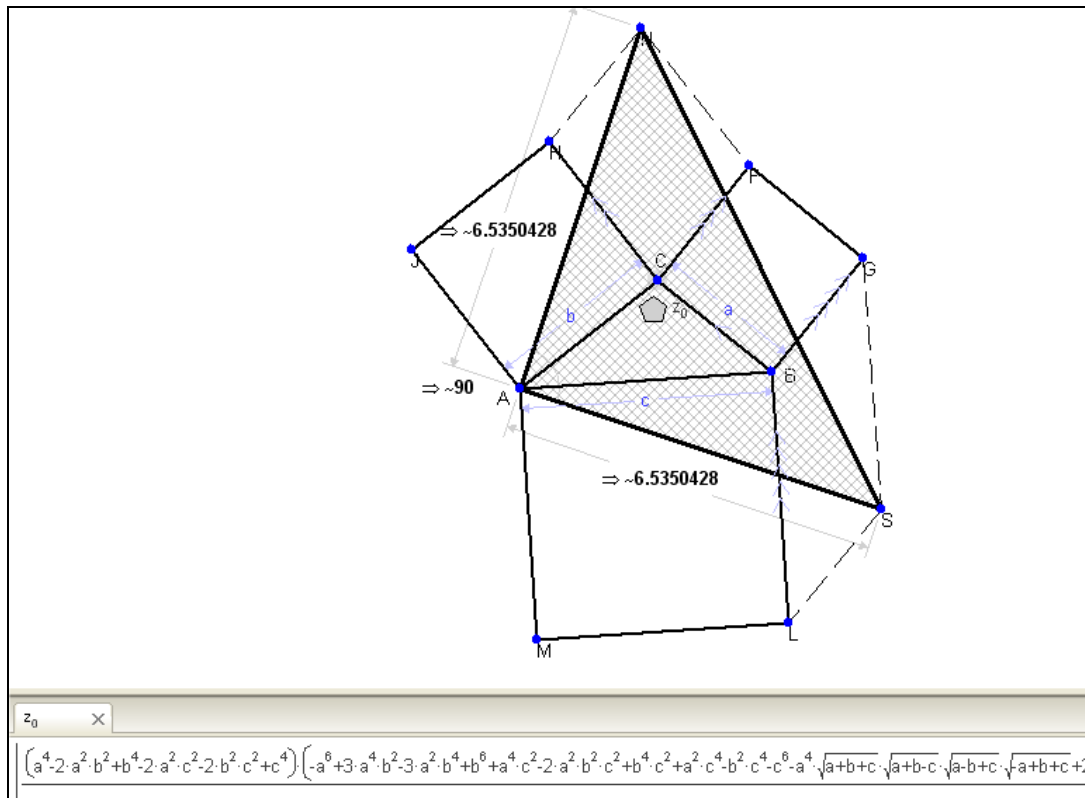
#18: $(\text{Ppt}(B, C, A) - B) \cdot (\text{Ppt}(A, B, C) - B) = 0$

#19: $|\text{Ppt}(B, C, A) - B| - |\text{Ppt}(A, B, C) - B| = 0$

#20: $(\text{Ppt}(C, A, B) - C) \cdot (\text{Ppt}(B, C, A) - C) = 0$

#21: $|\text{Ppt}(C, A, B) - C| - |\text{Ppt}(B, C, A) - C| = 0$

I asked myself if this special triangle would have a nice result for its area and reproduced the construction using GeometryExpressions. Josef



At the bottom you can see a part of the formula for the area with given sides a, b and c of the initial triangle. It is a “nice” formula, indeed. I can export this formula to DERIVE, *MATHEMATICA*, ... - and with the new GE-version to TI-NspireCAS, too.

```
Abs[
((Abs[(a)^(2) + (b)^(2) * (-1) + (c)^(2))])^(1/2) *
(Sin[
((ArcTan[(((a)^(2) + (b)^(2) * (-1) + (c)^(2))^(1/2) * ((a * (-1) + b + c))^(1/2) + ((a + (b * (-1) + c))^(1/2) *
((a + b + (c * (-1)))^(1/2) + ((a + b + c))^(1/2) * (-1)) * (-1)) +
ArcTan[(((a)^(2) + (b)^(2) * (-1) + (c)^(2))^(1/2) * ((a * (-1) + b + c))^(1/2) + ((a + (b * (-1) + c))^(1/2) *
((a + b + (c * (-1)))^(1/2) + ((a + b + c))^(1/2))])^(1/2) * (-1)) *
(((a)^(6) * (-1) + (b)^(4) * (a)^(4) * 3) + (b)^(4) * (a)^(2) * (-3) + (b)^(6) + (c)^(2) * (a)^(4)) +
|((c)^(2) * (b)^(2) * (a)^(2) * (-2)) + ((c)^(2) * (b)^(4)) + ((c)^(4) * (a)^(2)) + ((c)^(4) * (b)^(2) * (-1)) +
(c)^(6) * (-1) + (((a * (-1) + b + c))^(1/2) * ((a + (b * (-1) + c))^(1/2) * ((a + b + (c * (-1)))^(1/2) *
((a + b + c))^(1/2) * (a)^(4) * (-1)) +
(((a * (-1) + b + c))^(1/2) * ((a + (b * (-1) + c))^(1/2) * ((a + b + (c * (-1)))^(1/2) * ((a + b + c))^(1/2) * (b)^(2) * (a)^(2) * 2) +
(((a * (-1) + b + c))^(1/2) * ((a + (b * (-1) + c))^(1/2) * ((a + b + (c * (-1)))^(1/2) * ((a + b + c))^(1/2) * (b)^(4) * (-1)) +
(((a * (-1) + b + c))^(1/2) * ((a + (b * (-1) + c))^(1/2) * ((a + b + (c * (-1)))^(1/2) * ((a + b + c))^(1/2) * (c)^(4)) +
(Abs[(a)^(2) + (b)^(2) * (-1) + (c)^(2)]) * ((a * (-1) + b + c))^(1/2) * ((a + (b * (-1) + c))^(1/2) *
((a + b + (c * (-1)))^(1/2) * ((a + b + c))^(1/2) * (a)^(2)) +
(Abs[(a)^(2) + (b)^(2) * (-1) + (c)^(2)]) * ((a * (-1) + b + c))^(1/2) * ((a + (b * (-1) + c))^(1/2) *
((a + b + (c * (-1)))^(1/2) * ((a + b + c))^(1/2) * (b)^(2) * (-1)) +
(Abs[(a)^(2) + (b)^(2) * (-1) + (c)^(2)]) * ((a * (-1) + b + c))^(1/2) * ((a + (b * (-1) + c))^(1/2) *
((a + b + (c * (-1)))^(1/2) * ((a + b + c))^(1/2) * (c)^(2) * (-1)) +
(Abs[(a)^(2) + (b)^(2) * (-1) + (c)^(2)]) *
(((a)^(4) * (-1) + (b)^(2) * (a)^(2) * 2) + (b)^(4) * (-1) + (c)^(4) +
(((a * (-1) + b + c))^(1/2) * ((a + (b * (-1) + c))^(1/2) * ((a + b + (c * (-1)))^(1/2) * ((a + b + c))^(1/2) * (a)^(2)) +
(((a * (-1) + b + c))^(1/2) * ((a + (b * (-1) + c))^(1/2) * ((a + b + (c * (-1)))^(1/2) * ((a + b + c))^(1/2) * (b)^(2) * (-1)) +
(((a * (-1) + b + c))^(1/2) * ((a + (b * (-1) + c))^(1/2) * ((a + b + (c * (-1)))^(1/2) * ((a + b + c))^(1/2) * (c)^(2)) +
(Abs[(a)^(2) + (b)^(2) * (-1) + (c)^(2)]) * (a)^(2) * 2) +
(Abs[(a)^(2) + (b)^(2) * (-1) + (c)^(2)]) * ((a * (-1) + b + c))^(1/2) * ((a + (b * (-1) + c))^(1/2) *
((a + b + (c * (-1)))^(1/2) * ((a + b + c))^(1/2)))] *
((a)^(4) + (b)^(2) * (a)^(2) * (-2) + (b)^(4) + ((c)^(2) * (a)^(2) * (-2)) + ((c)^(2) * (b)^(2) * (-2)) + (c)^(4)) *
(((a)^(2) + (b)^(2) * (-1) + (c)^(2))^(1/2) * ((a * (-1) + b + c))^(1/2) * ((a + (b * (-1) + c))^(1/2) * ((a + b + (c * (-1)))^(1/2) *
((a + b + c))^(1/2) * (-1/2)) * ((a + b + c))^(1/2) * (b)^(2) * (-1/2)) *
((a + b + (c * (-1)))^(1/2) * ((a + b + c))^(1/2) * (b)^(2) * (-1/2) * (a)^(2) * (-1/2) * 1/8)]
```

The *MATHEMATICA* – output of this formula does not improve its appearance!

Peter delivered a formula for the area:

$$[A := [0, 0], B := [b, 0], C := [c, d]]$$

$$\text{Ppt}(A, B, C) = [c, b + d]$$

$$\text{Ppt}(B, C, A) = [-d, c - b]$$

$$\text{Ppt}(C, A, B) = [b + d, -c]$$

Area of the initial Triangle

$$\left| \frac{1}{2} \cdot \text{CROSS}([b, 0] - [0, 0], [c, d] - [0, 0]) \right| = \frac{|b \cdot d|}{2}$$

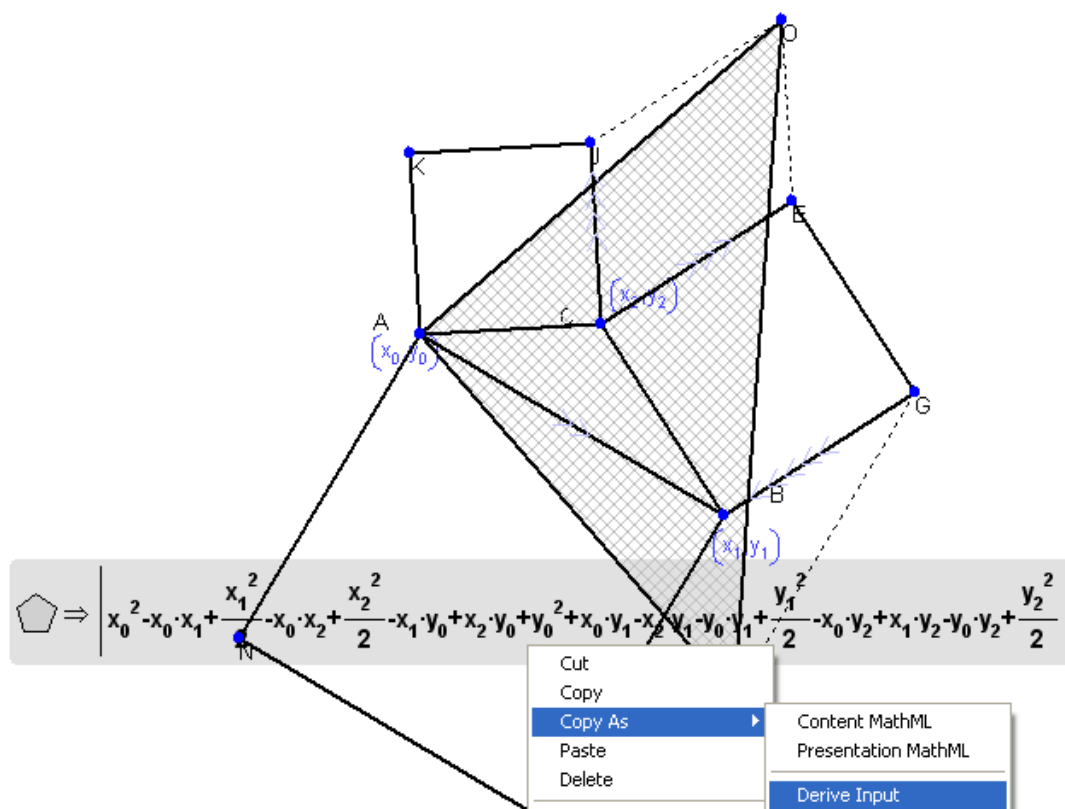
Area of the goal triangle

$$\left| \frac{1}{2} \cdot \text{CROSS}([c, b + d] - [0, 0], [b + d, -c] - [0, 0]) \right| = \frac{b^2 + 2 \cdot b \cdot d + c^2 + d^2}{2}$$

or

$$\left| \frac{1}{2} \cdot \text{CROSS}([c, b + d] - [b + d, -c], [b + d, -c] - [0, 0]) \right| = \frac{b^2 + 2 \cdot b \cdot d + c^2 + d^2}{2}$$

I repeated the construction using GeometryExpressions with generalized coordinates of the edges A, B and C and hoped for a nicer result:



Then I exported this result to DERIVE (other possibilities are to MAPLE, MATHEMATICA, TI-Nspire, Maxima, Mupad).

#1: InputMode := Word

$$\#2: \left| \begin{aligned} & x_0^2 + x_1 \cdot x_0 \cdot (-1) + \frac{x_1^2 \cdot 1}{2} + x_2 \cdot x_0 \cdot (-1) + \frac{x_2^2 \cdot 1}{2} + y_0 \cdot x_1 \cdot (-1) + y_0 \cdot x_2 + y_0^2 + y_1 \cdot x_0 + y_1 \cdot x_2 \cdot (-1) + y_1 \cdot y_0 \cdot (-1) + \\ & \frac{y_1^2 \cdot 1}{2} + y_2 \cdot x_0 \cdot (-1) + y_2 \cdot x_1 + y_2 \cdot y_0 \cdot (-1) + \frac{y_2^2 \cdot 1}{2} \end{aligned} \right|$$

$$\#3: x_0^2 - x_0 \cdot (x_1 + x_2 - y_1 + y_2) + \frac{x_1^2}{2} + x_1 \cdot (y_2 - y_0) + \frac{x_2^2}{2} + x_2 \cdot (y_0 - y_1) + y_0^2 - y_0 \cdot (y_1 + y_2) + \frac{y_1^2}{2} + \frac{y_2^2}{2}$$

$$\#4: \left| \begin{aligned} & 0^2 + b \cdot 0 \cdot (-1) + \frac{b^2 \cdot 1}{2} + c \cdot 0 \cdot (-1) + \frac{c^2 \cdot 1}{2} + 0 \cdot b \cdot (-1) + 0 \cdot c + 0^2 + 0 \cdot 0 + 0 \cdot c \cdot (-1) + 0 \cdot 0 \cdot (-1) + \frac{0^2 \cdot 1}{2} + d \cdot 0 \cdot (-1) + \\ & d \cdot b + d \cdot 0 \cdot (-1) + \frac{d^2 \cdot 1}{2} \end{aligned} \right|$$

$$\#5: \frac{b^2}{2} + b \cdot d + \frac{c^2}{2} + \frac{d^2}{2}$$

I substituted Peter's special values and could confirm Peter's results. This formula looks much better, isn't it?

Many thanks to Peter for his inspiring contribution.

By the way, I gave this problem to my students at the Technical University Vienna and surprisingly many of them tried the proof not using vector calculations but worked with trig functions (sine and cosine rule ...).

Website of GeometryExpressions:

www.geometryexpressions.com

Everyone knows that mathematics offers an excellent opportunity to learn demonstrative reasoning, but I contend also that there is no other subject in the usual curricula of the schools that affords a comparable opportunity to learn plausible reasoning. I address my self to all interested students of mathematics of all grades and I say: Certainly, let us learn proving, but also *let us learn guessing*.

G. Polya (1954). *Mathematics and Plausible Reasoning*. Princeton, NJ. Princeton University Press.

Differential Equations Made Easy

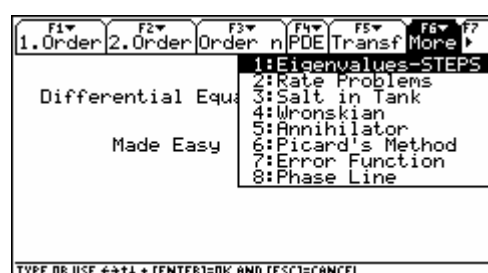
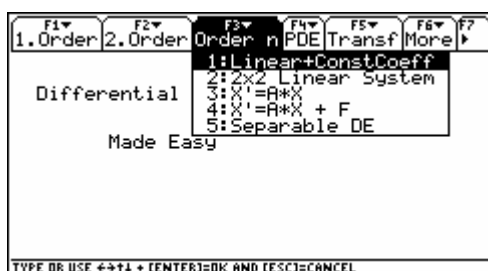
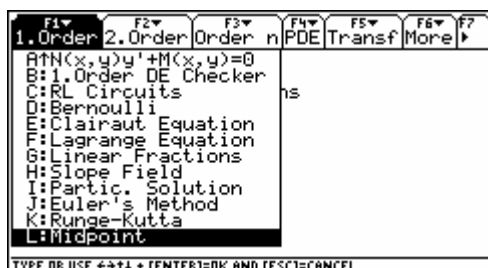
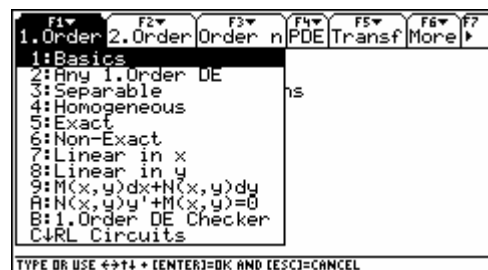
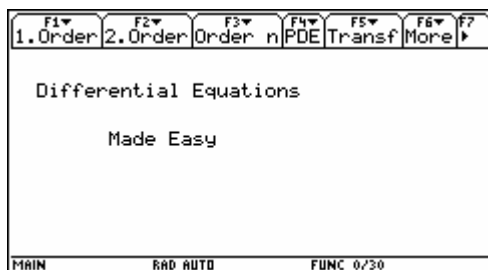
Review and Description of some features, Josef Böhm

DUG Member Nils Hahnfeld gathered a bundle of programs and functions into one application DEQME v.7.0, which can be downloaded and purchased from www.ti89.com. The application is available for TI-89, TI-92+ and Voyage 200.

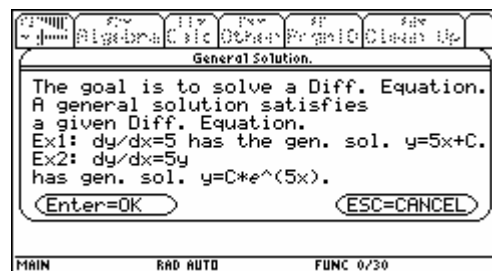
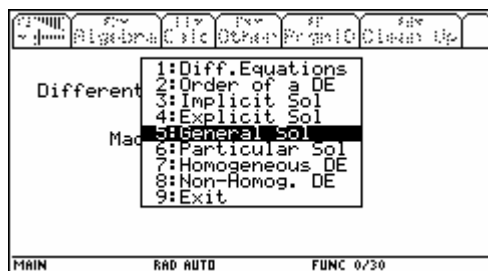
Nils and I had an extended exchange of emails and files. In some details he could refer to earlier DNL contributions.

I will give an overview about the many features of DEQME.

First of all see screen shots which show all options offered in the various menus reaching from Basics of 1st order DEs over PDEs to special DEs, Laplace Transforms and Eigenvalues.

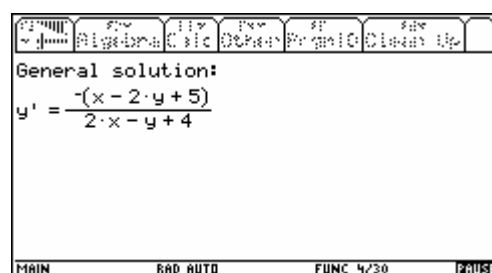
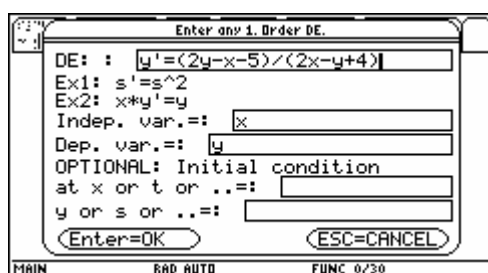


I open the first menu point under F1 1.Order, 1:Basics and want to inform about General Sol(ution):

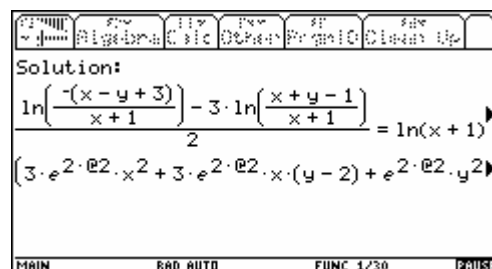
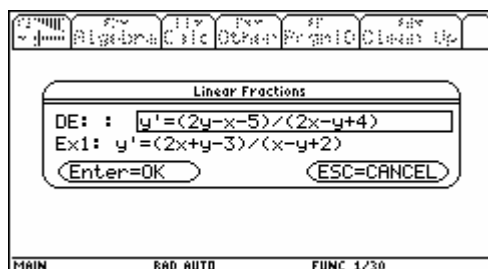


Then I switch back to F1, 2:Any 1.Order DE and would like to solve the differential equation

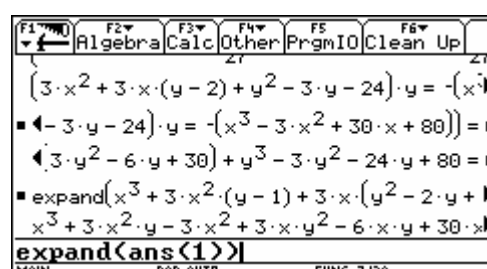
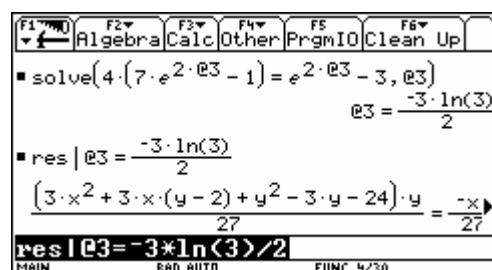
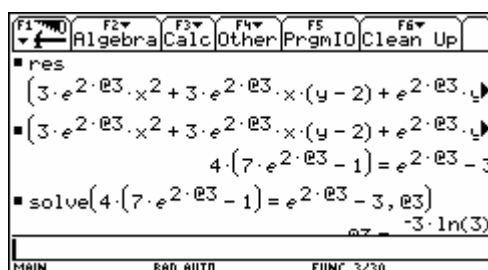
$$y' = \frac{2y-x-5}{2x-y+4}; y(0) = 4.$$



I don't receive the solution because the TI built in **desolve** cannot solve this kind of DE. Under F1 I can find the option G:Linear Fractions. So I try again:



This is the general solution, but how to obtain the special solution? The solution which is presented in the Prgm IO screen is stored as **res** and can be recalled in the Home screen. In F7 Menu you can find the respective note.



Just for checking the result I load DERIVE and apply the LIN_FRAC-function:

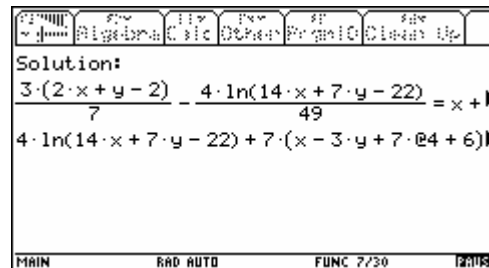
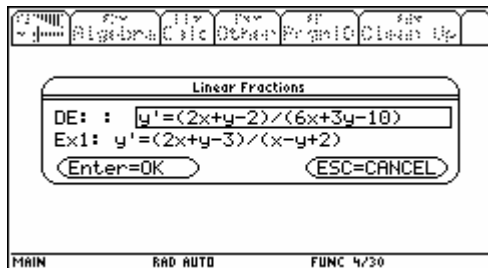
$$\text{LIN_FRAC}\left(\frac{2 \cdot y - x - 5}{2 \cdot x - y + 4}, -1, 2, -5, 2, -1, 4, x, y, 0, 4\right)$$

$$\frac{\text{LN}\left(-\frac{x - y + 3}{x + 1}\right)}{2} - \frac{3 \cdot \text{LN}\left(\frac{x + y - 1}{x + 1}\right)}{2} = \text{LN}(x + 1) - \frac{3 \cdot \text{LN}(3)}{2}$$

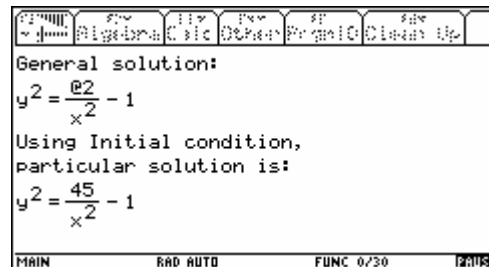
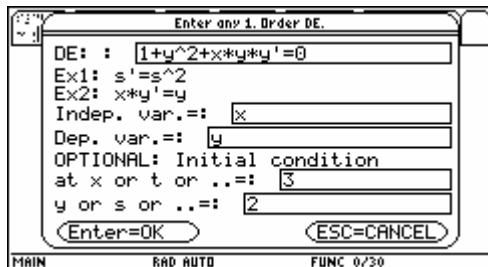
After some manipulations I obtain the same solution! Well done, DEQME!!

$$\#11: x^3 + 3 \cdot x^2 \cdot y - 3 \cdot x^2 + 3 \cdot x \cdot y^2 - 6 \cdot x \cdot y + 30 \cdot x + y^3 - 3 \cdot y^2 - 24 \cdot y = -80$$

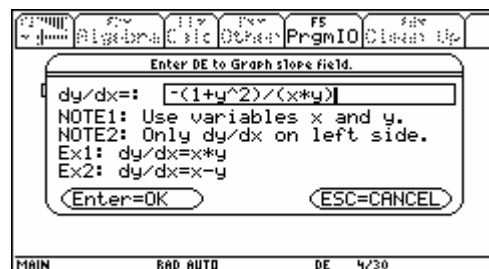
In case of non intersecting linear functions in numerator and denominator DERIVE provides a special utility function FUN_LIN_CFF_GEN. Nils implemented this special case as you can see below:



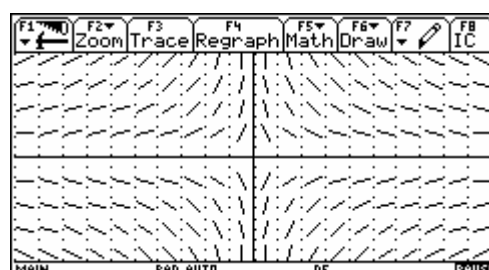
I try solving another DE using the first menu point "Any 1. Order DE" $1 + y^2 + x \cdot y \cdot y' = 0$; $y(3) = 2$.

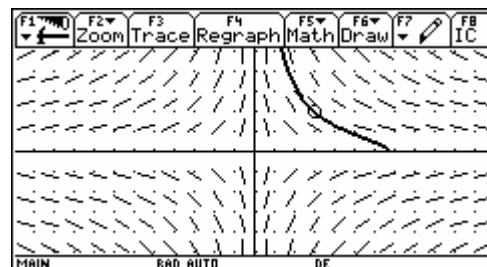
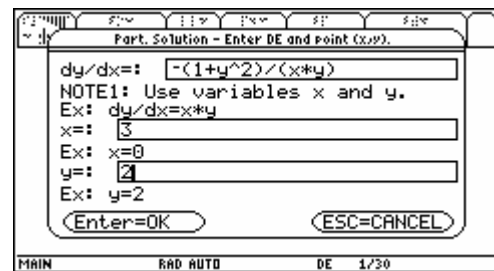
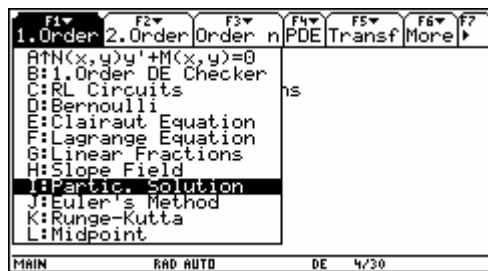


And I'd like to plot its slope field and the special solution.

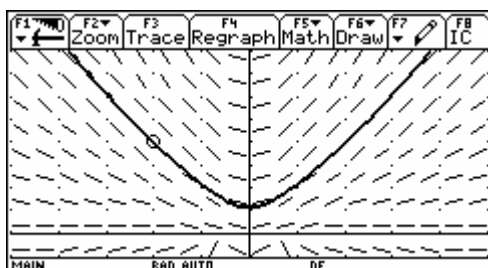
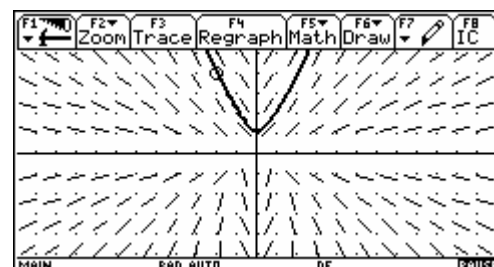
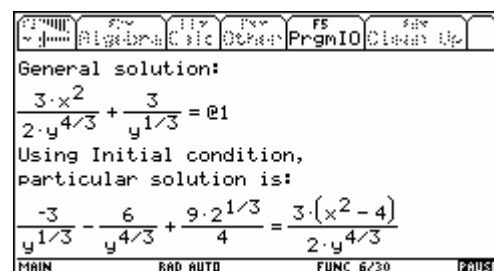


As you can read in the bottom line, the GRAPH Mode changed from FUNC to DE.

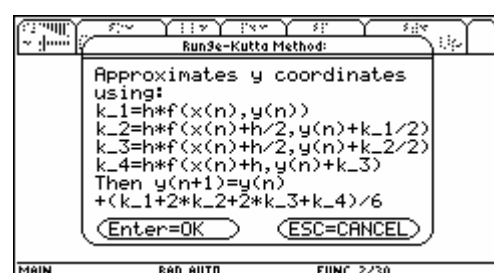
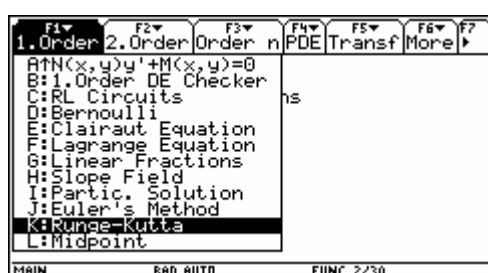




$y' = \frac{3xy}{2x^2 + y}$, $y(-2) = 4$. Solve and plot the solution together with the direction field.



Runge-Kutta is implemented in order to find numerical solutions.



Enter RK4 Information

dy/dx=: $3x*y/(2x^2+y)$

x=: -2

y(x)=: 4

Step Size Δx =: .05

Number of points=n=: 10

Enter=OK ESC=CANCEL

MAIN RAD AUTO FUNC 2/30

x=

-1.500

y=

3.044

Press Enter to continue

MAIN RAD AUTO FUNC 2/30

Next examples deal with separation of variables – and I switch to the TI-89:

F1+ Basics F2+ Any 1. Order DE F3+ Separable F4+ Homogeneous F5+ Exact F6+ Non-Exact F7+ Linear in x F8+ Linear in y

MAIN RAD AUTO FUNC 0/30

F1+ Enter F2+ Solve F3+ Steps F4+ Comp. F5+ Graph F6+ Euler F7+ Exit

1: dy/dx=f(x,y)
2: y'(t)=k*y(t)
3: y'(t)=k*(y(t)-H)
4: y'(t)=k*(A-y(t))
5: y'(t)=k*y*(A-y)

Differential Equations

MAIN RAD AUTO FUNC 0/30

Newton's Law of Cooling

dy/dt=k*(y-A) has analytic Solution:
 $y=A+Ce^{kt}$

its Graph looks as follows:

Enter=OK ESC=CANCEL

MAIN RAD AUTO FUNC 0/30

Enter Differential Equation

dy/dt=k*(y-A)

Below, enter value for A.
Use F1 if only 1 cond. is given.

dy/dt=: k*(y-20)

when t0=: 5

y(t0)=: 50

when t1=: 25

y(t1)=: 12

Enter=OK ESC=CANCEL

MAIN RAD AUTO FUNC 0/30

F1+ Enter F2+ Solve F3+ Steps F4+ Comp. F5+ Graph F6+ Euler F7+ Exit

1: dy/dx=f(x,y)
2: y'(t)=k*y(t)
3: y'(t)=k*(y(t)-H)
4: y'(t)=k*(A-y(t))
5: y'(t)=k*y*(A-y)

Differential Equations

MAIN RAD AUTO FUNC 0/30

y = 50 * e^{k*t} - 5 * k - 20

with: k = -ln(5/3)/10

k = -.051

y = $\frac{50 \cdot \sqrt{15} \cdot (3/5)^{10}}{3} - 20$

MAIN RAD AUTO FUNC 0/30

I'd like to see the calculation steps:

F1+ Enter F2+ Solve F3+ Steps F4+ Comp. F5+ Graph F6+ Euler F7+ Exit

1: Show Steps

TYPE OR USE \leftrightarrow F1+ (ENTER) OR (ESC)

MAIN RAD AUTO FUNC 0/30

Enter Separable Differential Eqn.

dy/dx=: $y \cdot \ln(y) / \sin(x)$

Ex1) dy/dx=y/(2*x)
Ex2) dy/dx=x*y
Ex3) dy/dx=1+y^2

Initial value (finds c):

x=: $\pi/2$

y=: 1/2

Ex1) x=1, y=2

Enter=OK ESC=CANCEL

MAIN RAD AUTO FUNC 0/30

0) Differential Eqn:

$\frac{dy}{dx} = \frac{y \cdot \ln(y)}{\sin(x)}$

1) Separate Variables:

$\int 1/(y \cdot \ln(y)) dy = \int 1/\sin(x)$

MAIN RAD AUTO FUNC 0/30

2) Integrate:

$\ln(\ln(y)) = \ln\left(\frac{\sin(x)}{\cos(x)+1}\right) + c$

3) Solve for y:

$\frac{\sin(x)}{\cos(x)+1} \geq 0$ and $y = e^{\frac{e^c \cdot \sin(x)}{\cos(x)+1}}$

MAIN RAD AUTO FUNC 0/30

3) Solve for y:

$\frac{\sin(x)}{\cos(x)+1} \geq 0$ and $y = e^{\frac{e^c \cdot \sin(x)}{\cos(x)+1}}$

4) Find c using $(\pi/2, 1/2)$:

false

MAIN RAD AUTO FUNC 0/30

5) Particular Solution is:

$\ln(\ln(y)) + \ln(\ln(2)) - \pi \cdot i \Rightarrow$

$y = \frac{1}{\sin(x)} \cdot \frac{1}{2 \cos(x)+1}$ and sign(si

MAIN RAD AUTO FUNC 0/30

5) Particular Solution is:

$\ln(\ln(y)) + \ln(\ln(2)) - \pi \cdot i \Rightarrow$

$y = \frac{1}{\sin(x)} \cdot \frac{1}{2 \cos(x)+1}$ and sign(si

MAIN RAD AUTO FUNC 0/30

Compare with the deSolve – result. It might be a nice problem for students to verify the identity of the trig expressions.

F1+ Tools F2+ Algebra F3+ Calc F4+ Other F5+ PrgmIO F6+ Clean Up

deSolve(y' = $\frac{y \cdot \ln(y)}{\sin(x)}$, x, y)

$\ln(\ln(y)) = \ln\left(\tan\left(\frac{x}{2}\right)\right) + @2$

MAIN RAD AUTO FUNC 1/30

F1+ Tools F2+ Algebra F3+ Calc F4+ Other F5+ PrgmIO F6+ Clean Up

$\ln(\ln(y)) = \ln\left(\tan\left(\frac{x}{2}\right)\right) + @2$

■ solve($\ln(\ln(y)) = \ln\left(\tan\left(\frac{x}{2}\right)\right)$

$y = e^{e^{2 \cdot \tan\left(\frac{x}{2}\right)}}$ and $\tan\left(\frac{x}{2}\right)$

■ $\tan\left(\frac{x}{2}\right) | x = \pi/2$ and $y = 1/2$

MAIN RAD AUTO FUNC 2/30

F1+ Tools F2+ Algebra F3+ Calc F4+ Other F5+ PrgmIO F6+ Clean Up

$y = e^{e^{2 \cdot \tan\left(\frac{x}{2}\right)}}$ and $\tan\left(\frac{x}{2}\right)$

■ $y = e^{e^{2 \cdot \tan\left(\frac{x}{2}\right)}}$ and $\tan\left(\frac{x}{2}\right)$

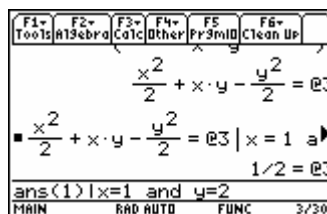
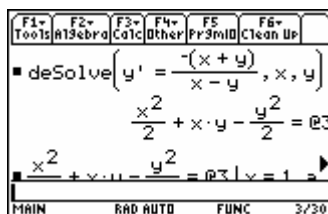
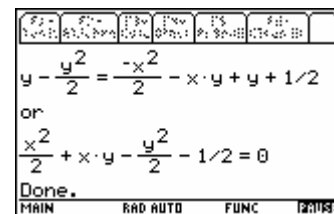
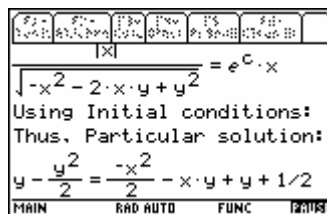
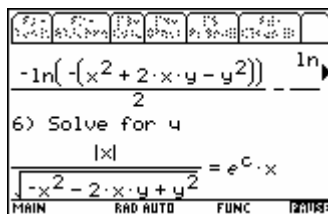
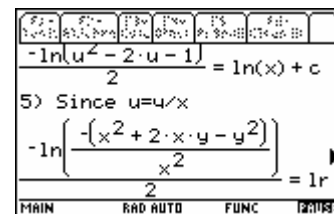
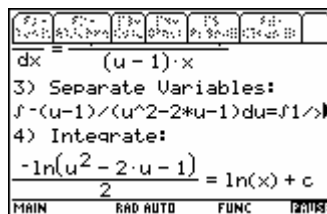
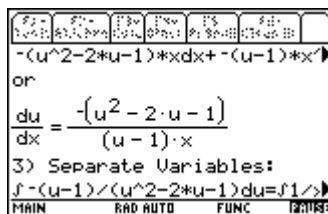
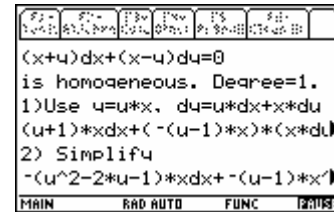
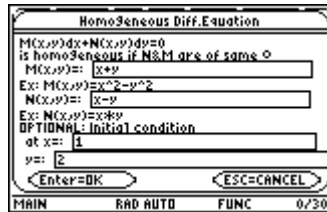
$1/2 = e^{e^{2 \cdot \tan\left(\frac{x}{2}\right)}}$

■ solve(ans(1), @2)

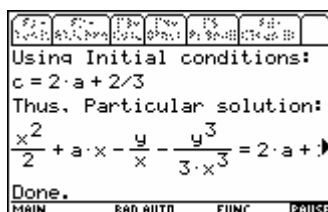
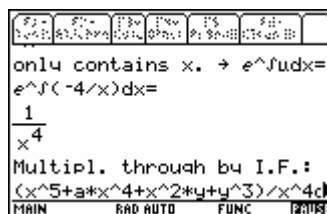
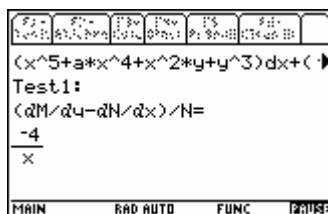
MAIN RAD AUTO FUNC 3/30

$y' = \frac{x+y}{y-x}$; $y(1) = 2$. This is a homogeneous DE.

Again I'd like to follow the steps. Then I will compare with deSolve.



How to find an integrating factor-- if there is any? DEQME gives the answer:



(will be continued)

Titbits(37) - Factoring Integers with DERIVE 2

(c) Johann Wiesenbauer, Vienna University of Technology

This is a sequel of my last titbits where I started with the discussion of several algorithms for the factorization of integers, namely Pollard's two factoring methods (rho and p-1) as well as Lenstra's ECM. In the following I'll talk about another basic idea when it comes to factoring integers, which deals with nontrivial solutions x, y of the congruence $x^2 = y^2 \pmod n$, where n is again the positive integer to be factored and "non-trivial" means here that $x \not\equiv \pm y \pmod n$. Since due to these conditions n is a divisor of the product $(x+y)(x-y)$, but not of its factors $x+y$ and $x-y$, it easily follows from this that both $\gcd(x+y, n)$ and $\gcd(x-y, n)$ will be nontrivial divisors of n .

This simple idea can be exploited in several ways, but in the following I'll focus on the so-called quadratic sieve only. Here one considers polynomials $Q(x)$ in $\mathbb{Z}[x]$ of the form

$$Q(x) = (ax+b)^2 - n = af(x), \text{ with } f(x) = ax^2 + 2bx + c \text{ and } c = (b^2 - n)/a \text{ an integer}$$

and their values in the interval $I = [-m, m]$ that contains both real zeros and has the property that the extrema of $Q(x)$, located at the endpoints of I and near its center, have about the same absolute size. This will guarantee that the values of $f(x)$ for x in I are relatively small, which is important as we want $f(x)$ to be B -smooth for a bound $B > 0$ of moderate size, that is we are only interested in values of x in I such that $p \leq B$ for all prime divisors of $f(x)$. Well, actually we also consider also the cases where this is "almost" true, in the sense that it is true except for at most one not too divisor p , exceeding B not too much, say $B < p \leq 10 \cdot B$. The underlying idea is that if we find two representations $z_1^2 = y_1 \cdot p \pmod n$ and $z_2^2 = y_2 \cdot p \pmod n$ with y_1 and y_2 both B -smooth numbers and some common prime $p > B$, then combining them like this

$$(z_1 \cdot z_2 / p)^2 = y_1 \cdot y_2 \pmod p$$

yields a representation where now the right side is perfectly B -smooth. This situation will occur surprisingly often, which is also known as "birthday paradox".

As for the choice of a , there are several possibilities, but we choose here one of the simplest, where $a = q^2 \sim \sqrt{2n}/m$ for some prime q . Furthermore, we choose b to be a solution of $x^2 = n \pmod a$ with $|b| < a/2$. These choices of a will guarantee that the coefficient c of $f(x)$ is an integer indeed, and that the extremal values of $f(x)$, located at the endpoints and the center of I , have about the same absolute size.

D-N-L#74	Johann Wiesenbauer: Titbits 37	p33
----------	--------------------------------	-----

Since $Q(x)$ is a square mod n and a fortiori also mod p for any prime divisor p of n , we need only consider primes p for which n is a square mod p . This is always fulfilled for $p=2$ and for an odd prime p equivalent to $\text{jacobi}(n,p)=1$, where $\text{jacobi}()$ is the so-called Jacobi symbol, which exists as a library function in DERIVE. Hence, if we consider the set

$$FB = \{-1,2\} \cup \{p \leq B \mid p \text{ is an odd prime and } \text{jacobi}(p,n)=1\}$$

then we want to factor $f(x)$ using only numbers in FB . For this reason, FB is also called a factorbase w.r.t B , denoted by f in the program.

Lets start our implementation with the fundamental routine $\text{setup}(v,P)$ that sets the values of a number of important global variables, namely

n = the number v to be factored,

b = upper bound for the primes in f (if this parameter is omitted, it will be chosen by the program in a reasonable way)

f = factorbasis consisting of $-1,2$ and the odd primes p below b such that n is a quadratic residue mod p ,

m = $\text{ceiling}(b/2)$, which is chosen in such a way that the interval length of $I=[-m,m]$ is about b ,

fp = product of all primes in f , which will be needed later for technical reasons

User:

```

      setup(v, β := 0) :=
      Prog
        n := v
        b := IF(β > 0, β, CEILING(EXP(0.43*√(LN(n)*LN(LN(n))))))
#1:    f := APPEND([-1, 2], SELECT(PRIME(q_) ^ JACOBI(n, q_) = 1, q_, 3, b))
        m := CEILING(0.3*b)
        fp := Π(REST(f))
      true

```

User=Simp(User):

```
#2:                                setup(24961) = true
```

User=Simp(User):

```
#3:    [n, b, f, m, fp] = [24961, 9, [-1, 2, 3, 5], 3, 30]
```

The next routine $\text{getrel}()$ is supposed to collect a sufficiently large number k of relations, i.e. pairs (u_i, v_i) , such that

$$(u_i)^2 = v_i \text{ mod } n, \quad i=1,2,\dots,k$$

and v_i is b -smooth, where u_i and v_i are obtained by using polynomials $(ax+b)^{2-n}$ for various values of a and b as outlined above. The output of `getrel()` is a vector x_{rel} , containing the u_i , and a matrix E_{rel} , whose row vectors represent the exponents of the prime factorizations of the v_i over the factor basis f .

```
getrel(k := 0, a_, b_, c_, d_, e_, i_ := 0, k_ := 0, m_, p_, q_, x_, y_, u_ := [], v_ := [], w_ := []) :=
  Prog
  If k = 0
    k := DIM(f) + 1
    p_ := FLOOR((2*n)^(1/4)/sqrt(m))
    m_ := MIN(b^2, 10*b)
    e_ := ADJOIN(0, VECTOR(1, j_, DIM(f) - 1))
    Loop
      i_ := i_ + 1
      Loop
        p_ := NEXT_PRIME(p_)
        If JACOBI(n, p_) = 1 exit
        a_ := p_^2
        b_ := SQUARE_ROOT(n, p_)
        b_ := (n - b_^2) * INVERSE_MOD(2*b_, p_)
        b_ := MODS(b_, a_)
        c_ := (b_^2 - n)/a_
        q_ := INVERSE_MOD(p_, n)
        x_ := -m
      Loop
        y_ := a_*x_^2 + 2*b_*x_ + c_
        d_ := fp
        Loop
          d_ := GCD(y_, d_)
          If d_ = 1 exit
          y_ := y_/d_
        Loop
          If ABS(y_) <= m_
            Prog
```

(This is not the complete code of the program `getrel()`. Please refer to the DERIVE file, Josef)

User=Simp(User):

#5: `getrel()` = true

User=Simp(User):

#6: $[x_{rel}, E_{rel}] = \left[\begin{bmatrix} 22789 \\ 23065 \\ 23052 \\ 23039 \\ 23026 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 2 & 1 \\ 0 & 4 & 3 & 0 \\ 1 & 0 & 0 & 2 \\ 1 & 4 & 2 & 0 \\ 0 & 0 & 1 & 2 \end{bmatrix} \right]$

User=Simp(User):

#7: $SOLVE(MOD(22789^2, n) = MOD(-2^0 \cdot 3^2 \cdot 5^1, n)) = true$

User=Simp(User):

#8: $SOLVE(MOD(23065^2, n) = MOD(2^4 \cdot 3^3 \cdot 5^0, n)) = true$

User=Simp(User):

#9: $SOLVE(MOD(23052^2, n) = MOD(-2^0 \cdot 3^0 \cdot 5^2, n)) = true$

User=Simp(User):

#10: $SOLVE(MOD(23039^2, n) = MOD(-2^4 \cdot 3^2 \cdot 5^0, n)) = true$

User=Simp(User):

#11: $SOLVE(MOD(23026^2, n) = MOD(2^0 \cdot 3^1 \cdot 5^2, n)) = true$

The next point on our agenda is to find subset's of all the v_i , such that their product is a perfect square, i.e. has a vector representation with only even numbers, which is the null vector 0 of suitable size when considered mod 2. In other words, we must find vectors x such that $x \cdot E_{rel} = 0 \pmod{2}$. Those vectors form a subspace of the corresponding vector space, which can be given by a basis.

User:

```

kernel(c, e_, i_, j_ := 0, p_, t_, u_, v_, w_) :=
  Prog
    c := MOD(c, 2)
    t_ := DIM(c')
    e_ := IDENTITY_MATRIX(DIM(c))
    u_ := {1, ..., DIM(c)}
    Loop
      Loop
        j_ :=+ 1
        If j_ > t_
          Prog
            d := SELECT( $\sum(c \downarrow k_)$  = 0, k_, 1, DIM(c))
            RETURN VECTOR(e_  $\downarrow$  k_, k_, d)
#12:      v_ := SELECT( $c \downarrow k_ \downarrow j_ = 1$ , k_, u_)
          If v_  $\neq$  {} exit
          p_ := FIRST(v_)
          u_ := u_ \ {p_}
          v_ := v_ \ {p_}
          Loop
            If v_ = {} exit
            i_ := FIRST(v_)
            w_ := MOD( $c \downarrow i_ + c \downarrow p_$ , 2)
             $c \downarrow i_ := w_$ 
            w_ := MOD( $e_ \downarrow i_ + e_ \downarrow p_$ , 2)
             $e_ \downarrow i_ := w_$ 
            v_ := REST(v_)

```

User=Simp(User):

$$\#13: \quad \text{kernel}(E_{rel}) = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

User=Simp(User):

$$\#14: \quad \begin{bmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 2 & 1 \\ 0 & 4 & 3 & 0 \\ 1 & 0 & 0 & 2 \\ 1 & 4 & 2 & 0 \\ 0 & 0 & 1 & 2 \end{bmatrix} = \begin{bmatrix} 2 & 4 & 2 & 2 \\ 0 & 4 & 4 & 2 \end{bmatrix}$$

The computation above shows that we are very close to our ultimate goal of finding solutions of $x^2 = y^2 \pmod{n}$, by simply combining the components of x_{rel} and the row vectors of E_{rel} in the way indicated above by the 1's in the vectors of the kernel k , as well as of the the matrix $(k E_{rel})/2$. See the examples below for details.

User=Simp(User):

$$\#15: \quad \text{SOLVE} \left(\text{MOD} \left(\text{MOD} \left(x_{\text{rel}}^1 \cdot x_{\text{rel}}^1, n \right)^2, n \right) = \text{MOD} \left((-2^2 \cdot 3^1 \cdot 5^1)^2, n \right) \right) = \text{true}$$

Simp(User'):

$$\#16: \quad \text{SOLVE}(\text{MOD}(24792^2, n) = \text{MOD}((-60)^2, n)) = \text{true}$$

User=Simp(User):

$$\#17: \quad [\text{GCD}(24792 - 60, n), \text{GCD}(24792 + 60, n)] = [229, 109]$$

User=Simp(User):

$$\#18: \quad \text{SOLVE} \left(\text{MOD} \left(\text{MOD} \left(x_{\text{rel}}^1 \cdot x_{\text{rel}}^1, n \right)^2, n \right) = \text{MOD} \left((2^2 \cdot 3^2 \cdot 5^1)^2, n \right) \right) = \text{true}$$

Simp(User'):

$$\#19: \quad \text{SOLVE}(\text{MOD}(24454^2, n) = \text{MOD}(180^2, n)) = \text{true}$$

User=Simp(User):

$$\#20: \quad [\text{GCD}(24454 + 180, n), \text{GCD}(24454 - 180, n)] = [109, 229]$$

Ok, it's high time we put all this together into one single routine that will do all the steps above for us. Here, o is an offset as to the number of relations with regard to the size dim(f) of the factor base (usually the default value o=1 will do) and the memory consuming matrices x_rel and E_rel will be purged by default after their use. (This may help to keep the file small, if you want to save the session after applying qs() to a huge number n, as global variables are also stored.) I also included in a textbox the output of all the divisions of n found by this routine, trivial as well as nontrivial ones, so you might get a better feeling for the overall success rate of this routine.

User:

```

qs(n, o := 1, dispose := true, d_, k_, x_, y_) :=
  Prog
    setup(n)
    getrel(DIM(f) + o)
    k_ := kernel(E_rel)
    If k_ = []
      RETURN "Too small number of relations!"
    x_ := FIRST(EXP(k_.LOG(x_rel)))')
#21:  y_ := EXP(k_.E_rel/2.LOG(f))
    If dispose
      [x_rel :=, E_rel :=]
    d_ := VECTOR(GCD(z_, n), z_, x_ + y_)
    DISPLAY(d_)
    d_ := SELECT(1 < z_ < n, z_, d_)
    If d_ = []
      RETURN 1
    MIN(d_)

```

[229, 109]

User=Simp(User):

$$\#22: \quad \text{qs}(24961) = 109$$

D-N-L#74	Johann Wiesenbauer: Titibits 37	p37
----------	---------------------------------	-----

Well, this was a rather small number, hence let's conclude by trying a bigger one, viz. the notorious Mersenne number $n=2^{67}-1$, which Mersenne himself thought mistakenly to be prime. Needless to say that for the numbers of this size the overhead of the quadratic sieve is still too big to be a real competitor to the other factoring methods we have been discussing so far. All the same, if you have a look at the huge matrix below you might still wonder how Derive can be so fast, namely less than 3s on my PC! As always, have fun when trying out examples of your own!

User=Simp(User):

```
#23:                      67
                      setup(2  - 1) = true
```

User=Simp(User):

```
#24:    [n, DIM(n), b, DIM(f), m] = [147573952589676412927, 21, 312, 31, 94]
```

User=Simp(User):

```
#25:                      getrel() = true
```

User:

```
#26:    [x_rel, E_rel]
```

Simp(#26):

85138228182631575566	1	1	2	2	1	0	0	1	1	0	0	2	0	0	0	0
25432000691266516464	2	1	6	1	2	1	0	1	1	0	0	0	0	0	0	0
27131959020989732516	1	0	2	1	2	2	2	0	1	0	0	0	0	1	0	0
91834232634966827117	0	1	2	1	0	2	0	2	0	0	0	0	0	0	0	0
21205539602124754815	1	1	2	2	0	0	0	1	1	0	0	0	0	0	1	0
138631399311064240574	0	2	5	1	1	1	0	0	0	0	1	0	0	0	2	0
134120451326901583109	0	1	0	0	0	0	1	0	1	0	1	0	1	0	0	0
17080547131876123141	0	1	1	1	1	1	1	0	1	0	0	0	0	0	0	0
61343671582884063679	1	1	3	1	0	0	2	0	1	1	0	0	1	0	0	1

(This is again only a part of the output. Please refer to the file, Josef)

User:

```
#28:    kernel(E_rel)
```

Simp(#28):

```
#29:    [ 0  0  0  1  1  0  1  0  0  1  0  1  1  1  0  1  0  0  1  1  1  0  1  0
          0  0  1  0  1  0  1  0  1  0  1  1  1  0  1  1  0  0  1  0  1  1  1  0
          0  1  0  1  0  1  1  1  0  1  0  0  1  1  0  1  0  0  0  0  1  0  1  0
          1  1  0  1  0  0  0  0  0  0
          1  1  0  0  0  1  0  0
          0  0  0  0  0  0  1  1 ]
```

```
[761838257287, 193707721, 147573952589676412927]
```

User=Simp(User):

```
#30:                      67
                      qs(2  - 1) = 193707721
```

These are the abstracts of the special session "*Applications and Libraries development in Derive*"

**G. Aguilera, J. L. Galán, M. À. Galán, Y. Padilla, P. Rodríguez,
University of Málaga**

Generating random Samples from continuous and discrete distributions

In this talk we will introduce the utility file RandomDistributions.mth. This file has been developed for generating random values from main continuous and discrete distributions. The programs contained in the file can be grouped within the following blocks:

- **Random values from uniform distribution:** the program RandomUniform returns an uniform random sample between 0 and 1. This program is the base of the other generations. Different algorithms have been used to develop this program in order to improve the Derive's build-on function to generate samples from a continue uniform random distribution.
- **Random values from discrete distributions:** some generic algorithms for any discrete distribution have been implemented as well as specific algorithms for some discrete distributions (Uniform, Poisson, Binomial, Geometric, Negative Binomial, ...).
- **Random values from continuous distributions:** specific algorithms for main continuous distributions have been implemented (Uniform, Exponential, Normal, Log-normal, Cauchy, Chi-square, Student's t, F, Gamma, Beta, ...).
- **Graphical approach:** a program to plot the obtained samples together with the density function has been developed. With these drawings we can check graphically if the generated samples fit the distributions.

The use of this utility file is useful for simulating any process which follows a specific distribution.

Michel Beaudin, École de technologie supérieure, Montréal
Another Look at a Trusted Mathematical Assistant

From the DERIVE user manual (version 3, September 1994), we can read the following: "Making mathematics more exciting and enjoyable is the driving force behind the development of the DERIVE program". In this talk, we will try to show how some mathematical concepts, studied by engineering students at university level - differential equations, multiple variable calculus, systems of non linear equations -, can be easily illustrated by DERIVE. Some will object that any other CAS could do the same: well, this is probably true but, according to us, not as quickly and naturally: "To accomplish this DERIVE not only has to be a tireless, powerful and knowledgeable mathematical assistant, it must be an easy, natural, and convenient tool". Consequently, time can be spent to prove some theorem or formula and the computer algebra system helps to reinforce the mathematical concepts. Our examples will also make use of new features added in the latest version of DERIVE (version 6.10 released in October 2004); features that were not exploited as should be - DERIVE has never been enough documented. But we are still convinced that Derive 6 was "Far too good just for students" (<http://www.scientific-computing.com/scwmarapr04derive6.html>).

Josef Böhm, ACDCA & Technical University of Vienna
CAS Tools for Exercising

There is no doubt that even in times of CAS a certain amount of manipulating skills in various fields of math education is still necessary.

Students need more or less exercising for mastering expanding and factorizing expressions, finding GCD and LCM, solving triangles, applying differentiation and integration rules to name only some of the fields where training of skills might be useful.

We present a respective library developed in Derive reaching from set theory to calculus, which can support the students (and teachers as well) offering random generated problems together with the respective solutions.

A. García, F. García, G. Rodríguez, A. de la Villa,
Univ. de Madrid, Univ. de Salamanca, Univ. Pontificia Comillas
Toolboxes with DERIVE

The European Area of Higher Education implies a profound change in the Spanish university. We are heading towards a competency-based teaching and a learning model with greater autonomy for the student, who becomes the centre of the educational model. New technologies can play an important role in this new scenario. This paper suggests a "new" possibility in the use of new technologies: The design of a "toolbox" which could be used later on by the student when needed in other subjects of their curriculum or in their careers. This "toolbox" will contain instructions, which may either be suggested by the teacher or developed by the student as being interesting and convenient so as to explore the mathematical concepts associated with them.

There are two examples of possible "toolboxes." In the first one, instructions are related to **geometrical aspects** of the plane which are studied in high school. The instructions are concerning with drawings, equations of different geometric objects and some distance between the quoted objects.

In the second one, a toolbox is built for the subject **Calculus of a single variable** in engineering studies. In this toolbox there can be instructions about complex numbers to express them in their different expressions, calculate their power, roots, etc. It also incorporates the instructions that enable to analyze the concepts of differential calculus such as the tangent, the study of increasing and convexity, etc.

About integral calculus several tools can be included for the geometric applications. Also procedures for the trapeze method and the Simpson method are implemented.

In the study of approximate methods of solving equations we can encourage students to use algorithms NEWTON and FIXED_POINT, which DERIVE has incorporated, and design a procedure for the method of the bisection of the interval.

**E. Roanes Lozano, F. A. González Redondo,
Univ. Complutense de Madrid**

An Implementation on the Mayan numbering system in DERIVE

The Mayan number system is a base 20, positional (to be read from top to bottom, not from left to right) system that makes use of a symbol representing zero. It has slightly different variations when used for counting days (in religious and astronomical contexts).

Therefore, 20 symbols are needed to represent 0,1,2,...,19. Of these, the positive ones were represented using dots (the value of each dot is 1), and horizontal segments (the value of each segment is 5), while the zero was denoted by a shell (see figure below).

If a number is greater than 20, the symbols corresponding to units, twentieths, 400's, 8000's... are stacked from bottom (units) to top in pure base 20, while in our decimal system, the different orders correspond to tenths, hundreds, thousands, etc.

We can choose the input and output bases in DERIVE in Options > Mode Settings from 2, 8, 10 and 16. Therefore we have implemented a procedure that allows to convert numbers between any bases, and that returns the output in (row) vector style. We have implemented another procedure that builds the 20 Mayan symbols for 0,1,2,...19 (making use of the DISPLAY command). Finally, another procedure (denoted Maya), that uses the previously mentioned procedures, converts any number from base 10 to base 20 and represents it in the Mayan numbering system. These procedures only make use of the standard DERIVE commands.

We do not know of any other similar implementations in CASs, apart from [1] (that uses special facilities for inserting graphics) and the similar [2], by the same authors of this paper, but written in Maple.

We believe this is an interesting example of synergy among different branches of knowledge (Mathematics, History of Mathematics and Computer Science), that can increase the interest of the students for different topics.

K. Schmidt, Schmalkalden University of Applied Sciences

Making Life in an Introductory Linear Algebra Course Easier with DERIVE

In teaching linear algebra we have to deal with the following problem: while the level of the mathematical skills which are required to work with examples is generally low (students only need to add, subtract and multiply), the number of calculations is usually large. Therefore, working with examples is time-consuming and error-prone if done by hand. Students get tired quickly and lose their interest in this increasingly important area of mathematics. The faculty therefore decided to move the introductory linear algebra course from the classroom to the PC lab, and acquired a special DERIVE license that allows its use on all the PCs the faculty owns, and also on the private PCs of the students.

A utility file was then developed to facilitate teaching by providing functions for the computation of zero matrices and vectors, matrices and vectors of ones, as well as idempotent and orthogonal matrices, "just-in-time" whenever they are needed during the course. The utility file also contains functions that test if a given matrix is symmetric, idempotent or orthogonal.

Jim FitzSimons

DERIVE or Maple can not factor this. $x^5 - 5x + 12$

I have been told a radical root exists for this polynomial.

Richard Schmitt

[schmitttrichard@yahoo.com]

Suggestion:

"The multi-valued nature of inverse functions" by Jeffrey

Available at

<http://www.apmaths.uwo.ca/~djeffrey/offprints.html>

I have a question, too:

Is the Lambert W function available in Derive 6? Thank you.

Jim FitzSimons

Here is all I have:

#1: PrecisionDigits := 64

#2: NotationDigits := 64

#3: InputMode := Word

#4: f1(y) := y + LN(y)

f6(x) uses iteration.

#5: f6(x) := ITERATE(e^{x-y} , y, 0.27, 10)

f8(x) uses Newton's method.

#6: f8(x, y0) := ITERATE $\left(-\frac{y \cdot (\text{LN}(y) - x - 1)}{y + 1}, y, y0, 32\right)$

f9(x) :=

If $x = -1 + \pi \cdot i$ v $x = -1 - \pi \cdot i$

#7: $\frac{-1}{f8(x, \text{IF}(\text{RE}(x) < -1, f6(x), 0.3))}$

#8: Wrightomega(z) := f9(z)

Test the results.

#9: a1 := VECTOR([x, f9(x)], x, -4, 4, 0.05)

$\text{VECTOR}\left(\frac{f1(y)}{2} - y, y, a1\right)$

#10:

#11: a2 := VECTOR([x, f9(LN(x))], x, -4, 4, 0.05)

I must admit that I have not heard about Lambert's W function before. I found some useful information in CRC Concise Encyclopedia of Mathematics:

Lambert's W-function is the inverse of the function $f(W) = W e^W$. $W(1)$ is called *Omega Constant* and can be considered a sort of "Golden Ratio" of exponentials since

$$e^{-W(1)} = W(1).$$

I checked Jim's procedure calculating the Omega Constant:

```
#12: [1, f9(LN(1))]
```

```
#13: [1, 0.5671432904097838729999686622103555497538157871865125081351310792]
```

```
#14: NSOLVE(1 = x·ex, x)
```

```
#15: x = 0.5671432904097838729999686622103555671469246551420461565531033239
```

```
#16: e-0.5671432904097838729999686622103555497538157871865125081351310792
```

```
#17: 0.5671432904097838729999686622103555497538157871865125081351310792
```

In *MATHEMATICA* is this function implemented as `ProductLog[z]`:

```
In[1]:= ProductLog[1]
```

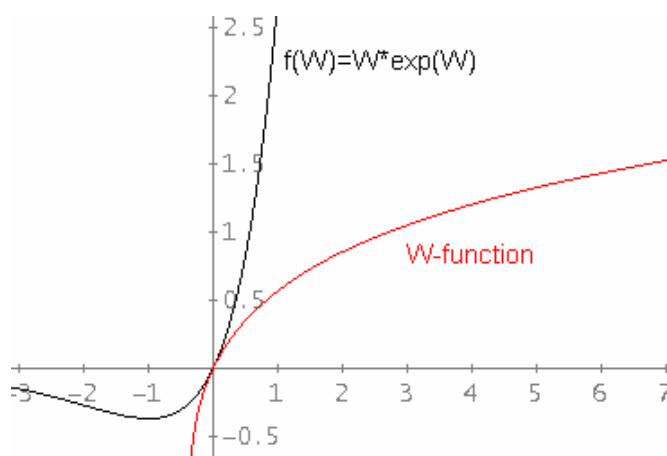
```
Out[1]= W(1)
```

```
In[2]:= ProductLog[1.0]
```

```
Out[2]= 0.567143
```

```
In[4]:= N[ProductLog[1], 20]
```

```
Out[4]= 0.56714329040978387300
```



The graph shows $f(W)$ and its inverse – Lambert's W -function.

Among many other papers you can download another of D. Jeffrey's papers from the above mentioned website:

R.M. Corless, H. Ding, N.J. Higham, D.J. Jeffrey: The solution of $S \cdot \exp(S) = A$ is not always the Lambert W function of A . In: "ISSAC 2007", Editor C.W. Brown, pp 116--121, ACM Press, 2007

Nils Hahnfeld, Virgin Islands

Hallo Josef

How can I convince the TI-89 that the 2×2 matrix $\begin{bmatrix} 3 & 2 \\ -2 & -1 \end{bmatrix}$ does not have complex Eigenvalues but twice the Eigenvalue 1. Do you know any tricks? Or is it necessary to take the imaginary part < 0.000001 as 0? Would be nice if the next OS would perform better ...

Three more questions:

- Is it possible to check the language setting of the device within a program?
- Is it possible to present an Integral in pretty print without performing an integration?
- When will it be possible to have a dialogue in TI-Nspire programs?

Best regards

Nils

These are the matrices which were sent. The first one gives the strange Eigenvalues:

```

F1 F2 F3 F4 F5 F6
Algebra Calc Other PrgmIO Clean Up
■ eigU1  $\begin{bmatrix} 3 & 2 \\ -2 & -1 \end{bmatrix}$ 
   $\langle 1.000 + 5.292E-8 \cdot i \quad 1.000 - 5.292E-8 \cdot i \rangle$ 
■ eigU1  $\begin{bmatrix} 3 & 2 & 3 \\ -2 & -1 & 0 \\ 5 & -2 & 3 \end{bmatrix}$ 
   $\langle 6.819 \quad -.909 + 1.890 \cdot i \quad -.909 - 1.890 \cdot i \rangle$ 
■ eigU1  $\langle 3, 2, 3; 2, -1, 0; 5, -2, 3 \rangle$ 
MAIN RAD EXACT DE 2/30

```

```

F1 F2 F3 F4 F5 F6
Algebra Calc Other PrgmIO Clean Up
■ eigU1  $\begin{bmatrix} -1 & -3 & -3 \\ 0 & 2 & -1 \\ 2 & 2 & 5 \end{bmatrix}$ 
   $\langle 1.000 \quad 2.000 \quad 3.000 \rangle$ 
■ eigU1  $\begin{bmatrix} -2 & -2 & 2 \\ 4 & 5 & 0 \\ -4 & -2 & 3 \end{bmatrix}$ 
   $\langle i, 1.5 - 1.9364916731037 \cdot i, 3 \rangle$ 
MAIN RAD EXACT DE 1/4

```

```

F1 F2 F3 F4 F5 F6
Algebra Calc Other PrgmIO Clean Up
■ eigU1  $\begin{bmatrix} -1 & -3 & -3 \\ 0 & 2 & -1 \\ 2 & 2 & 5 \end{bmatrix}$ 
   $\langle 1.000 \quad 2.000 \quad 3.000 \rangle$ 
■ eigU1  $\begin{bmatrix} -2 & -2 & 2 \\ 4 & 5 & 0 \\ -4 & -2 & 3 \end{bmatrix}$ 
   $\langle i, 1.5 - 1.9364916731037 \cdot i, 3 \rangle$ 
MAIN RAD EXACT DE 1/4

```

DERIVE shows the correct Eigenvalue, but does not consider that 1 is a double Eigenvalue.

I try to solve the characteristic equation. This seems to work similar to the DERIVE procedure: I receive the Eigenvalue 1 but not its double occurency.

```

F1 F2 F3 F4 F5 F6
Control I/O Var Find... Mode
:my_eigv(m)
:Func
:cSolve(det(m-identity(dim(m)[1]))*x=0,x)
:EndFunc
MAIN RAD EXACT DE

```

```

F1 F2 F3 F4 F5 F6
Algebra Calc Other PrgmIO Clean Up
■ my_eigv  $\begin{bmatrix} 3 & 2 \\ -2 & -1 \end{bmatrix}$ 
   $\langle 1.500 + 1.936 \cdot i \quad 1.500 - 1.936 \cdot i \quad 3.00 \rangle$ 
  x = 1
■ my_eigv  $\begin{bmatrix} 3 & 2 & 3 \\ -2 & -1 & 0 \\ 5 & -2 & 3 \end{bmatrix}$ 
  x = -.909 + 1.890 · i or x = -.909 - 1.890 · i
■ my_eigv  $\langle [3, 2, 3] [-2, -1, 0] [5, -2, 3] \rangle$ 
MAIN RAD EXACT DE 6/30

```

```

F1 F2 F3 F4 F5 F6
Algebra Calc Other PrgmIO Clean Up
■ my_eigv  $\begin{bmatrix} 0 & 2 & -1 \\ 2 & 2 & 5 \end{bmatrix}$ 
  x = 1.000 or x = 2.000 or x = 3.000
■ my_eigv  $\begin{bmatrix} -2 & -2 & 2 \\ 4 & 5 & 0 \\ -4 & -2 & 3 \end{bmatrix}$ 
  x = 1.500 + 1.936 · i or x = 1.500 - 1.936 · i
■ my_eigv  $\langle [-2, -2, 2] [4, 5, 0] [-4, -2, 3] \rangle$ 
MAIN RAD EXACT DE 8/30

```

Now I check for a very small imaginary part (this is what you proposed to do). Multiple Eigenvalues are presented.

```

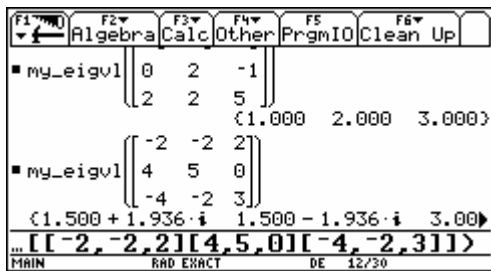
F1 F2 F3 F4 F5 F6
Control I/O Var Find... Mode
:my_eigv1(m)
:Func
:Local l_1,i,1
:eigU1(m)>l_1
:O+1
:For i,1,dim(l_1)
:If abs(imag(l_1[i]))≤10^(-6) Then
:  augment(l_1,real(l_1[i]))+1
:Else
:  augment(l_1,(l_1[i]))+1
:EndIf
:EndFor
MAIN RAD EXACT DE

```

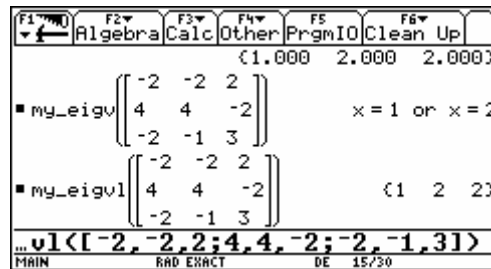
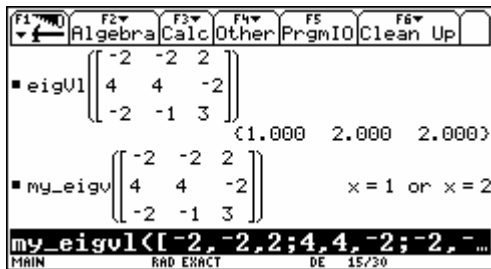
```

F1 F2 F3 F4 F5 F6
Algebra Calc Other PrgmIO Clean Up
x = 1.500 + 1.936 · i or x = 1.500 - 1.936 · i
■ my_eigv1  $\begin{bmatrix} 3 & 2 \\ -2 & -1 \end{bmatrix}$ 
   $\langle 1 \quad 1 \rangle$ 
■ my_eigv1  $\begin{bmatrix} 3 & 2 & 3 \\ -2 & -1 & 0 \\ 5 & -2 & 3 \end{bmatrix}$ 
   $\langle 6.819 \quad -.909 + 1.890 \cdot i \quad -.909 \rangle$ 
■ my_eigv1  $\langle [3, 2, 3] [-2, -1, 0] [5, -2, 3] \rangle$ 
MAIN RAD EXACT DE 10/30

```



Here is another example with a double solution of the characteristic equation:



I must admit that I don't have any idea why the TI delivers so strange Eigenvalues for your 2x2 matrix. (Ti-NspireCAS shows the same behaviour!)

To your other questions:

The language cannot be checked.

It is not possible to have the Integral character within a text.

We hope that future versions of Nspire will enable dialogues.

Josef

Big Air in Summertime

Tania Koller sent a model of a jumping snowboarder (as background picture in DERIVE). This is a very welcome occasion to cool off on hot summer days. Many thanks to Tania and her students.

