Grand Tour of *Derive*

André Heck
CAN Expertise Center

March 1995

Contents

1 General Information 3
1.1 Short Description . . . 3
1.2 Current Version . . . 3
1.3 Hardware Requirements . . . 3
1.4 User Interface . . . 4
1.5 Software Documentation . . . 6
1.6 Licensing Possibilities . . . 6
1.7 Distributors/Dealers . . . 6
1.8 Derive Addresses . . . 8
1.9 Long Description . . . 9
1.9.1 General Description . . . 9
1.9.2 Features . . . 9

2 A Tour of Derive 13
2.1 Launching Derive . . . 13
2.2 Arithmetic . . . 15
2.3 Graphics . . . 17
2.4 Algebraic Facilities . . . 24
2.4.1 Calculus . . . 24
2.4.2 Solving Systems of Equations . . . 28
2.4.3 Ordinary Differential Equations . . . 29
2.4.4 Matrices . . . 31
2.5 Numerical Calculus . . . 32
2.6 Programming . . . 34
2.6.1 Functional Programming . . . 34
2.6.2 Iteration . . . 35
2.6.3 Recursion . . . 36
2.7 Simplification Rules . . . 37
2.8 Derive Packages . . . 38
2.9 Formatting . . . 40
2.10 Interface . . . 40
2.11 Saving and Quitting . . . 42

3 Literature 43
3.1 Books, Manuals, and Reports . . . 43
3.2 Papers and Reviews . . . 47

Chapter 1

General Information

1.1 Short Description

Derive is a general purpose symbolic calculator which is continually being developed by David Stoutemyer and Albert Rich of Soft Warehouse, Inc. It is the successor of muMATH. It can handle many of the mathematical computations learned in secondary school, polytechnic, and university mathematics, and it can handle two- and three-dimensional graphing. *Derive* can be thought of as having a relationship to algebra and calculus similar to that of any ordinary calculator to basic arithmetic. *Derive* also incorporates some functional programming facilities that allow the user to define mathematical procedures. All of *Derive*'s own utility functions are written in the *Derive* language.

1.2 Current Version

The current version is *Derive* 3. At present, the exact version number is 3.02, indicating that minor improvements have been made since the major release upgrade.

1.3 Hardware Requirements

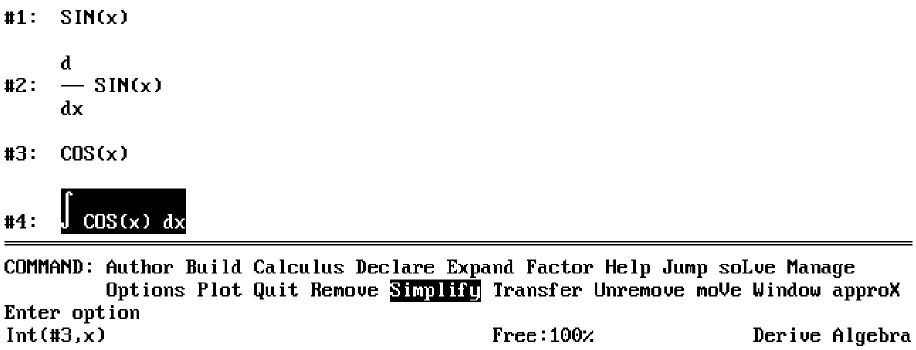
There are two versions of the *Derive* program: *Classic Derive*, and *Profesional Derive.* They operate identically; the difference is in memory usage.

Classic Derive requires a PC compatible computer with at least 512K of main memory, including RM Nimbus 186/286/386 and Archimedes with PC emulator, running MS-DOS (version 2.1 or higher), and with a 3.5" diskette drive.

Profesional Derive requires a 386 or 486 based PC compatible computer with at least 2MB of extended memory and can make use of up to 4GB of extended memory to solve large problems.

At present, *Derive* is the only computer algebra system which is available on palmtop PCs such as POQET PC, HP95LX, HP100LX, and HP200LX. *Derive* supports both the 95LX graphics mode and the CGA graphics modes of the 100LX and 200LX computers. Therefore, version 3 can be downloaded and run on all three computers (the 95LX, 100LX, and 200LX) provided it has at least one megabyte (1Mb) of RAM. Also a PCMCIA RAM card is recommended for storing DERIVE program files. Also, Version 3 automatically takes measures to conserve power when running HP palmtops off a battery.

Derive supports common monochrome monitors (MDA and Hercules) and color graphics monitors (CGA, EGA, MCGA, and VGA). Supported printers include HP LaserJet Series II and III, Epson FX80, IBM Proprinter, and compatible printers. Screen images can be saved as TIF (tagged image format) files for further processing in word processing or desktop publishing programs.



```
#1:  SIN(x)

     d
#2:  — SIN(x)
     dx

#3:  COS(x)

     ⌠
#4:  ⌡ COS(x) dx
COMMAND: Author Build Calculus Declare Expand Factor Help Jump soLve Manage
         Options Plot Quit Remove Simplify Transfer Unremove moVe Window approX
Enter option
Int(#3,x)                                 Free:100%            Derive Algebra
```

Figure 1.1: A *Derive* session in progress

1.4 User Interface

Derive is a menu-driven computer algebra system. Therefore, most operations can be carried out using single key-strokes. The interface can be best illustrated by a simple example: see Figure 1.1 for a screen dump of a *Derive* session in progress.

The *Derive* screen consists of three parts: the *working arrea*, the *command menu*, and the *status line*. You can select a menu item by tapping the space bar (or < TAB > key) enough times until the requested is high-lighted followed by pressing the < ENTER > key. A more experienced user will probably select a menu item by immediately entering of the upper case character in the name of the requested menu item. In Figure 1.1 the sine function has been entered by the **Author** command, the derivative has been computed, and the indefinite integral of the cosine function is going to be computed (as the annotation in the status line indicates).

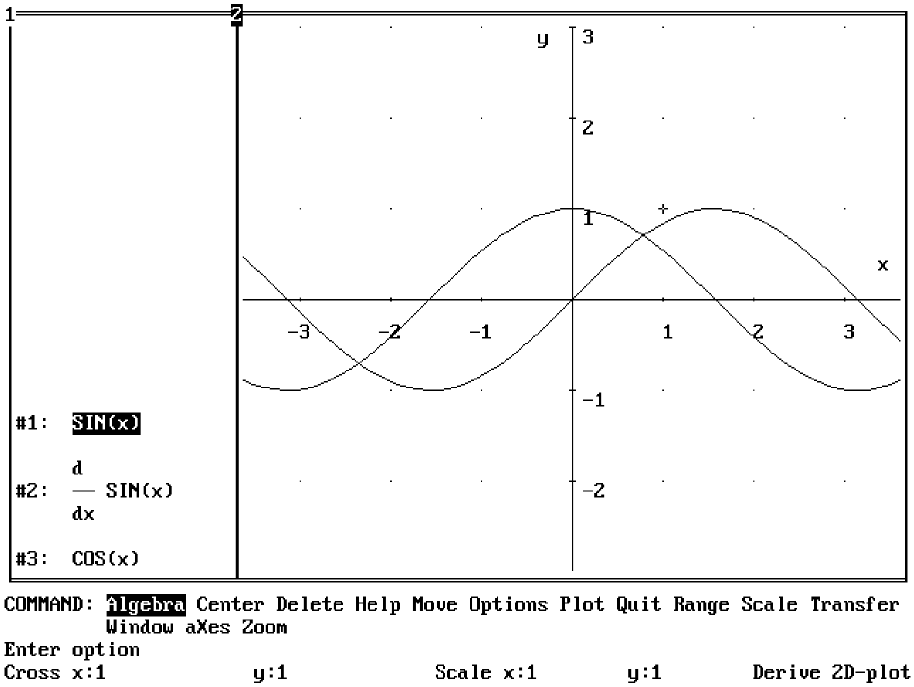
The *Derive* screen may consist of several windows. There are two types of windows: an algebra window and a plot window. An algebra window contains the numbered list of expressions you have created. A plot window contains graphs of functions of one or two variables that have been selected from an algebra window. A typical *Derive* session is illustrated in Figure 1.2.

There is on-line information available via help screens. A typical help screen looks like Figure 1.3.

Derive can also be used in demonstration mode by reading the commands that correspond with menu choices from a file with name-extension `.DMO`. *Derive* will then pause after executing a command A short explanation of the current command can be found in the status line. *Derive* proceeds with the next command in a demo file after a key stroke.

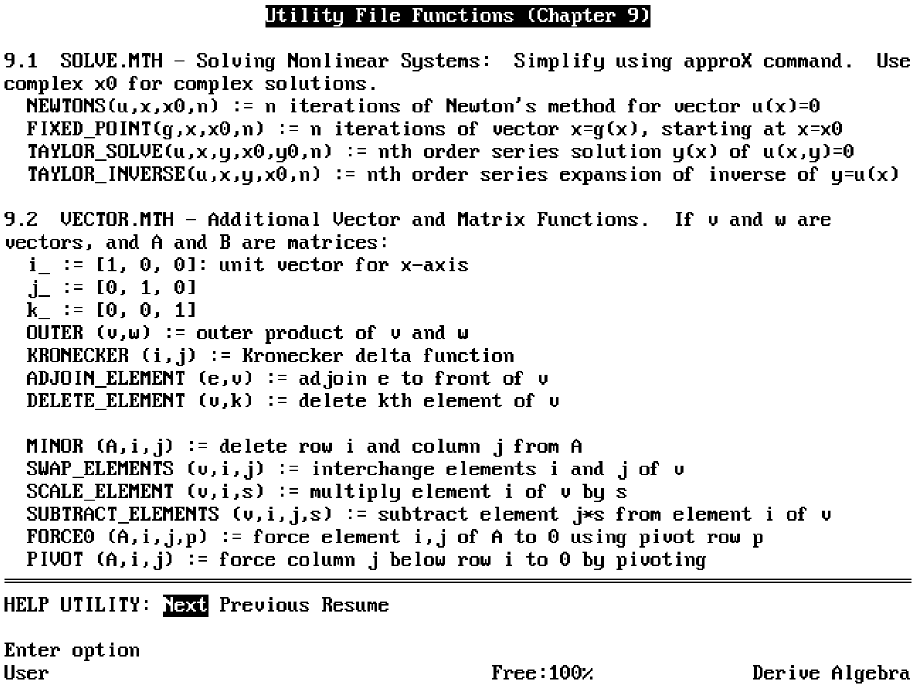
Although *Derive* is a menu-driven system, all commands and declarations of state variables that control precision, notation, domain, and so on, can be made on the **Author** line.

The *Derive* menu and the annotations can be customized. E.g., frequently used menu commands can be moved to the top-level menu, commands can be hidden, and menu command names or annotations can be translated into other languages. In Figure 1.4 you see a Dutch menu that only allows plotting of functions.



```
#1:  SIN(x)

     d
#2:  — SIN(x)
     dx

#3:  COS(x)

COMMAND: Algebra Center Delete Help Move Options Plot Quit Range Scale Transfer
         Window aXes Zoom
Enter option
Cross x:1           y:1           Scale x:1       y:1         Derive 2D-plot
```

Figure 1.2: An algebra and plot window on the screen



```
                  Utility File Functions (Chapter 9)

9.1  SOLVE.MTH - Solving Nonlinear Systems:  Simplify using approX command.  Use
complex x0 for complex solutions.
  NEWTONS(u,x,x0,n) := n iterations of Newton's method for vector u(x)=0
  FIXED_POINT(g,x,x0,n) := n iterations of vector x=g(x), starting at x=x0
  TAYLOR_SOLVE(u,x,y,x0,y0,n) := nth order series solution y(x) of u(x,y)=0
  TAYLOR_INVERSE(u,x,y,x0,n) := nth order series expansion of inverse of y=u(x)

9.2  VECTOR.MTH - Additional Vector and Matrix Functions.  If v and w are
vectors, and A and B are matrices:
  i_ := [1, 0, 0]: unit vector for x-axis
  j_ := [0, 1, 0]
  k_ := [0, 0, 1]
  OUTER (v,w) := outer product of v and w
  KRONECKER (i,j) := Kronecker delta function
  ADJOIN_ELEMENT (e,v) := adjoin e to front of v
  DELETE_ELEMENT (v,k) := delete kth element of v

  MINOR (A,i,j) := delete row i and column j from A
  SWAP_ELEMENTS (v,i,j) := interchange elements i and j of v
  SCALE_ELEMENT (v,i,s) := multiply element i of v by s
  SUBTRACT_ELEMENTS (v,i,j,s) := subtract element j*s from element i of v
  FORCE0 (A,i,j,p) := force element i,j of A to 0 using pivot row p
  PIVOT (A,i,j) := force column j below row i to 0 by pivoting

HELP UTILITY: Next Previous Resume

Enter option
User                                     Free:100%              Derive Algebra
```

Figure 1.3: A help screen

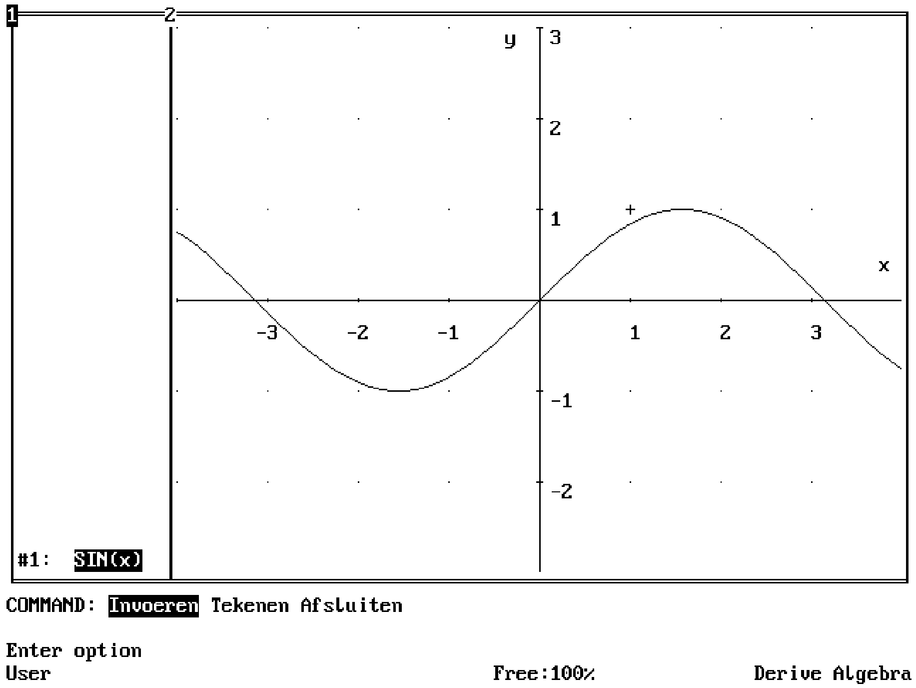


Figure 1.4: A customized Dutch Derive menu

1.5 Software Documentation

The *Derive* User Manual is the main source for information about *Derive*. In this book of 373 pages is explained how to use the software and what is available. Many examples explain the functionality of the software. The *Derive* User Manual has been translated into other languages.

1.6 Licensing Possibilities

A single license copy of *Derive* allows usage of the software on at most one computer at a time or on one computer terminal in a network at a time. Educational Lab licenses, site licenses, and home study licenses are also available.

1.7 Distributors/Dealers

In the following European countries official Distributors/Dealers of Derive are present (December 22, 1994):

Austria

Uni Software Plus GmbH
Hauptstrasse 99
4232 Hafenberg
Phone: +43 7236 3338
Fax: +43 7236 3769

Belgium

QWERTY
Boornsesteenweg 290
2020 Antwerpen
Phone: +32 3 2380993
Fax: +32 3 2308550

Belgium

Bytecom sprl/bvba
Av. de la Seigneurie
1325 Dion-Valmont
Phone: +32 10 223455
Fax: +32 10 241730

Czech Republic

CS1 (Czech Software First sro)
Kounicova 13
60200 Brno
Phone: +42 5 41211762
Fax: +42 5 41211605

Denmark

Orfeus
Skaering Skolevei 202
8250 Ega
Phone: +45 86229955
Fax: +45 86229956

France

EDUSOFT
132 boulevard Camelinat
92247 Malakoff Cedex
+33 1 46730555 (46730563)
+33 1 46730565

France

Softworld
17 Avenue Emile Zola
75015 Paris
Phone: +33 1 40590299
Fax: +33 1 45799555

Germany

BoldBuyer$
Schwertstr. 1-2
71065 Sindelfingen
Phone: +49 7031 811095
Fax: +49 7031 812542

Germany

Cornelsen Software GmbH&Co
Postfach 330109
14171 Berlin
Phone: +49 30 89785526
Fax: +49 30 89785567

Germany

Merlin Software Service GmbH
Klingenthaler Str. 1a
65232 Taunusstein
Phone: +49 6128 84011
Fax: +49 6128 86697

Germany

Uni Software Plus GmbH - Deutschland
Almweg 6
83112 Frasdorf
Phone: +43 7236 333883
Fax: +43 7236 3769

Greece

STEP Computer Shop
Ipirou 5sA
41222 Larissa
Phone: +30 41 532722
Fax: +30 41 532291

Italy

McGraw-Hill Libri Italia srl
Piazza Emilia 5
20129 Milano
Phone: +39 2 719661
Fax: +39 2 733643

Italy

Media Direct
Viale Asiago 85
36061 Bassano del Grappa-VI
Phone: +39 424 504650
Fax: +39 424 504651

Italy

ACS Data Systems OHG
7 Bastioni Maggiori
39042 Bressanone (BZ)
Phone: +39 472 36200
Fax: +39 472 34203

The Netherlands

CAN Expertise Center
Kruislaan 419
1098 VA Amsterdam
Phone: +31 20 5608400
Fax: +31 20 5608448
Email: can@can.nl

Norway

Polysoft a/s
Rosenkrantzgt. 95
3007 Drammen
Phone: +47 32892240
Fax: +47 32895900

Poland

Envitech Ltd
Staffa 13/44
01891 Warszawa

Fax: +48 22 350113

Portugal

GENIUS
Rua Castilho, 211-2
1000 Lisboa
Phone: +351 1 3876525
Fax: +351 1 3881887

Slovenia

ABAKUS d.o.o
Vernek 3
61281 Kresnice
Phone: +38 61 877078
Fax: +38 61 877078

Spain

TR, Technical Research
c/Mallorca, 250 tico
08008 Barcelona
Phone: +34 3 4880606
Fax: +34 3 2160681

Sweden

GrafiTex-Data
Storgatan 11
59040 Kisa
Phone: +46 494 10077
Fax: +46 494 10077

Switzerland

COMSOL AG
Morgenstr. 129
3018 Bern
Phone: +41 31 9984411
Fax: +41 31 9984418

Switzerland

SESCO AG
Nordstr. 15
4665 Oftringen
Phone: +41 6297 4848
Fax: +41 6297 5301

United Kingdom

Cartwell Bratt Ltd.
Old Orchard, Bickley Road
Bromley, Kent BR1 2NE
Phone: +44 81 4671956
Fax: +44 81 4671754

United Kingdom

Oxford Educational Supplies Ltd.
Weston Business Park,
Weston on the Green
Bicester, Oxon OX6 8SH
Phone: +44 869 89369
Fax: +44 869 89654

United Kingdom

System Science
3-5 Cynthia Street
London N1 9JF
Phone: +44 71 8376411
Fax: +44 71 8331022

1.8 Derive Addresses

Soft Warehouse, Inc.
3660 Waialae Avenue, Suite 304
Honolulu, HI 96816-3236
USA
Phone: +1 808 734 5801
Fax: +1 808 735 1105
Email: swh@aloha.com

Soft Warehouse Europe GmbH
Schloss Hagenberg
A-4232 Hagenberg
Austria
Phone: +43 7236 3297
Fax: +43 7236 3769
Email: kutzler@swp.co.at

Derive User Group
Attn. Josef Böhm
D'Lust 1
A-3042 Würmla
Austria
Phone/Fax: +43 2275 8207

International Derive Journal
Paramount Publishing
Campus 400, Mayland Avenue
Hemel Hempstead
Herts. HP2 7EZ, United Kingdom
Phone: +44 442 882 164
Fax: +44 442 882 265

The **Derive Internet Mailing List** is a new electronic discussion list set up after the First International Derive Conference at Plymouth in July 1994. It covers matters relating to the computer algebra package *Derive* and its use in teaching. You can join by sending a message to `mailbase@mailbase.ac.uk` with the single line `join derive-news firstname lastname` as the text of the message. To contribute to the derive mailing list, send email to the address `derive-news@mailbase.ac.uk`. The bulletin is maintained by the CTI Centre for Mathematics and Statistics in Birmingham, UK.

1.9 Long Description

1.9.1 General Description

Derive is a general purpose symbolic calculator which is continually being developed by David Stoutemyer and Albert Rich of Soft Warehouse, Inc. It is the successor of muMATH. A design goal that distinguishes *Derive* from all other computer algebra systems is the impressive effort to make computer algebra possible on simple, cheap, and readily available hardware platforms. This design goal is already met in the sense that the software fits on a diskette and works on almost any personal computer. *Derive* is also the only computer algebra system that is available on the palmtop calculator.

Another important design issue is *Derive*'s ease of use. Most mathematical operations and manipulations can be carried out in the software via its menu-driven user interface. Furthermore, via arrow keys you can easily and quickly select part of an expression to work further with. The *Derive* screen can be split into special purpose parts such as algebra windows and graphics windows. Screen dumps and saving of plots in TIF files enable the user to combine *Derive* with his/her favorite word processor and prepare good-quality mathematical documents. In classroom situations, the demonstration mode of *Derive* is convenient: Step by step one can go through a calculation. The *Derive* menu and the annotations can be customized. E.g., frequently used menu commands can be moved to the top-level menu, menu commands can be hidden, and menu command names or annotations can be translated into other languages.

Derive can handle many of the mathematical computations learned in secondary school, polytechnic, and university mathematics, and it can handle two- and three-dimensional graphing. To mention a few features, *Derive* offers mathematical knowledge of common mathematical functions and their simplification rules, it provides you with calculus tools for differentiation, integration, computing of limits and Taylor series, and analytical solving of ordinary differential equations, and it allows you to do all basic arithmetic of vectors and matrices (addition and multiplication, inner and outer product, determinant, transpose, inverse, eigenvalues and eigenvectors, and so on). With *Derive* you can graph functions in one or two unknowns in various coordinate systems. Many options like scales, center, viewpoint, grid, and color can be changed via the menu. The 3rd party AcroSpin software allows you to rotate, translate, and scale 3-dimensional wire frames and point clouds on the fly.

Derive also incorporates some functional programming facilities that allow the user to define mathematical procedures. In the *Derive* language you have iteration, the if-then-else control structure, logical and relational operators, and recursion at your disposal. All of *Derive*'s own utility functions are written in the *Derive* language. Their use is well-documented in the Derive User Manual, which is translated into other languages.

1.9.2 Features

Features of *Derive* include:

Arithmetic

Exact arithmetic to many digits;
Approximate arithmetic to user-defined precision;
Mixed exact/approximate arithmetic;
Integer factoring, greatest common divisors, and prime numbers;
Complex and infinite arithmetic.

Algebra

Simplification formulae, expansion or factorization of polynomials;
Declaration of variable domains and values;
Numeric solving of equations at desired precision;

Algebraic solving of equations and inequalities;
Solving of systems of linear equations.

Graphics

2D Plotting:
  Plotting in rectangular or polar coordinates;
  Parametric plotting and space curves;
  Implicit Plots;
  Locating and tracing points by movable cross;
  Optional automatic vertical scaling;
  Choice of center of plot;
  Easy-to-handle plot range specification;
  Zooming in or out.

3D Plotting:
  Wire frame plots with hidden line removal;
  Choice of viewpoint and plot scale;
  Optional automatic vertical centering and scaling.

Calculus

Symbolic limits, derivatives, and integrals;
Arc lengths, area, volume, and related functions;
Taylor and Fourier series;
Laplace transforms;
Exact solving of ordinary differential equations;
Runge-Kutta method for systems of ODE's.

Matrix and Vector Calculus

Transpose, determinant, trace, and inverse of matrices;
Eigenvalues and eigenvectors;
Dot, cross and outer products;
Differential and integral vector calculus.

Functions

Exponential, logarithmic, trigonometric, and hyperbolic functions;
Complex-valued functions;
Probability, statistics, financial, and annuity functions;
Bessel, hypergeometric, Chi-square, Zeta, and other special functions;
Pseudo-random number generator.

Derive Language

If-then-else conditional programming;
Logical and relational operators;
Recursive programming;
Iteration and recurrence relations;
Selection in vectors.

Menu-Driven Interface

Split and/or overlaid algebra and plot windows;
2D mathematical display of formulae;
Highlighting and extraction of subexpressions;
Demo facilities;
CGA, EGA, VGA, MCGA, and Hercules graphics;
Custamizable menu;
Simple annotation of expressions.

Input/Output

Saving and loading of expressions in files;
Multi-file loading;
(Color) printing of plots, expressions, and screens;
Saving of graphical images as TIF files;
Loading of numeric data;
Code generation in C, Fortran, Pascal, and Basic.

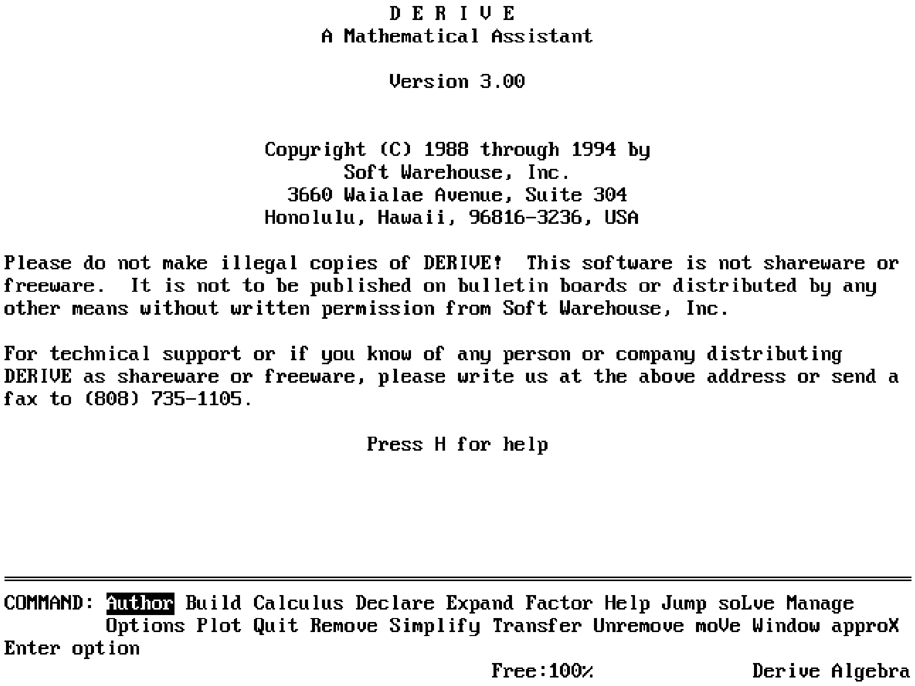
Chapter 2

A Tour of Derive

In this chapter we show most features of *Derive* in the form of a pseudo *Derive* session interleaved with explanatory remarks.

2.1 Launching Derive

To start *Derive* enter DERIVE from the DOS shell. The system comes up with the initial screen (see Figure 2.1).



```
                              D E R I V E
                      A Mathematical Assistant

                            Version 3.00


                  Copyright (C) 1988 through 1994 by
                         Soft Warehouse, Inc.
                    3660 Waialae Avenue, Suite 304
                  Honolulu, Hawaii, 96816-3236, USA

Please do not make illegal copies of DERIVE!  This software is not shareware or
freeware.  It is not to be published on bulletin boards or distributed by any
other means without written permission from Soft Warehouse, Inc.

For technical support or if you know of any person or company distributing
DERIVE as shareware or freeware, please write us at the above address or send a
fax to (808) 735-1105.

                            Press H for help




COMMAND: Author Build Calculus Declare Expand Factor Help Jump soLve Manage
         Options Plot Quit Remove Simplify Transfer Unremove moVe Window approX
Enter option
                                          Free:100%           Derive Algebra
```

Figure 2.1: The initial *Derive* screen

Like any other *Derive* screen it consists of three parts: the working area, the command menu, and the status line. When you select the menu item **Author** by tapping the space bar (or < TAB > key) enough times until **Author** is high-lighted followed by pressing the < ENTER > key, or when you immediately select the menu item **Author** by pressing the **A** key, you will be prompted to enter an expression. You enter for example

```
sin(x)
```

and *Derive* responds in the working area with

```
1:    SIN(x)
```

when *Derive* is in case-insensitive word input mode (use the **Otion Input** menu to change the settings if necessary).

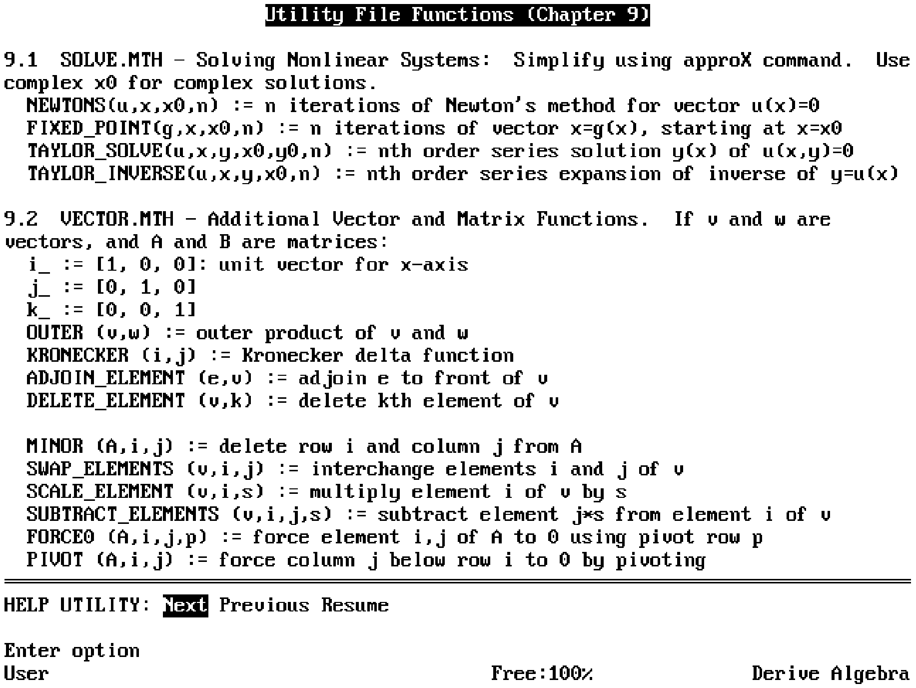
The computer algebra system *Derive* is a symbolic calculator. So, it can for example compute the derivative of the expression. For this, you issue the **Calculus Differentiate** command, and choose to compute the first derivative with respect to x. The system responds with

```
      d
2:    -- SIN(x)
      dx
```

This is what you wanted, so you can select **Simplify** from the menu to activate the "real computation" and get from *Derive*

```
3:    COS(x)
```

There is on-line information available via help screens. A typical help screen looks like Figure 2.2. You can interrupt a lengthy computation in *Derive* or cancel your menu choice by pressing the <Esc> key.



```
                    Utility File Functions (Chapter 9)

9.1  SOLVE.MTH - Solving Nonlinear Systems:  Simplify using approX command.  Use
complex x0 for complex solutions.
  NEWTONS(u,x,x0,n) := n iterations of Newton's method for vector u(x)=0
  FIXED_POINT(g,x,x0,n) := n iterations of vector x=g(x), starting at x=x0
  TAYLOR_SOLVE(u,x,y,x0,y0,n) := nth order series solution y(x) of u(x,y)=0
  TAYLOR_INVERSE(u,x,y,x0,n) := nth order series expansion of inverse of y=u(x)

9.2  VECTOR.MTH - Additional Vector and Matrix Functions.  If v and w are
vectors, and A and B are matrices:
  i_ := [1, 0, 0]: unit vector for x-axis
  j_ := [0, 1, 0]
  k_ := [0, 0, 1]
  OUTER (v,w) := outer product of v and w
  KRONECKER (i,j) := Kronecker delta function
  ADJOIN_ELEMENT (e,v) := adjoin e to front of v
  DELETE_ELEMENT (v,k) := delete kth element of v

  MINOR (A,i,j) := delete row i and column j from A
  SWAP_ELEMENTS (v,i,j) := interchange elements i and j of v
  SCALE_ELEMENT (v,i,s) := multiply element i of v by s
  SUBTRACT_ELEMENTS (v,i,j,s) := subtract element j*s from element i of v
  FORCE0 (A,i,j,p) := force element i,j of A to 0 using pivot row p
  PIVOT (A,i,j) := force column j below row i to 0 by pivoting
================================================================================
HELP UTILITY: Next Previous Resume

Enter option
User                                     Free:100%              Derive Algebra
```

Figure 2.2: A help screen

Now everything is ready for our tour of *Derive*. Throughout the tour we shall use some conventions: typewriter style words like **soLve** refer to a menu command in *Derive*. The boldface character is the capital character that can be typed to trigger the menu command. We shall refer to a sequence of commands entered from successive menus by concatenating the menu option names, like e.g. in the above phrase "issue the **Declare Variable** command". We shall often use *Derive* screen dumps to show results. Note that this also means that the commands are not displayed.

2.2 Arithmetic

Derive provides three modes of computing with numbers: exact, arbitrary precision, and mixed arithmetic. Let us first look at exact arithmetic. First, issue the **Option Precision** command and make sure that the `Exact` mode is chosen. Next, **Author** `3^5` to get the number

```
       5
4:    3
```

Derive only displays the number in exponential format, but to get it in ordinary number format you must **Simplify** it further

```
5:    243
```

Unlike a numerical calculator, *Derive* can compute exactly with large numbers. For example, you can enter 333^{555} and simplify it.

```
         555
6:    333
```

```
7:    904793758405842302339751965847805693334010290411675875418030355725136884.........
```

On the screen, the number is displayed in one line; to see more of the output, repeatedly press the < Ctrl > and right arrow key. Although the number is too large to fit on the screen, *Derive* has internally stored the number in exact mode. So, you can process it further and, for example, **Factor** it.

```
       1110   555
8:    3    37
```

You can compute approximate results. For example, the previous result can be approximated by the **approX** command. After a few seconds, you get

```
                1399
9:    9.04793 10
```

In *Derive*, you can compute at any desired precision. Let us compute the numeric value of $\sqrt{5+2\sqrt{6}}$ in forty digits. On dumb terminals this number is displayed as

```
10:  SQRT(5 + 2 SQRT(6))
```

On PC compatible computers you can also define the square root by pressing < Alt > Q and it is displayed as $\surd$. In exact mode the square roots can be denested (issue the **Simplify** command).

```
10:   SQRT(2) + SQRT(3)
```

Now, set the precision to forty digits by selecting **Option Precision** `Digits 40`.

```
11:   PrecisionDigits := 40
```

By the way, you get to the **Digits** field by pressing the < Tab > key. Henceforth, numerical computations are done in a precision of forty digits; then number $\sqrt{2}+\sqrt{3}$ is approximated to

```
12:   3.146264369941972342329135065715570445512
```

In approximate mode, *Derive* simplifies a number to the simplest rational number that approximates the original number accurate to the current precision and converts it in scientific notation. *Derive* also provides a mixed mode of arithmetic in which irrational number are approximated but rational numbers are not. Note the difference in the next two approximations of $\sqrt{1-8/9}$; the first result is obtained in approximate mode, the second in mixed mode.

```
          /     8 \
13:   SQRT|1 - ---|
          \     9 /

14:   0.3333333333333333333333333333333333333333

       1
15:   ---
       3
```

Derive can also handle complex numbers (the complex number is denoted by $\sharp i$ and can be entered on PC compatible computers by pressing <ALT>i).

```
           1
17:   ----------
       2 + 3 #i

        2     3 #i
18:   ---- - ------
       13      13

      /  /     1    \    /     1    \\
19:   |RE|----------|, IM|----------||
      \  \ 2 + 3 #i /    \ 2 + 3 #i //

      /  2       3 \
20:   |----, - ----|
      \ 13      13 /
```

Derive allows easy access to different branches of complex multi-valued functions. For example, the principle branch choice of $(-8)^{\frac{1}{3}}$ is $1+\sqrt{3}i$, but selecting the **Real** branch option under the **Manage Branch** menu command gives the result -2.

```
21: Branch := Real

        1/3
22: (-8)

23: -2

24: Branch := Principal

        1/3
25: (-8)

26: 1 + SQRT(3) #i
```

Derive “knows” about standard mathematical functions.

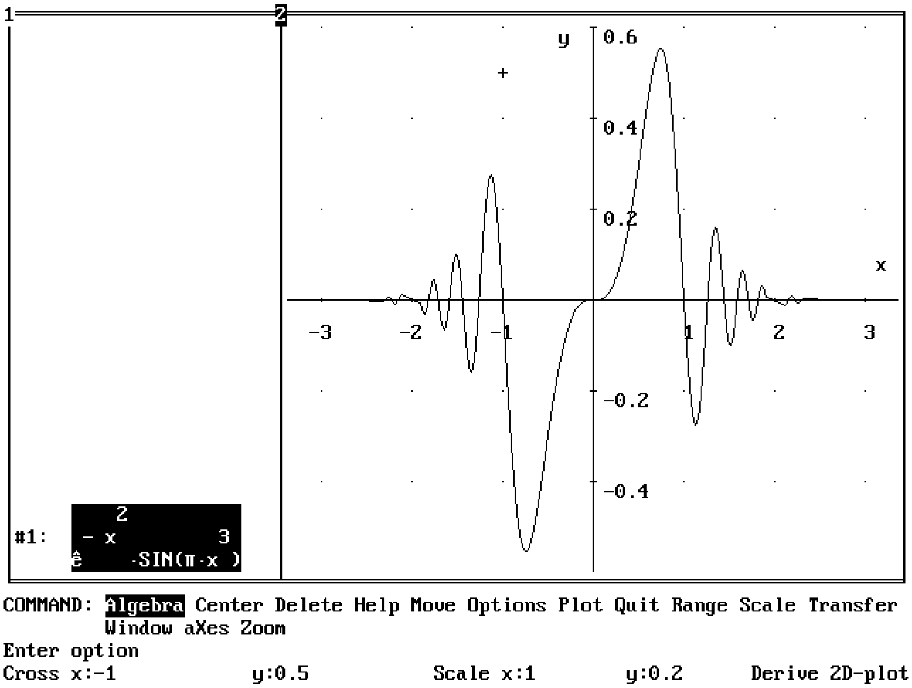
```
27:   SIN(pi/6)

       1
28:   ---
       2
```

```
29:   ACOS(1/2)

       pi
30:   ----
        3
```

The number π can be entered in *Derive* as `pi` or by pressing <ALT>p.

2.3 Graphics

You can plot the graph of a function, say of $e^{-x^2}\sin(\pi x^3)$, over some interval. You select the formula that you want to plot and choose the menu item **`Plot`**. You are asked whether you want the graph beside, below, or on top of the algebra window. In Figure 2.3 we have chosen to split vertically the working area into an algebra window and a plot window. You can switch between the split windows via the `F1` function key. In the plot window we have rescaled the vertical axis and zoomed in to make the picture as large as possible on the screen. The cross is at position $(-1, 0.5)$ as can be seen from the status line.



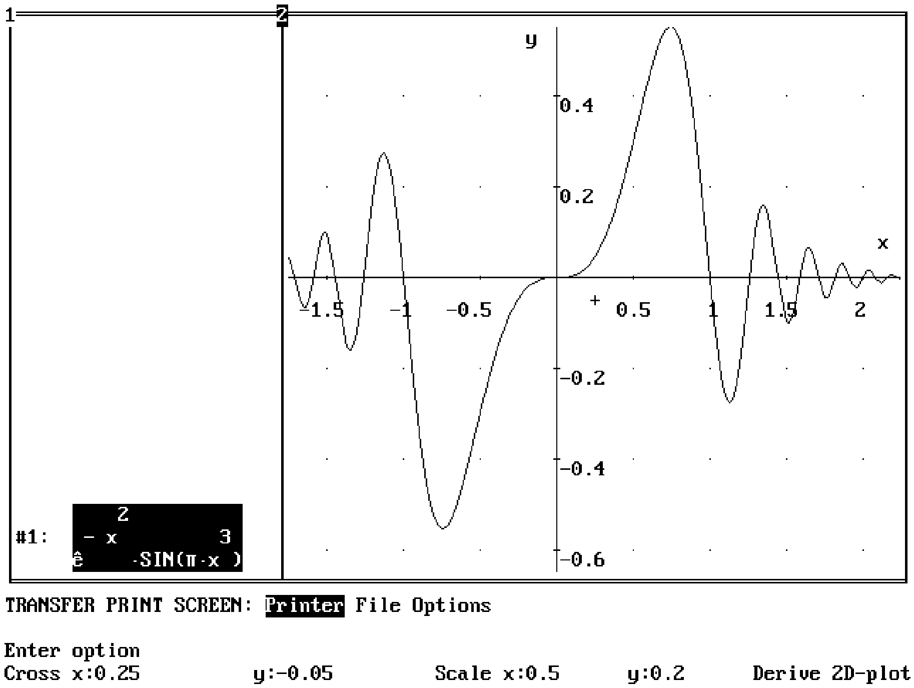
```
           2
#1:     - x          3
       ê      ·SIN(π·x )
```

```
Enter option
Cross x:-1          y:0.5          Scale x:1       y:0.2      Derive 2D-plot
```

Figure 2.3: 2D Plot

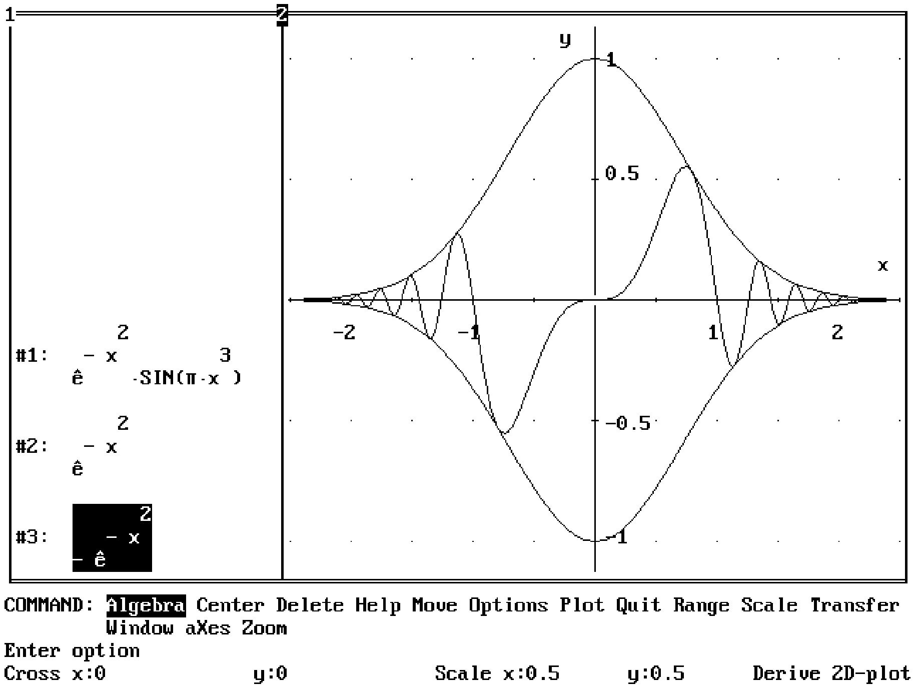
You can change many of the options that determine how the graph actually looks like. In Figure 2.4 we have centered the plot around the point $(0.25, -0.05)$ via the menu item **`Center`**, we have zoomed in along the vertical axis, and have changed the horizontal scaling via the **Range** command.

More than one graph in a plot window is no difficulty: you simply select a formula that you want to plot, too, and plot it in the usual way. It will appear in a new color on the screen. In Figure 2.5 the graphs of e^{-x^2} and e^{-x^2} are also present.



```
#1:       2
       - x        3
      ê     ·SIN(π·x )

TRANSFER PRINT SCREEN: Printer File Options

Enter option
Cross x:0.25          y:-0.05         Scale x:0.5      y:0.2       Derive 2D-plot
```

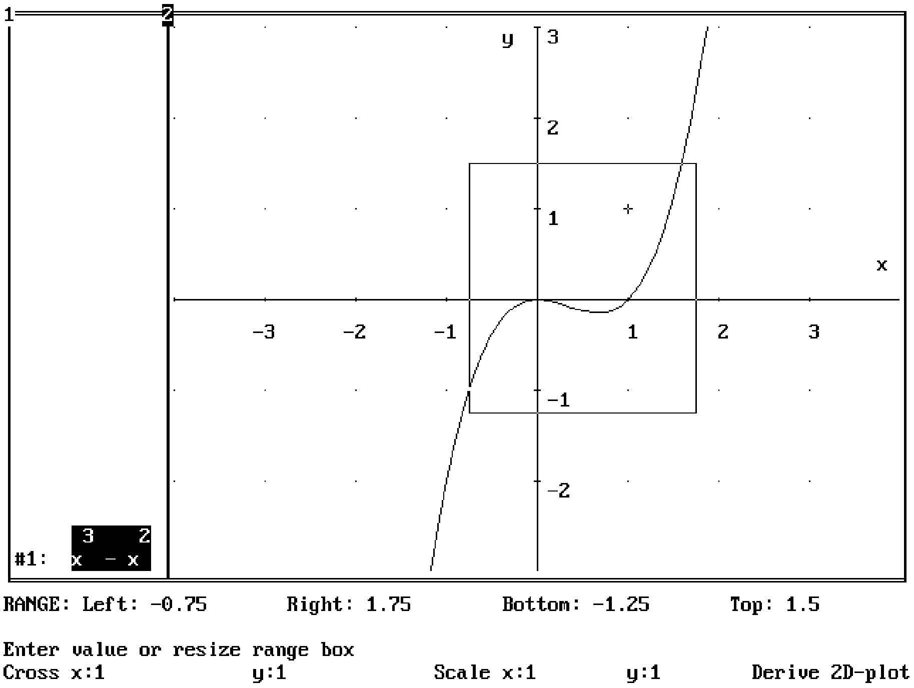
Figure 2.4: 2D Plot with center (0.25,-0.05)



```
          2
#1:    - x         3
      ê     ·SIN(π·x )

          2
#2:    - x
      ê

             2
#3:       - x
      - ê

COMMAND: Algebra Center Delete Help Move Options Plot Quit Range Scale Transfer
         Window aXes Zoom
Enter option
Cross x:0             y:0             Scale x:0.5      y:0.5       Derive 2D-plot
```

Figure 2.5: Three graphs in one 2D plot

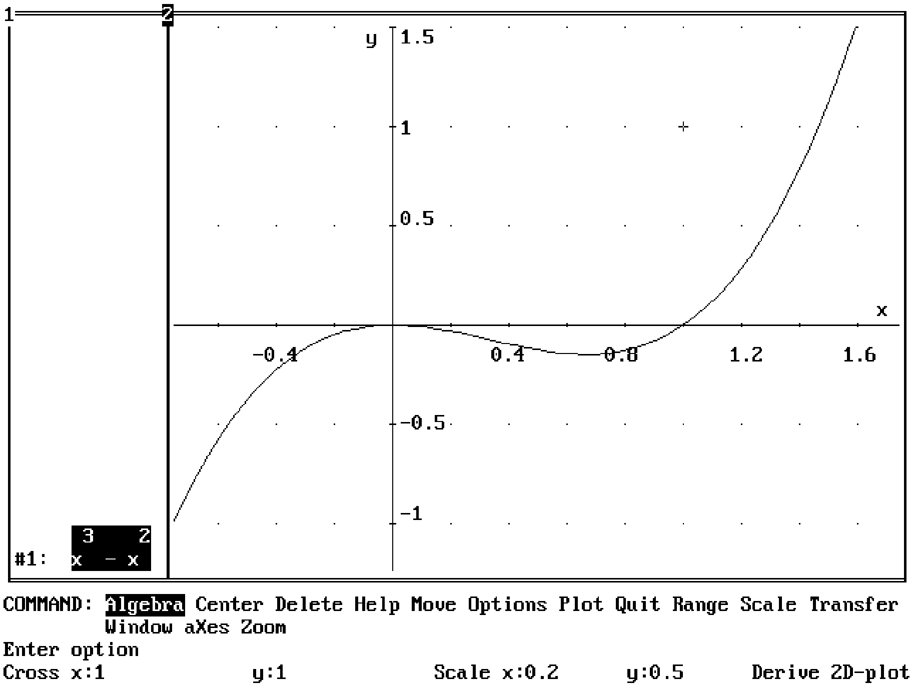
The **Range** command allows you to interactively specify the plot range of 2D plot windows; by the arrow keas you can size and position a rectangular zoom box as shown in Figure 2.6 below.



```
RANGE: Left: -0.75         Right: 1.75         Bottom: -1.25         Top: 1.5

Enter value or resize range box
Cross x:1              y:1              Scale x:1        y:1        Derive 2D-plot
```

Figure 2.6: 2D Plot with zoom box

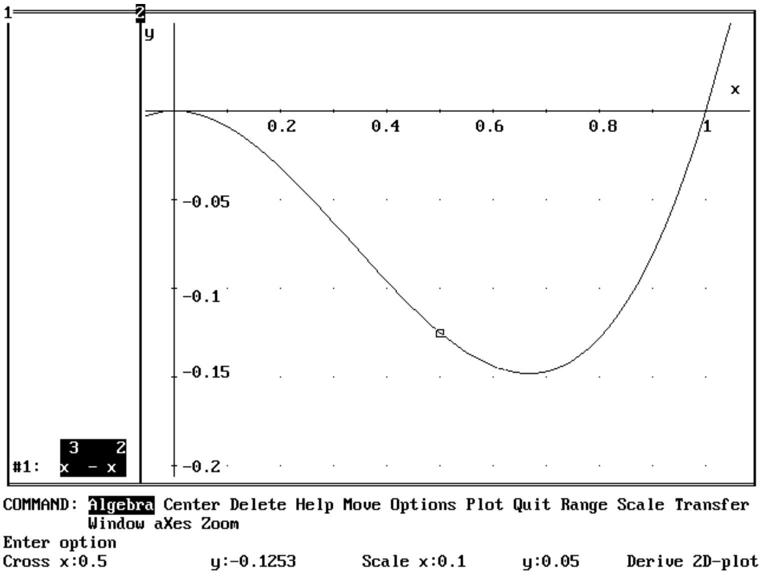
The selection is replotted in Figure 2.7.



```
Enter option
Cross x:1              y:1              Scale x:0.2      y:0.5      Derive 2D-plot
```

Figure 2.7: 2D Plot after zooming in

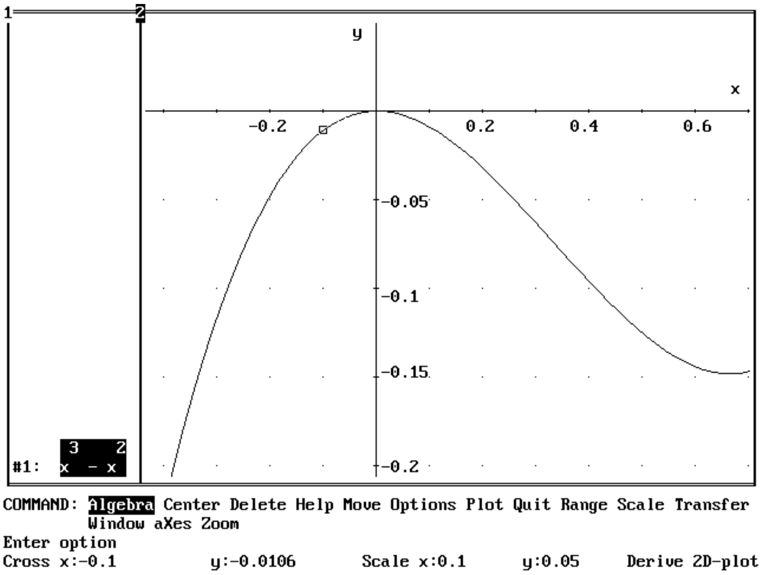
In *Derive* 3 you can trigger `trace mode` by the `F3` function key so that the cross is confined to moving along a plot line. The cross displays in trace mode as a small square box whose center point lines on the plot line (see Figure 2.8).



```
COMMAND: Algebra Center Delete Help Move Options Plot Quit Range Scale Transfer
         Window aXes Zoom
Enter option
Cross x:0.5           y:-0.1253       Scale x:0.1       y:0.05      Derive 2D-plot
```

Figure 2.8: 2D Plot in trace mode

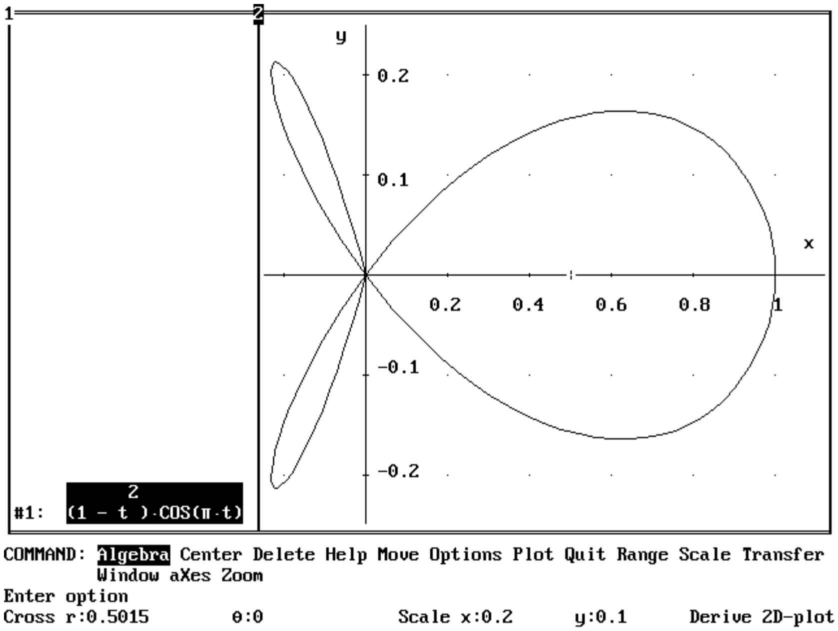
When you issue in the plot 2D window **Options State** command and select the `follow cross mode`, then the plot range is automatically adjusted whenever the cross is moved out of the current plot range. In Figure 2.9 you see the effect of moving the cross in Figure 2.8 to the left out of the plot range: the plot range is adjusted so the cross remains visible.



```
COMMAND: Algebra Center Delete Help Move Options Plot Quit Range Scale Transfer
         Window aXes Zoom
Enter option
Cross x:-0.1          y:-0.0106       Scale x:0.1       y:0.05      Derive 2D-plot
```

Figure 2.9: 2D Plot in trace mode

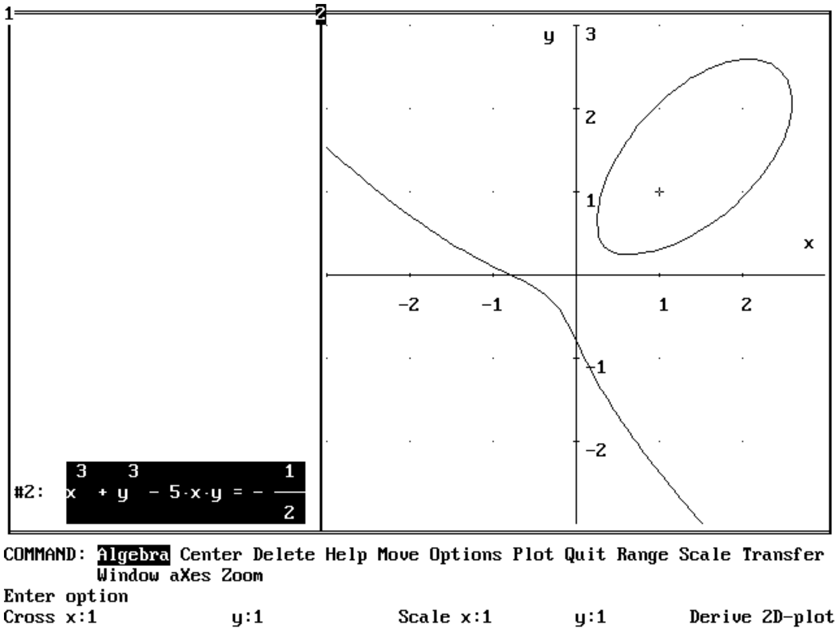
Special two-dimensional graphics is provided, too. For example, you can plot a parametric curve in polar coordinates (see Figure 2.10). All you have to do is to set **Options State** `Coordinates: Polar`.



```
#1:  (1 - t^2)·COS(π·t)
COMMAND: Algebra Center Delete Help Move Options Plot Quit Range Scale Transfer
         Window aXes Zoom
Enter option
Cross r:0.5015        θ:0              Scale x:0.2      y:0.1       Derive 2D-plot
```

Figure 2.10: 2D parametric curve plot

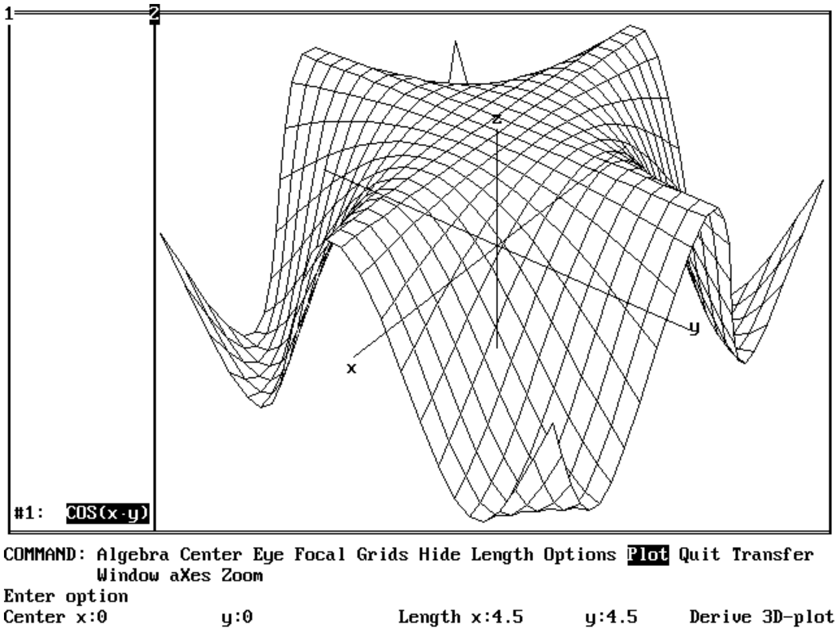
Or you make an implicit plot as shown in Figure 2.11 below.



```
#2:  x^3 + y^3 - 5·x·y = - 1/2
COMMAND: Algebra Center Delete Help Move Options Plot Quit Range Scale Transfer
         Window aXes Zoom
Enter option
Cross x:1             y:1              Scale x:1        y:1         Derive 2D-plot
```

Figure 2.11: implicit plot

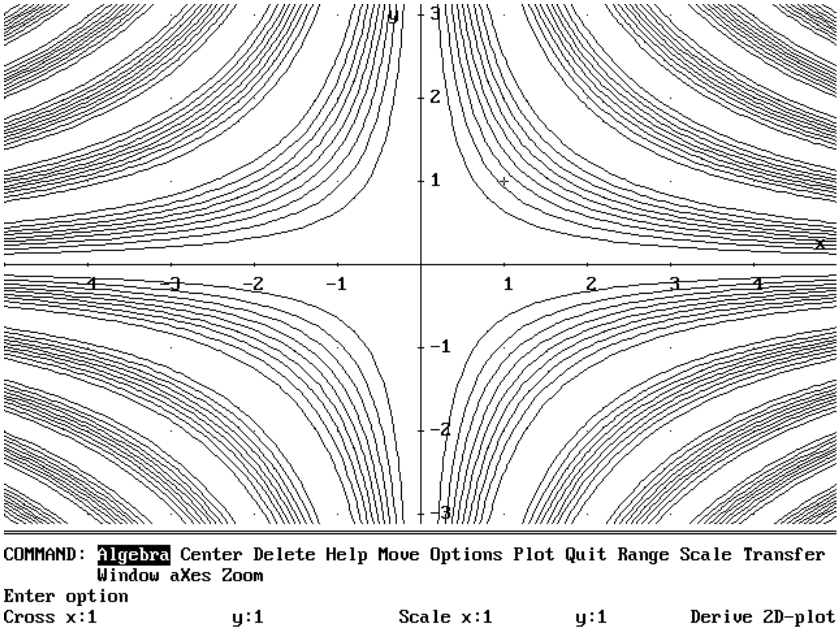
A three-dimensional surface plot is shown in Figure 2.12. Again, many options can and have been changed interactively. The Figure 2.12 was drawn with a 25×25 grid, with a higher eye point than usual.



```
Enter option
Center x:0           y:0                Length x:4.5       y:4.5      Derive 3D-plot
```

Figure 2.12: 3D surface plot

Special 3D graphics is provided, too. For example, Figure 2.13 shows a contour plot of the function $z = \cos(xy)$ as z varies from -1 to 1 in steps of 0.2



```
Enter option
Cross x:1            y:1                Scale x:1        y:1          Derive 2D-plot
```

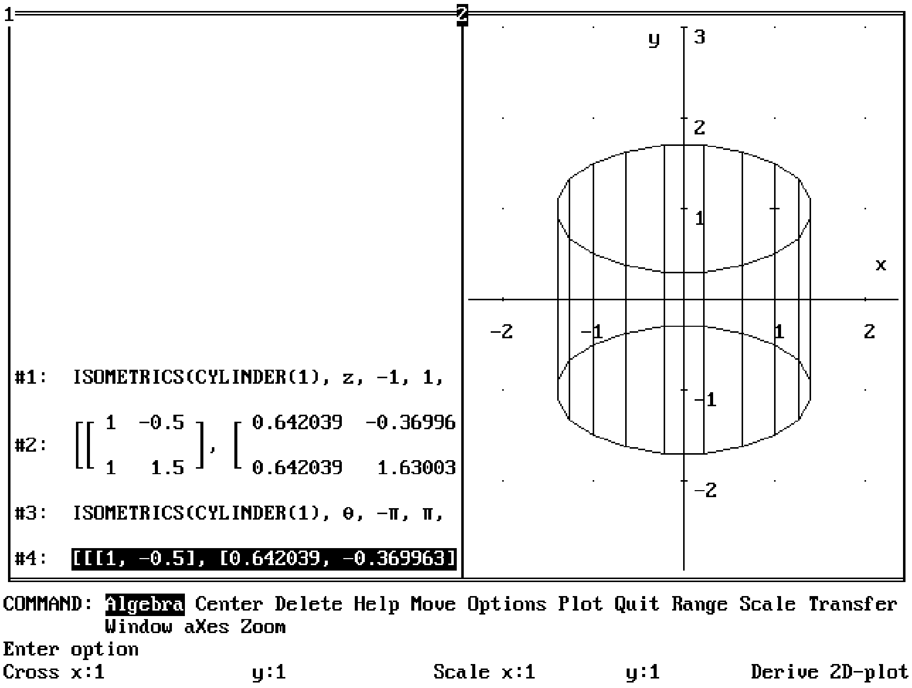
Figure 2.13: contour plot

In the algebra window you enter

```
1: VECTOR( z = COS(x y), z, -1, 1, 0.2 )
```

Simplify and **P**lot the expression.

Figure 2.14 shows a cylinder obtained by the following sequence of commands.



```
#1:   ISOMETRICS(CYLINDER(1), z, -1, 1,

        ┌┌ 1  -0.5 ┐  ┌ 0.642039  -0.36996
#2:     ││           │, │
        └└ 1   1.5 ┘  └ 0.642039   1.63003

#3:   ISOMETRICS(CYLINDER(1), θ, -π, π,

#4:   [[[1, -0.5], [0.642039, -0.369963]

COMMAND: Algebra Center Delete Help Move Options Plot Quit Range Scale Transfer
         Window aXes Zoom
Enter option
Cross x:1            y:1            Scale x:1       y:1       Derive 2D-plot
```

Figure 2.14: 3D Cylinder plot

First, you have to issue the **T**ransfer **L**oad **U**tility command and load the utility file GRAPHICS.MTH. In the algebra window you enter

```
1:   ISOMETRICS( CYLINDER(1), z, -1, 1, 1, theta, -pi, pi, 20 )
```

Issue the appro**X** command and **P**lot the data structure. You can plot it in one color by selecting **O**ptions **C**olor **A**uto `Change: no`; you must also choose **O**ptions **S**tate `Coordinates: Rectangular`, `Mode: Connected`, and `Size: Small` to connect the sample points.

```
     // 1  -0.5 \  / 0.642039  -0.369963 \  / 0.221231  -0.301598 \
2:   ||         |, |                     |, |                     |, ...
     \\ 1   1.5 /  \ 0.642039   1.63003  /  \ 0.221231   1.69840  /
```

At this stage, only vertical lines are shown. By interchange of arguments, you can also connect the points horizontally and get a “real” cylinder.

```
3:   ISOMETRICS( CYLINDER(1), theta, -pi, pi, 20, z, -1, 1, 1 )

4:   [[[1, -0.5], [0.642039, -0.369963], .....
```

In the output we have omitted part of the expressions.

2.4 Algebraic Facilities

2.4.1 Calculus

Differentiation

Though calculus is the study of real functions with infinitesimal tools like differentiation, integration and series expansion, many of its computations are in fact of algebraic nature. For instance, differentiation can be implemented as a set of transformation rules applied to a formula consisting of elementary functions like powers, trigonometric functions, exponential mapping, logarithm, etc. One of the transformation rule that can be realized in *Derive* is

$$\frac{d}{dx}x^n = nx^{n-1}.$$

First you **`Author`** `x^n` to get the formula

```
       n
1:    x
```

Next you issue the **`Calculus Differentiate`** command and you choose to compute the first derivative with respect to x. The system responds with

```
      d   n
2:    -- x
      dx
```

This is what you wanted, so you can **`Simplify`** the expression.

```
         n - 1
3:    n x
```

You can issue successive **`Calculus Differentiate`** commands to obtain mixed partial derivatives. For example, take the expression

```
4:    x TAN (y) + TAN (x)
```

Note that you may also use in input mode a space instead of the * symbol for multiplication. The mixed second-order partial derivative of $x\tan y + \tan x$ with respect to x and y displays as

```
      d  d
5:    -- -- (x TAN (y) + TAN (x))
      dy dx
```

It simplifies to

```
             2
6:    TAN (y)  + 1
```

By the **`Factor`** command it is transformed into

```
          1
7:    --------
             2
      COS (y)
```

Function Definition

Defining a function. e.g.,

$$f : x \mapsto \sqrt{x^2 + \frac{1}{y}}$$

can be done in *Derive* in two ways. You can issue the **Declare Function** command and enter its name and definition (by `SQRT(x^2+1/y)`). *Derive* displays it on dumb screens as

```
                      / 2    1 \
8:   F (x, y) := SQRT |x  + ---|
                      \      y /
```

Alternatively, you **Author** this line from scratch. On PC compatible computers you can also define the square root by pressing <ALT>`Q` and it is displayed as √ . Now, `F (u, v)` simplifies to

```
          /  2       \
          | u  v + 1 |
9:   SQRT |----------|
          \     v    /
```

`F (3, 4)` simplifies to the exact number

```
      SQRT (37)
10:  -----------
          2
```

The appro**X**imate command does credit to its name and yields the corresponding numerical value. Before this you can issue the **Options Precision** command to set the number of digits. E.g., $\frac{\sqrt{37}}{2}$ approximates in 20 digits to

```
11:  3.0413812651491098444
```

Integration

Use the **Calculus Integrate** command to compute antiderivatives and definite integrals. After you select the expression, *Derive* prompts you to enter the name of the variable, the lower limit and the upper limit. When you do not specify values for the limits *Derive* will compute the antiderivative. *Derive* knows all classical integration tricks. For example, it computes integral

$$\int \frac{1}{x^4\sqrt{1+x^2}}\,dx$$

and comes back with the answer

```
             2          2
12:   SQRT (x  + 1) (2 x  - 1)
     ---------------------------
                    3
                 3 x
```

Note that *Derive* follows the style of integral tables and leaves out the integration constant. In contrast with integral tables, *Derive* offers you the possibility to check the answer by differentiation (check the example!).

Derive is quite powerful in computing definite integrals. The following definite integral can only be computed without any effort with *Derive* and not in other computer algebra systems such as Mathematica 2.3, Maple V Release 3, Macsyma 419, or Axiom 2. It is entered as

```
INT( ASIN(1/SQRT(4-x^2)), x, 0, 1 ) =
```

and *Derive* displays the result as

```
      1
     /      /       1      \       pi          / 1 \
     |  ASIN|--------------| dx = ---- - 3*ASIN|---|
13:  |      |           2  |        2          \ 3 /
    /      \ SQRT(4 - x ) /
      0
```

This is a general scheme: if an expression is entered on the author line followed by an = sign, then *Derive* displays an equation whose left side is the unsimplified form of the expression and whose right side is the simplified form of the expression.

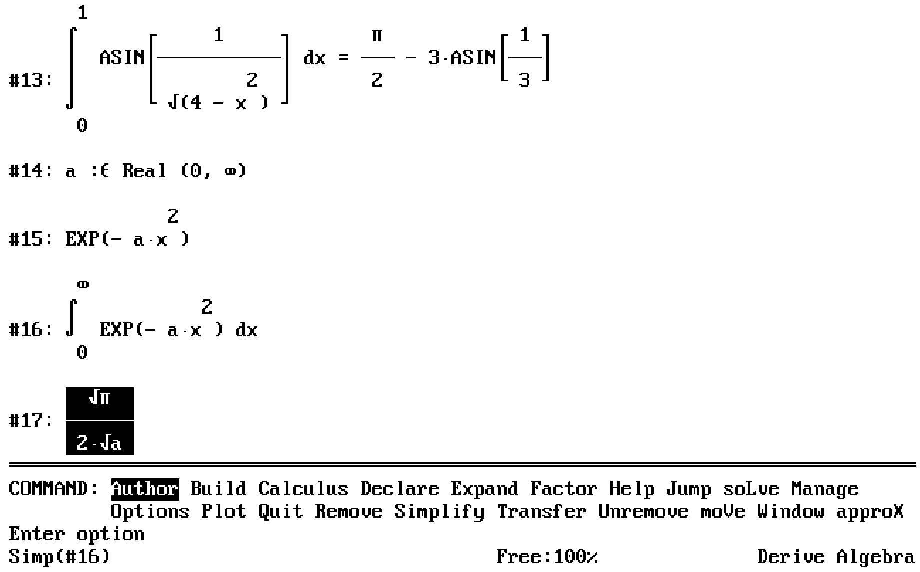
Sometimes you have to specify the range of a variable which occurs in the integrand. For example,

$$\int_0^\infty e^{-ax^2}\,dx$$

can only be computed by *Derive* if you first issue the **`Declare Variable`** command to specify that a is a positive real number. The corresponding *Derive* session looks on a dumb screen like

```
14:   a :epsilon Real (0, inf)


                2
15:   EXP (- a x )


       inf
      /              2
16:   |    EXP (- a x ) dx
      /
       0

       SQRT (pi)
17:   ------------
       2 SQRT (a)
```

On most screens it looks however much better (see Figure 2.15).



```
        1
      ⌠      ┌     1      ┐        π
#13:  │ ASIN │ ────────── │ dx = ─── - 3·ASIN┌1┐
      ⌡      │         2  │        2         └3┘
             └ √(4 - x )  ┘
        0

#14: a :∈ Real (0, ∞)

                  2
#15: EXP(- a·x )

        ∞
      ⌠              2
#16:  ⌡  EXP(- a·x ) dx
        0

      √π
#17: ────
     2·√a
COMMAND: Author Build Calculus Declare Expand Factor Help Jump soLve Manage
         Options Plot Quit Remove Simplify Transfer Unremove moVe Window approX
Enter option
Simp(#16)                              Free:100%            Derive Algebra
```

Figure 2.15: assumptions

When a closed-form of an integral does not exist you can always try to find a numerical approximation by the `approXimate` command.

```
        1
       /
18:    |  SIN (SIN (x)) dx
       /
        0

19:    0.430606
```

Taylor Series

Use the **`Calculus Taylor`** command to compute the Taylor series approximation of an expression around some expansion point and up to some order. For example, the series expansion of $\frac{1}{2-x}$ around 0 up to order 8 is found by *Derive* as

```
              /   1           \
19:    TAYLOR |-------, x, 0, 8|
              \ 2 - x         /

         8       7       6      5      4      3     2
        x       x       x      x      x      x     x     x     1
20:    ----- + ----- + ----- + ---- + ---- + ---- + --- + --- + ---
        512     256     128     64     32     16     8     4     2
```

Limits

Use the **Calculus Limits** command to compute limits or use the `LIM` command in the author line followed by the = sign to show the limit and the result in equational format.

```
                         2
             1 - COS (x)     1
21:   lim  --------------  = ---
      x->0   x SIN (2 x)     2
```

Derive can also compute directional limits.

```
                   SQRT (p)
22:   lim  (1 + p)
      p->0-

23:   1
```

2.4.2 Solving Systems of Equations

Derive can solve systems of *linear* equations by the `soLve` command.

```
24:   [x + a y = 1, a x - y = -1]

      /      1 - a         a + 1 \
25:   |x = --------, y = --------|
      |      2             2     |
      \     a  + 1        a  + 1 /
```

Systems of *nonlinear* equations cannot be solved automatically by Derive. For example, consider the following system of polynomial equations

```
       2    2
26:   x  + y  = 1

27:   x y = 1
```

With the `soLve` command you can rewrite the second equation as

```
           1
28:   x = ---
           y
```

Issue the **Manage** `Substitute` command to substitute this expression for `x` into equation `26`.

```
      / 1 \2    2
29:   |---|  + y  = 1
      \ y /
```

and solve it for `y`. This is in this example basically the same as solving a univariate polynomial equation of fourth degree, which can be done analytically by the `soLve` command.

```
           SQRT(3)     #i
30:   y = --------- + ----
             2          2
```

```
           SQRT(3)     #i
31:   y = --------- - ----
             2         2

             SQRT(3)     #i
32:   y = - --------- + ----
               2         2

             SQRT(3)     #i
33:   y = - --------- - ----
               2         2
```

Four complex solutions. The reciprocal values are the corresponding `x` values. For example, the first `y` value has

```
               1
      ------------------
34:     SQRT(3)     #i
       --------- + ----
          2         2

       SQRT(3)     #i
35:   --------- - ----
         2         2
```

as corresponding `x` values.

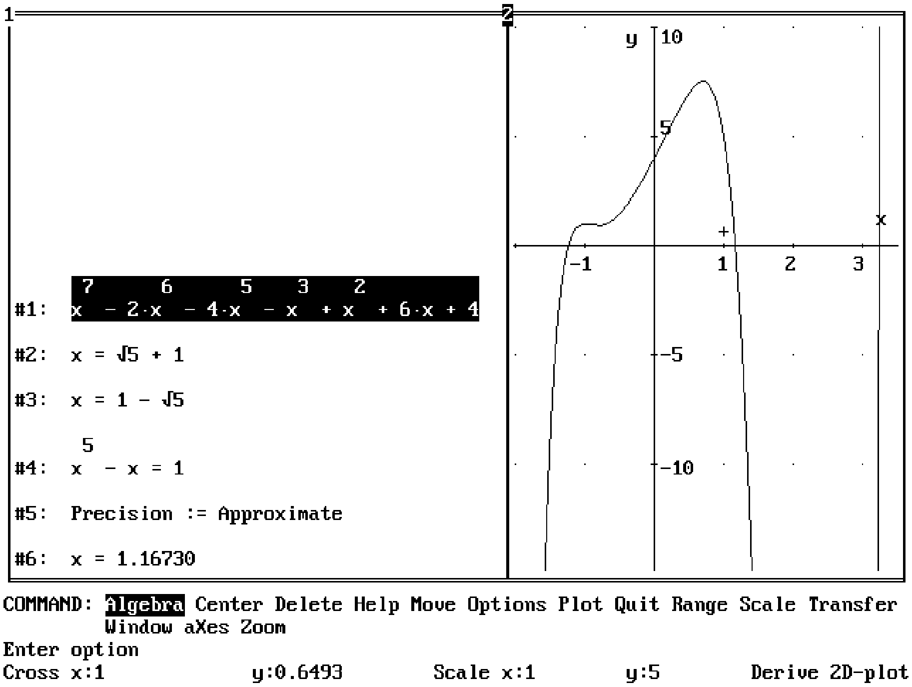
The `soLve` command can also be used to find roots of a polynomial of degree less than four. For higher degree polynomials or polynomial equations, the `soLve` command will try to find a numerical approximation of a real root after you have turned *Derive* in approximate mode. The system will ask you for the domain to search for a solution. A graph of the polynomial may indicate where to look for real roots. Figure 2.16 on the next page illustrates the root finding of a seventh degree polynomial. In this screen dump you see that the original polynomial has two real roots that can be found analytically. One is left with the fifth degree polynomial equation $x^5 - x - 1 = 0$ for which one real root can be approximated, viz., $x = 1.16730$.

2.4.3 Ordinary Differential Equations

For first and second order ordinary differential equations provides facilities to solve the ODEs analytically and numerically. We show the classical example of the step response of a linear damped mathematical pendulum

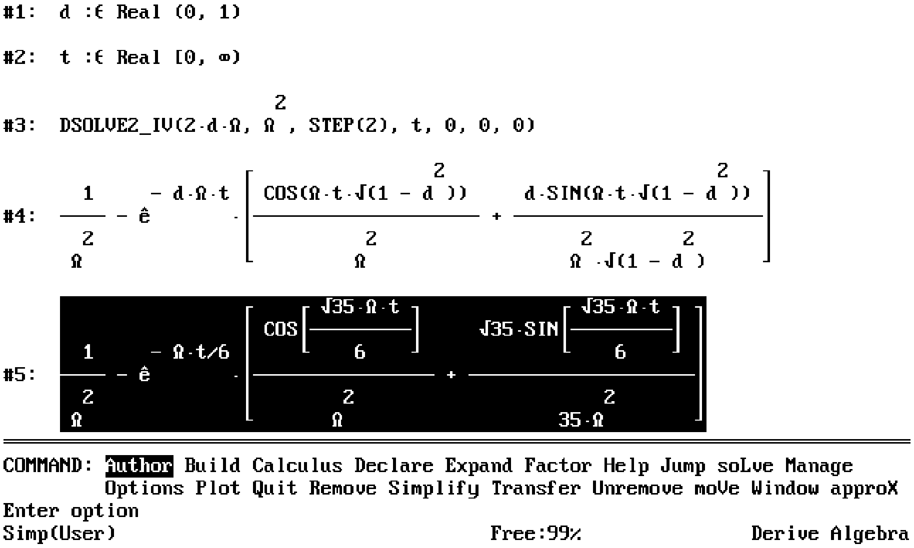
$$y'' + 2\,d\,\omega\,y' + \omega^2 y = \delta(t),$$

with initial values $y(0) = 0, y'(0) = 0$, and underdamping $0 < d < 1$. First, you issue the **`Transfer Load Utility`** command and load the file ODE2.MTH. Next, you issue the **`Declare Variable`** command and specify for `d` the interval $(0, 1)$ and declare `t` as a nonnegative real number. Now, you are ready to enter the initial value problem into *Derive* and solve the ODE analytically. In Figure 2.17 you see the analytical solution and the specialization $d = 1/6$.



```
       7      6      5     3     2
#1:  x  - 2·x  - 4·x  - x  + x  + 6·x + 4

#2:  x = √5 + 1

#3:  x = 1 - √5

      5
#4:  x  - x = 1

#5:  Precision := Approximate

#6:  x = 1.16730

COMMAND: Algebra Center Delete Help Move Options Plot Quit Range Scale Transfer
         Window aXes Zoom
Enter option
Cross x:1            y:0.6493        Scale x:1        y:5        Derive 2D-plot
```

Figure 2.16: Root finding



```
#1:  d :∈ Real (0, 1)

#2:  t :∈ Real [0, ∞)

                          2
#3:  DSOLVE2_IV(2·d·Ω, Ω , STEP(2), t, 0, 0, 0)

                                        2                          2
      1       - d·Ω·t   ┌ COS(Ω·t·√(1 - d ))     d·SIN(Ω·t·√(1 - d )) ┐
#4:  ──── - ê         · │ ─────────────────── + ───────────────────── │
       2                │          2                2         2       │
      Ω                 └         Ω                Ω ·√(1 - d )       ┘

                        ┌      ┌ √35·Ω·t ┐               ┌ √35·Ω·t ┐ ┐
                        │  COS │ ─────── │       √35·SIN │ ─────── │ │
      1       - Ω·t/6   │      └    6    ┘               └    6    ┘ │
#5:  ──── - ê         · │ ──────────────────── + ─────────────────── │
       2                │          2                      2          │
      Ω                 └         Ω                   35·Ω           ┘

COMMAND: Author Build Calculus Declare Expand Factor Help Jump soLve Manage
         Options Plot Quit Remove Simplify Transfer Unremove moVe Window approX
Enter option
Simp(User)                              Free:99%             Derive Algebra
```

Figure 2.17: Solving a 2nd order ODE

2.4.4 Matrices

To generate a 3×3 matrix whose (i,j)-th element is $\gcd(i,j)$ you can issue the **Declare Matrix** command and fill in on request all matrix elements.

```
         / 1  1  1 \
         |         |
1:       | 1  2  1 |
         |         |
         \ 1  1  3 /
```

Alternatively, you can enter the following command to assign `M` the requested matrix.

```
2:   m := VECTOR(VECTOR(GCD(i, j), i, 1, 3), j, 1, 3)
```

```
         / 1  1  1 \
         |         |
3:       | 1  2  1 |
         |         |
         \ 1  1  3 /
```

Standard matrix operations are available.

```
          / 3  4   5 \
      2   |          |
4:   m  = | 4  6   6 |
          |          |
          \ 5  6  11 /
```

```
5:   DET(m) = 2
```

```
           /   5           1  \
           |  ---   -1  - --- |
           |   2           2  |
      -1   |                  |
6:   m   = |   -1    1    0   |
           |                  |
           |    1         1   |
           | - ---   0   ---  |
           \    2         2   /
```

```
     /   5           1  \
     |  ---   -1  - --- |
     |   2           2  |   / 1  1  1 \   / 1  0  0 \
     |                  |   |         |   |         |
7:   |   -1    1    0   | . | 1  2  1 | = | 0  1  0 |
     |                  |   |         |   |         |
     |    1         1   |   \ 1  1  3 /   \ 0  0  1 /
     | - ---   0   ---  |
     \    2         2   /
```

```
8:   EIGENVALUES(m, alpha)

     /                          /     / 3 SQRT(111) \ \
     |                          | ATAN|-------------| |
     |                          |     \      37     / |
9:   |             4 SQRT(3) SIN|---------------------|
     |                          \          3          /
     |alpha = 2 - --------------------------------------,
     \                               3


                           /     / 3 SQRT(111) \        \
                           | ATAN|-------------|        |
                           |     \      37     /     pi |
              4 SQRT(3) SIN|--------------------- + ----|
                           \          3               3 /
     alpha = --------------------------------------------- + 2,
                                      3

                               /     / 3 SQRT(111) \        \ \
                               | ATAN|-------------|        | |
                               |     \      37     /     pi | |
                  4 SQRT(3) COS|--------------------- + ----| |
                               \          3               6 / |
     alpha = 2 - ---------------------------------------------|
                                     3                        /

10:  [alpha = 1.46081, alpha = 4.21432, alpha = 0.324869]
```

The procedure `eigenvalues` tries to compute the eigenvalues as explicitly as possible (therefore, it introduces radicals). The `approX` command gives the numerical approximations. Similarly, eigenvectors can be computed both analytically and numerically.

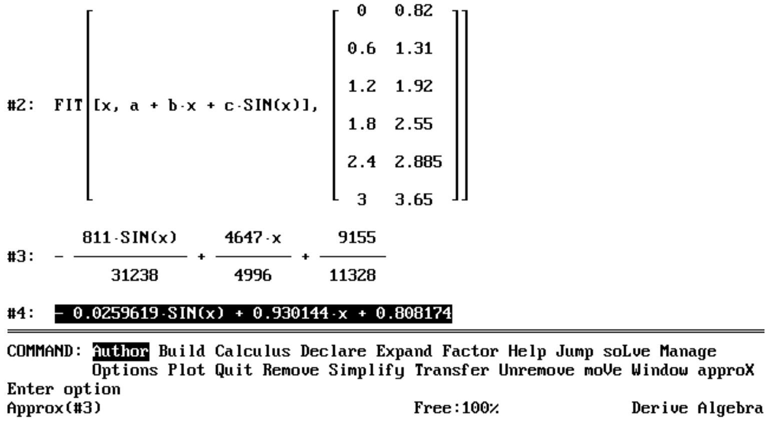
2.5 Numerical Calculus

Besides numerical integration and numerical solving of systems of equations *Derive* offers numerical tools like curve fitting and numerical solving of ODEs. As an example we shall consider linear regression and curve fitting, and Newton-Raphson's method for finding roots of functions.

In an algebra window you issue the **Declare Matrix** and select the number of rows and 2 columns. Enter the data in the row order 0, 0.82, 0.6, 1.31, ...

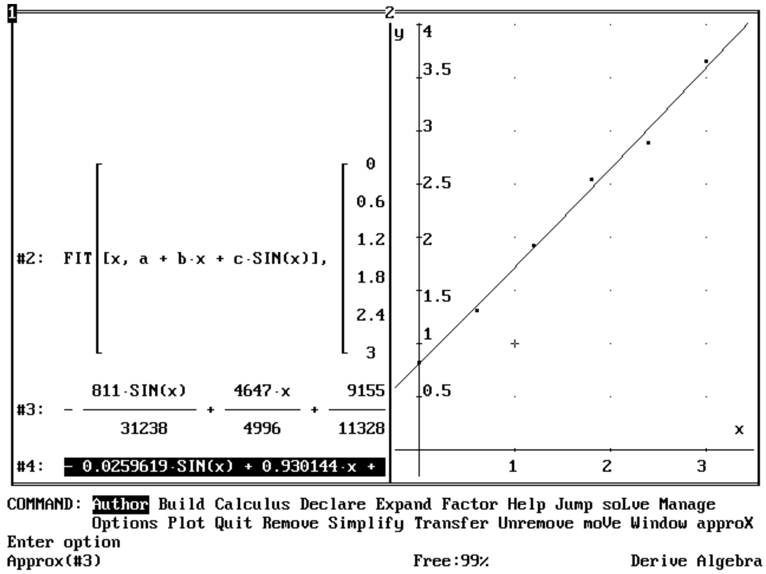
```
     /  0   0.82 \
     |           |
     | 0.6  1.31 |
     |           |
     | 1.2  1.92 |
1:   |           |
     | 1.8  2.55 |
     |           |
     | 2.4  2.85 |
     |           |
     \  3   3.65 /
```

Use the `Fit` command to find the the equation of the best fit of the form $a + bx + c\sin x$. You **`Author`** `FIT( [x, a + b x + c SIN(x)],` and then press the `F3` function key to insert the highlighted expression (in this case, the data matrix). You finish the command with a bracket. **`S`**`implify` to obtain the equation of the best fit. In Figure 2.18 you see the corresponding *Derive* screen. You can plot the matrix data values and the fitted curve (see Figure 2.19).



```
                                           ┌  0    0.82  ┐┐
                                           │ 0.6   1.31  ││
                                           │ 1.2   1.92  ││
#2:  FIT [x, a + b·x + c·SIN(x)],          │ 1.8   2.55  ││
                                           │ 2.4   2.885 ││
                                           └  3    3.65  ┘┘

        811·SIN(x)     4647·x      9155
#3:  − ─────────── + ──────── + ───────
          31238         4996       11328

#4:  − 0.0259619·SIN(x) + 0.930144·x + 0.808174

COMMAND: Author Build Calculus Declare Expand Factor Help Jump soLve Manage
         Options Plot Quit Remove Simplify Transfer Unremove moVe Window approX
Enter option
Approx(#3)                                    Free:100%          Derive Algebra
```

Figure 2.18: Linear regression



```
                                         ┌  0
                                         │ 0.6
                                         │ 1.2
#2:  FIT [x, a + b·x + c·SIN(x)],        │ 1.8
                                         │ 2.4
                                         └  3

       811·SIN(x)      4647·x       9155
#3:  − ─────────── + ──────── + ───────
          31238         4996       11328

#4:  − 0.0259619·SIN(x) + 0.930144·x +

COMMAND: Author Build Calculus Declare Expand Factor Help Jump soLve Manage
         Options Plot Quit Remove Simplify Transfer Unremove moVe Window approX
Enter option
Approx(#3)                                    Free:99%           Derive Algebra
```

Figure 2.19: Curve fitting

In order to be able apply the Newton-Raphson method for root finding of a function you must first issue the **`T`**`ransfer` **`L`**`oad` **`U`**`tility` command and load the utility file SOLVE.MTH. Then you have the

`Newtons` command at your disposal to do the work (issue the `approX` command instead of the `Simplify` command).

```
5:   NEWTONS([COS(x) - x], [x], [1], 4)

     /     1    \
     |          |
     | 0.750363 |
     |          |
6:   | 0.739112 |
     |          |
     | 0.739085 |
     |          |
     \ 0.739085 /
```

2.6 Programming

Derive's programming facilities are limited to procedural programming constructs such as iteration and choice, some functional programming paradigms, and recursion.

2.6.1 Functional Programming

To start with functional constructs, you can enter a vector of squares in a single command by

```
             2
1:   VECTOR(i , i, 1, 10)

2:   [1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

or by entering the **Declare** **vectoR** command and filling out all individual vector elements. To multiply each element by two, you can specify this as an operation to be applied on the whole expression (issue the **Build** command)

```
3:   [1, 4, 9, 16, 25, 36, 49, 64, 81, 100] 2

4:   [2, 8, 18, 32, 72, 98, 162, 200]
```

We give three other examples of mapping a function over a data structure.

```
             i
5:   VECTOR(x , i, -3, 3)

     /  1     1    1          2   3\
     |----, ----, ---, 1, x, x , x |
6:   |  3     2    x               |
     \ x     x                     /
```

```
     d  /  1     1    1          2   3\
     -- |----, ----, ---, 1, x, x , x |
7:   dx |  3     2    x               |
        \ x     x                     /

     /    3       2       1                 2\
     |- ----, - ----, - ----, 0, 1, 2 x, 3 x |
8:   |    4       3       2                  |
     \   x       x       x                   /

                    i + j
9:   VECTOR(VECTOR(x      - 1, i, 1, 2), j, 1, 2)

     /  2       3     \
     | x  - 1  x  - 1 |
10:  |                |
     |  3       4     |
     \ x  - 1  x  - 1 /
```

Factorizing all matrix elements gives

```
     /                                   2            \
     |    (x - 1) (x + 1)      (x - 1) (x  + x + 1)   |
11:  |                                                |
     |           2                             2      |
     \ (x - 1) (x  + x + 1)  (x - 1) (x + 1) (x  + 1) /
```

2.6.2 Iteration

With a number x_{old} we associate a new value x_{new} by applying the function f, i.e.

$$x_{new} = f(x_{old}), \quad \text{or} \quad x_{new} \xleftarrow{f} x_{old}.$$

Next, we interpret the number x_{new} as an new x_{old} in the above functional application. We speak of the n-fold iteration when we repeat the update formula $x \xleftarrow{f} x$ n times starting with x_{old}. E.g, for $n = 4$ we get

$$x_{new} = f(f(f(f(x_{old})))), \quad \text{or} \quad x_{new} \xleftarrow{f} x \xleftarrow{f} x \xleftarrow{f} x_{old}.$$

In a procedural programming language such as *Derive*, an n-fold iteration starting at x_0 can be implemented by the function call `ITERATE( f(x) ,x , x_0, n )` followed by the `Simplify` command. As a concrete example, we consider the following arithmetic sequence

$$3, 8, 13, 18, \ldots,$$

i.e.

$$v_0 = 3, \quad \text{and} \quad v_{i+1} = v_i + 5, \text{ for } i > 0.$$

The term v_{10} can be computed in the following way.

```
12:  ITERATE( x+5, x, 3, 10 )

13:  53
```

With `ITERATES` you get the vector of computed result.

```
14:  ITERATES( x+5, x, 3, 10 )

15:  [3, 8, 13, 18, 23, 28, 33, 38, 43, 48, 53]
```

When we want to approximate a zero a of the (2-times differentiable) function f by the Newton-Raphson method from the starting point s, we consider the iteration function

$$it : x \longrightarrow x - \frac{f(x)}{f'(x)}$$

and start applying it to s. We can implement it in *Derive* in a one-line procedure.

```
16:  FINDROOT(f,x,x0,n) := ITERATE(x-f/Dif(f,x), x, x0, n)


                2
17:  FINDROOT(x  - 2, x, 1, 5)
```

S`implify` yields

```
      886731088897
18:  --------------
      627013566048
```

whereas `appro`**X** gives the following result.

```
19:  1.41421
```

Our procedure `FINDROOT` is almost the same as the `Newton` routine in the utility file SOLVE.MTH.

```
NEWTONS(u,x,x0,n):=NEWTON_AUX(APPEND(VECTOR(DIF(u,ELEMENT(x,k_)),k_,1,DIMENSIO~
N(x)),[u])`,x,x0,n)
```

The tilde at the end of the first line in the definition of `NEWTONS` is the continuation character in *Derive*.

2.6.3 Recursion

Recursive procedures are allowed in *Derive*. For example, the factorial function can be defined as follows.

```
1:   FACTORIAL(n) := IF(n < 1, 1, n FACTORIAL(n - 1))

2:   FACTORIAL(5)

3:   120
```

This example also illustrates the `IF` control structure of choice in *Derive*. Although recursive programming is allowed in *Derive*, it may not give fast implementations. Below, the function `F` computes Fibonacci polynomials in a slow way.

```
4:   F(n, x) := IF(n < 2, n, x F(n - 1, x) + F(n - 2, x))

5:   F(15, y)

      14        12        10          8         6         4        2
6:   y   + 13 y   + 66 y   + 165 y  + 210 y  + 126 y  + 28 y  + 1
```

On a 486 WindDX2/50MHz PC this computations takes about twenty seconds, whereas the following iterative procedure computes the polynomial in one second.

```
7:   F(n, x) := element( ITERATE([ELEMENT(v, 2), x ELEMENT(v, 2) +~
                                  ELEMENT(v,1)], v, [0, 1], n), 1 )

8:   F(15, y)

      14        12        10         8         6         4        2
9:   y   + 13 y   + 66 y   + 165 y  + 210 y  + 126 y  + 28 y  + 1
```

The idea in the second procedure is to use a two-element vector `v` to retain the two most recently computed Fibonacci polynomials in the computation.

2.7 Simplification Rules

Derive provides the menu items **`Simplify`**, **`Expand`**, and **`Factor`** to simplify expressions in which special functions like trigonometric functions, logarithm, and exponential occur. *Derive* is very precise in checking whether transformations of formulae are valid or not on the domain of computation. If necessary, you have to issue the **`Declare Variable`** command and restrict the domain of a variable. For functions with simplification rules in two directions such as the rule $\ln(x\,y) = \ln x + \ln y$ you also have to specify the direction via the **`Manage`** menu item. Examples show best what can be achieved.

Let us consider the transformation $\ln(x\,y) = \ln x + \ln y$ in both directions. It is only valid when y is real or complex and x is nonnegative (or with the roles of x and y interchanged). *Derive* refuses to do the transformation unless the necessary condition is satisfied. So, let us first issue the **`Declare Variable`** command and specify `x` as a nonnegative number. Next, we issue the **`Manage Logarithm`** command to control the direction of simplification and choose the `Expand` option.

```
1:   x :epsilon Real [0,inf)

2:   Logarithm := Expand
```

The expression

```
3:   LN(x y)
```

is now **`Simplified`** to

```
4:   LN(x) + LN(y)
```

The `Collect` option in the **`Manage Logarithm`** menu must be chosen to **`Simplify`** in the opposite direction and combine sum of logarithms into the logarithm of a product.

```
5:   Logarithm := Collect

6:   LN(x y)
```

In order to illustrate that *Derive* only does those simplifications about which it is sure that the transformations are valid, we enter under the above circumstances the following sum of logarithms.

```
7:   LN(x) + LN(y) + LN(z)
```

Simplification only partly combines logarithms.

```
8:   LN(x y) + LN(z)
```

You can get the simplification

```
9:   LN(x z) + LN(y)
```

by choosing in the **`Manage Ordering`** command the x, z, y order of variables.

Trigonometric simplification is another highlight of *Derive*. Use the **`Manage Trigonometry`** menu and choose the direction of simplification and the preference of simplification of powers of sines or cosines. The `Auto` option can be chosen when you leave it up to *Derive* to make a heuristic choice. For example, in the `Auto` mode, *Derive* simplifies

```
           2           2
10:  COS(x)  + 2 SIN(x)
```

into

```
           2
11:  SIN(x)  + 1
```

But if you prefer cosines over sines, you select **`Manage Trigonometry`** `Toward: Cosines` and get

```
12:  Trigpower := Cosines
```

```
               2
13:  2 - COS(x)
```

When you enter

```
14:  SIN(x + y)
```

and select **`Manage Trigonometry`** `Direction: Expand`,

```
15:  Trigonometry := Expand
```

then simplification yields

```
16:  COS(x) SIN(y) + SIN(x) COS(y)
```

On its turn, choosing the `Collect` direction, gives back the original expression upon simplification.

2.8 Derive Packages

Many sophisticated mathematical procedures are written in the *Derive* programming language and stored in the so-called *utility files*. Below, we tabulate the utility files and their purposes; we leave out the extension .MTH that indicate that the file is a *Derive* file.

File name	*Purpose*
APPROX	Pade Rational Approximation
BESSEL	Bessel and Airy Functions
DIF_APPS	Applications of Differentiation
ELLIPTIC	Elliptic Integrals
ENGLISH	English Measurement Units
EXP_INT	Exponential, Log, Sine, and Cosine Integrals
FRESNEL	Fresnel Integrals
GRAPHICS	Plotting Space Curves, Complex Expressions
HYPERGEO	Hypergeometric Functions
INT_APPS	Applications of Integration
METRIC	Metric Measurement Units
MISC	Miscellaneous Utilities
NUMBER	Number Theory
NUMERIC	Numerical Differentiation and Integration
ODE1	First-order Ordinary Differential Equations
ODE2	Second-order Ordinary Differential Equations
ODE_APPR	Approximately Solving ODEs
ORTH_POL	Orthogonal Polynomials
PHYSICAL	Fundamental Physical Constants
PLOT2D, PLOT3D, PLOTPARA	Plot examples
PROBABIL	Additional Probability Functions
RECUREQN	Solving Recurrence Relations
SOLVE	Solving Nonlinear Systems & Complex Solutions
VECTOR	Additional Vector and Matrix Functions
ZETA	Zeta, Polylogarithm, and Dilogarithm Functions

We give one example: Laplace transforms. You make the following steps: Load the utility file INT_APPS.MTH, declare the variable `s` to be positive, enter the requested Laplace transform, and simplify. Two concrete functions:

```
               3/2
1:   LAPLACE(t    , t, s)

       inf
     /       - s*t  3/2
2:   /    #e     *t    dt
       0
```

Deliberately we forgot to declare the variable `s` as positive real number. If we do so, *Derive* can compute the Laplace transform.

```
       3 SQRT(pi)
4:   -----------
          5/2
        4 s

             /          2      \
5:           | 2     - t       |
     LAPLACE\t  + #e     , t, s/
```

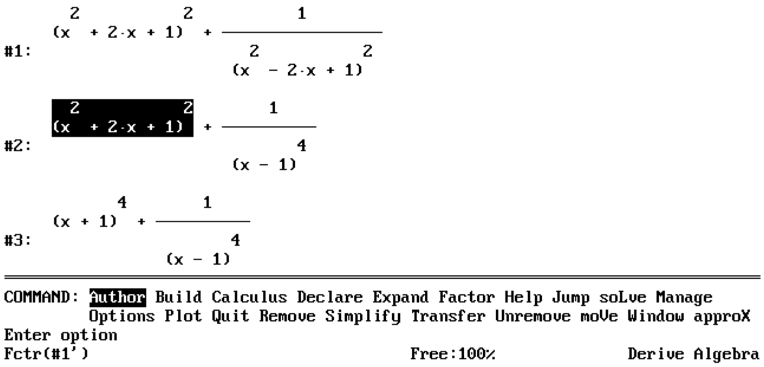
```
                  2
                 s / 4 /       / s \\
      SQRT(pi) #e      |1 - ERF|---||
6:                     \       \ 2 //      2
     --------------------------------- + ----
                     2                     3
                                          s
```

2.9 Formatting

Derive allows saving of expressions into different formats such as C, Fortran, Pascal, or Basic. For example, issue the **Transfer Save Fortran** command if you want to save an expression in Fortran style. But don't expect to much: it is a literal transcription from one language into another and no code optimization is done.

2.10 Interface

Much as already been said about *Derive*'s user interface. Obvious things like command line editing and screen saving are available in *Derive*. In this section we shall only consider two important aspects: manipulation of subexpressions and the demonstration mode.



```
       2          2             1
    (x  + 2·x + 1)  + ----------------
#1:                        2         2
                         (x  - 2·x + 1)

       2          2          1
    (x  + 2·x + 1)  + -----------
#2:                              4
                          (x - 1)

          4         1
    (x + 1)  + -----------
#3:                       4
                   (x - 1)

COMMAND: Author Build Calculus Declare Expand Factor Help Jump soLve Manage
         Options Plot Quit Remove Simplify Transfer Unremove moVe Window approX
Enter option
Fctr(#1')                                   Free:100%            Derive Algebra
```

Figure 2.20: Selecting a subexpression

In *Derive* it is easy to apply operations on parts of formulae. In Figure 2.20 you see how you can simplify the expression

$$(x^2+2x+1)^2+\frac{1}{(x^2-2x+1)^2}$$

into

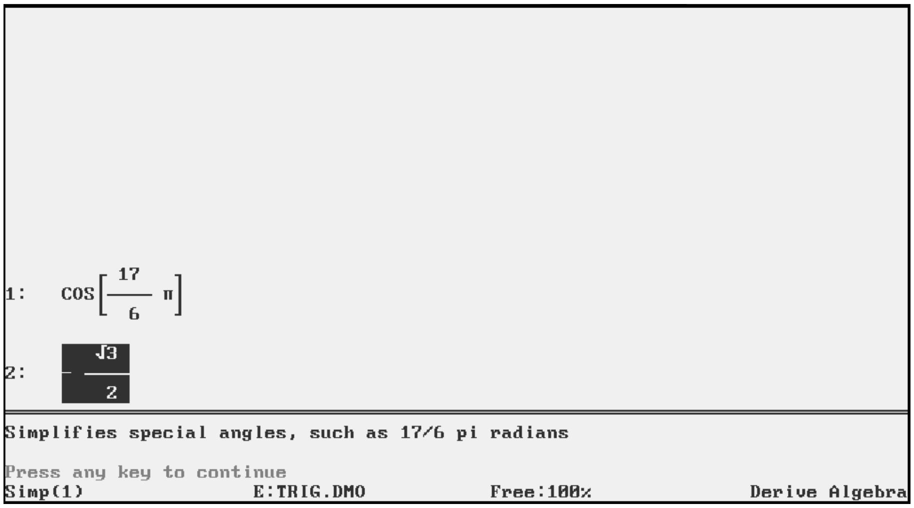
$$(x+1)^4+\frac{1}{(x-1)^4}.$$

Factorization of the whole expression just gives second expression in Figure 2.20. By the arrow keys you can select the second part of the first expression. When you issue the **Factor** command, it will apply factorization only on the highlighted part. This is how the third expression in Figure 2.20 was obtained. You also see on the screen how $(x^2 + 2x + 1)^2$ is selected in this expression to be processed further.

You can issue the **`Transfer` `D`**`emo` command to read expressions from a demonstration file (with default extension DMO). After each expression is read and displayed in the working area, the expression is simplified and the result is displayed. *Derive* then waits for you to press any key to continue or the < Esc > key to suspend the demo mode for free performance of *Derive* commands (issue again the **`Transfer` `D`**`emo` command to resume the demonstration). *Derive* comes with the following demonstration files: ALGEBRA, ARITH, CALCULUS, FUNCTION, MATRIX, and TRIG (with extension DMO). Let us look at one of these files, say `TRIG.DMO`. The beginning of its contents is

```
; Simplifies special angles, such as 17/6 pi radians
COS (17/6 pi)

; Type "DEG" after entering an angle in degrees
SIN (-30 DEG)

; A hidden opportunity for sin^2 + cos^2 --> 1
(SIN x)^3 + SIN x + (COS x)^2 SIN x

; Using different rules in different places
(1 - (COS x)^2)^4 (1 - (SIN x)^2)^3 ((SIN x)^2 + (COS x)^2)^5

; Judicious extraction of phase angles
SIN (pi / 4 + a) - SIN (0.25 (pi - 4 a))

SIN (1.5 pi/2+x) / COS (pi+x) + COS ((3pi-2x)/2) / SIN (pi/2+x)
```

The start of the demonstration is shown in Figure 2.21. Note that the comment "Simplifies special angles, such as 17/6 pi radians" which is the first line in the file starting with a semicolon is shown in the status line.



1: $\cos\left[\frac{17}{6}\pi\right]$

2: $-\frac{\sqrt{3}}{2}$

```
Simplifies special angles, such as 17/6 pi radians

Press any key to continue
Simp(1)              E:TRIG.DMO          Free:100%            Derive Algebra
```

Figure 2.21: *Derive* in demonstration mode

2.11 Saving and Quitting

You can save a *Derive* session in ascii format by the **`Transfer Save Derive`** command. Finally, to quit *Derive*, you select `Quit` from the menu. The system asks you to confirm this action.

Chapter 3

Literature

3.1 Books, Manuals, and Reports

[1] D.C. Arney. *Derive Laboratory Manual to Differential Equations.* Addison-Wesley, 1991. pp. 255, ISBN 0-201-57268-0.

[2] D.C. Arney. *Exploring Calculus with Derive.* Addison-Wesley, 1992. pp. 166, ISBN 0-201-52839-8.

[3] D.C. Arney. *Differential Equations with Derive.* MathWare, Urbana/USA, 1993. pp. 284, ISBN 0-9623629-3-X.

[4] O. Asai. *Derive Nyumon.* SEG Corporation, Tokyo/Japan, 1990. pp. 68, ISBN 4-87243-001-8.

[5] O. Asai. *Derive.* SEG Corporation, Tokyo/Japan, 1994. pp. 477, ISBN 4-87243-021-2.

[6] Baccelli, Lorenzi, and Perotti. *Analisi Matemática con Derive.* McGraw-Hill, Milano, 1992. pp. 242, ISBN 88-386-0675-7.

[7] A. van Bemmelen. Derive, een handleiding. Technical report, Hogeschool Gelderland, Educatieve Faculteit, 1990.

[8] J.S. Berry, E. Graham, and A.J.P. Watkins. *Learning Mathematics through Derive.* Ellis Horwood, 1993. pp. 371, ISBN 0-13-037532-2.

[9] H. Bjornestadt et al. *Matematikk for Oekonom og Samfunnsfag.* Hoyskole Forlaget - Norwegian Academic Press, 1994. pp. 489.

[10] J. Böhm. *Teaching Mathematics with Derive, Proceedings of the KREMS '92 conference.* Chartwell-Bratt, 1993. pp. 300, ISBN 0-86238-319-6.

[11] S. Cappucio and G.C. Barozzi. *Mathematica I.* Consorzio Nettuno, Bologna, 1993. ISBN 88-371-0667-X.

[12] A. Carrillo and I. Llamas. *DERIVE - Aplicaciones matemáticas para PC.* ra-ma, Madrid, 1994. pp. 251, ISBN 84-7897-138-6.

[13] I.C. Chadid. *Como Hacer Matematicas con DERIVE.* Editorial Reverte Colombiana, Bogota, 1992. pp. 446, ISBN 958-95511-0-6.

[14] J. Dabkowski. *Derive Wersja 2.53, Niezwykly nuaczyciel matematyki.* Warszawa, 1993. pp. 79, ISBN 83-85137-39-4.

[15] B. H. Denton. *Learning Linear Algebra Through Derive.* Ellis Horwood, 1994. pp. 300.

[16] P. Drijvers. *Wiskunde leren met Derive; Docentenboek, Leerlingenboek.* Wolters-Noordhoff, 1992. pp. 80, ISBN 90-01-25992-8.

[17] W. Ellis Jr. and E. Lodi. *A Tutorial Introduction to Derive.* Brooks/Cole Publishing Co., 1991. pp. 94, ISBN 0-534-15522-7.

[18] B. Evans and J. Johnson. *Uses of Technology in the Mathematics Curriculum.* CipherSystems, 717 Willow Dr., Stillwater, OK, 74075, USA, 1990.

[19] B. Evans and J. Johnson. *Linear Algebra with Derive.* John Wiley & Sons, 1990. pp. 280, ISBN 0-471-59194-7.

[20] A. Garcia. *Prácticas de Matemáticas con Derive.* CLAGSA, Madrid, 1993. pp. 418, ISBN 84-604-8958-2.

[21] L.G. Gilligan and J.F. Marquardt Sr. *Calculus and the Derive Program.* Gilmar Publishing Company, Cincinnati/USA, second edition, 1990. pp. 138, ISBN 0-9626661-2-2.

[22] J. Glynn. *Exploring Math from Algebra to Calculus with Derive, A Mathematical Assistant.* MathWare, 604 East Mumford, Urbana, IL, 61801, USA, 1989. pp. 153, ISBN 0-923629-0-5.

[23] P.E.J.M. Gondrie and G.A.T.M. van Alst. *Handleiding Derive, snel aan de slag met computeralgebra.* Academic Service, first edition, 1992. pp. 54, ISBN 90-6233-886-0.

[24] B. Grabinger. *Stochastik mit Derive.* Dümmler Verlag, Bonn, 1994. pp. 160.

[25] M. de Guzmán and B. Rubio. *Análisis Matemático 3 - sucesiones y series de funciones, números complejos, DERIVE, aplicaciones.* Ediciones Pirámide, Madrid, 1993. pp. 338, ISBN 84-368-0759-6.

[26] D. Harper, C. Wooff, and D. Hodgkinson. *A Guide to Computer Algebra Systems.* John Wiley & Sons, 1991. pp. 148.

[27] A. Heck. Calculus with Derive. Technical report, CAN Expertise Center, 1992. pp. 34.

[28] H. Heugl and B. Kutzler. *Derive in Education — Opportunities and Strategies, Proceedings of the KREMS '93 Conference.* Chartwell-Bratt, 1994. pp. 302, ISBN 0-86238-351-X.

[29] A. Hirlimann. *Enseignement des Mathematiques et Logiciels de Calcul Formel - Derive, un outil à intégrer.* Ministrére; de l'Education Nationale, Paris, 1993. pp. 199.

[30] Hischer. *Mathematikunterricht im Umbruch? Erörterungen zur möglichen "Trivialisierung" von mathematischen Gebieten durch Hardware und Software.* Franzbecker Verlag, Hildesheim, 1992.

[31] E. Hodes, M. Mallen, and M. Paige Yuhn. *Exploring Precalculus with Derive.* West Publishing Company, St. Paul/USA, 1994. pp. 161, ISBN 0-314-03687-3.

[32] L.R. Jaisingh. *College Algebra and the Derive Program.* West Publishing Company, St. Paul/USA, 1994. pp. 161, ISBN 0-314-03687-3.

[33] J. Johnson and B. Evans. *Discovering Calculus with Derive.* John Wiley & Sons, 1992. pp. 157, ISBN 0471-55155-4.

[34] M. Kammingaa-van Hulsen, P. Gondrie, G. van Alst. *Toegepaste wiskunde met computeralgebra.* Academic Service, 1994. pp. 290, ISBN 90-6233-956-5.

[35] H.-J. Kayser and R. Neveling. *Neue Medien im Unterricht - Derive.* Soester Verlagskantor, Hamm/Germany, 1993. pp. 157, ISBN 3-8D165-1759-5.

[36] J.R. Kirkwood. *Calulus Projects for Derive.* Wm. C. Brown, Dubuque/USA, 1994. pp. 190, ISBN 0-697-22888-6.

[37] W. Köpf, A. Ben-Israel, and B. Gilbert. *Mathematik mit Derive.* Vieweg, 1993. pp. 394, ISBN 3-528-06549-4.

[38] W. Köpf. *Hóhere Analysis mit Derive.* Vieweg, 1994. pp. 206, ISBN 3-528-06594-X.

[39] G. Kupchik. *DERIVE - Pomoschtschnik Matematika.* Omsk/Russia, 1993. pp. 52.

[40] B. Kutzler, F. Lichtenberger, und F. Winkler. *Softwaresysteme zur Formelmanipulation.* Expert Verlag, 7044 Ehningen bei Boblingen, Germany, 1990.

[41] B. Kutzler, B. Wall, und F. Winkler. *Mathematische Expertensysteme.* Expert Verlag, 7044 Ehningen bei Boblingen, Germany, 1992.

[42] B. Kutzler. *Mathematik am PC - Einführung in Derive.* Soft Warehouse Europe, Hagenberg/Austria, 1994. pp. 164, ISBN 3-9500364-0-7.

[43] B. Kutzler. *Mathematics on the PC - Introduction to Derive.* Soft Wareouse Europe, Hagenberg/Austria, 1994. pp. 164, ISBN 3-9500364-0-7.

[44] C.L. Leinbach. *Calculus Laboratories Using Derive.* Wadsworth Publishing Company, Belmont/USA, 1991. pp. 147, ISBN 0-534-15480-8.

[45] J.L. Llorens Fuster. *Introduccion al Uso de DERIVE - Aplicaciones al Algebra Lineal y al Calculo Infinitesimal.* Univ. Politecnica de Valencia, Valencia, 1993. pp. 270, ISBN 84-7721-199-X.

[46] J.L. Llorens Fuster. *Aplicaciones de DERIVE - Análisis Matemático - I (Cálculo).* Univ. Politecnica de Valencia, Valencia, 1993. pp. 345, ISBN 84-7721-237-6.

[47] J.L. Llorens Fuster. *Aplicaciones de DERIVE - Geometría Afín Y Euclídea de* R^3. Univ. Politecnica de Valencia, Valencia, 1993. pp. 66.

[48] J.L. Llorens Fuster. *Aplicaciones de DERIVE - Álgebra Lineal (Fundamentos).* Univ. Politecnica de Valencia, Valencia, 1994. pp. 345, ISBN 84-7721-255-4.

[49] M. Lokar. *DERIVE, program za simbolicno racunanje.* Ljubljana/Slovenia, 1991. pp. 56.

[50] Manara and Perotti. *Lineare Geometria con Derive.* McGraw-Hill, Milano, 1992.

[51] A. Marlewski. *Derive Pomocnik matematyczny wersja 2.0.* Wydawnictwo NAKOM, Poznan, 1992. pp. 355, ISBN 83-85060-49-9.

[52] P. de Marois. *College Algebra Laboratories Using Derive.* MathWare, 604 East Mumford, Urbana, IL, 61801, USA, 1989. pp. 125, ISBN 0-9623629-1-3.

[53] M.A. Martinez Sanchez. *Problems Y Prácticas de Algebra.* Univ. Politecnica de Madrid, Madrid. pp. 32, ISBN 84-87238-23-8.

[54] D.M. Mathews. *Precalculus Investigations Using Derive.* HarperCollins College Publishers, New York/USA. ISBN 0-673-99097-4.

[55] Mauve und Moos. *Mathematik mit Derive, Arbeitsblätter zur experimentellen Mathematik mit dem Computer.* Dümmlerbuch 4588. Dümmler Verlag, Bonn, 1993.

[56] R. Minano-Rubio. *?Cómo Lo Hago con DERIVE? - Guía para empezar a usar DERIVE.* ra-ma, Madrid. pp.512, ISBN 84-87238-20-3.

[57] B. Mis. *Derive 2.06.* Wydawnictwa Naukowo-Techniczne, Warszawa, 1993. pp. 149, ISBN 83-204-1577-2.

[58] J. Monaghan and T. Etchells. *Computer Algebra Systems in the Classroom.* Center for Studies in Science and Mathematics Education, University of Leeds, 1994. pp. 84, ISBN 0-904421-52-X.

[59] D.H. Olwell and P.J. Driscoll. *Calculus and Derive.* Saunders Publishing, Philadelphia, 1992. pp. 154, ISBN 0-03-076156-5.

[60] L. Overeem-Loohuis and C. de Ridder. Toegepaste Computeralgebra, experimentele introductie voor het HTO, studierichting electrotechniek. Technical report, Christelijke Hogeschool Windesheim, sector HTO, Zwolle, 1993.

[61] C. Paulogorrán and C. Pérez. *Cálculo Matemático con Derive para PC.* RA-MA, Madrid, 1993. pp. 389, ISBN 84-7897-119-X.

[62] E. Rahi. *Derive.* Kirjapaino Grafia Oy, Turku/Finland, 1990. pp. 52, ISBN 951-95722-7-9.

[63] J. Reusen and R. Stegeman. Derive bekeken. Technical report, Hogeschool Gelderland, Educatieve Faculteit, 1990.

[64] S. Rossetto. *Derive per la scuola.* McGraw-Hill Libri, Milano/Italy, 1992. pp. 52, ISBN 951-95722-7-9.

[65] M. Salter and L.G. Gilligan. *Linear Algebra Experiments Using the Derive Program.* Gilmar Publishing Company, 1992. pp. 122.

[66] G. Scheu. *Arbeitsbuch ComputerAlgebra mit Derive.* Dümmler Verlag, Bonn, 1992.

[67] G. Scheu. *Derive Im Mathematik- und Physikunterricht.* Dümmler Verlag, Bonn, 1994. pp. 154, ISBN 3-427-45721-4.

[68] S.A. Schonefeld. *Numerical Analysis via Derive.* MathWare, Urbana/USA, 1992. ISBN 0-9623629-2-1.

[69] D. Sjöstrand and R. Wasen. Experiment med matematik. DERIVE i undervisningen. Technical report, Onsala/Sweden, 1992. pp. 94.

[70] D.R. Stoutemyer, A. Rich, and J. Rich. *Derive User Manual.* Soft Warehouse Europe GmbH, 1991. pp. 248.

[71] L. Townsley-Kulich and B. Victor. *The Derive Calculus Workbook.* HarperCollins College Publishers, New York, 1994.

[72] R. Wasen, D. Sjöstrand. *Svensk Handledning med Ovningsuppgifter och Introduktion till Algebraisk Modellering och Datoralgebra Derive.* GrafiTex-Data, Kisa/Sweden, 1991. pp. 33.

[73] A.J.P. Watkins. *A Guide to Using Derive.* Polytechnic South West, Plymouth, 1992.

[74] A.J.P. Watkins. *Derive Based Investigations for Post-16 Core Mathematics.* Chartwell-Bratt, 1992. pp. 64, ISBN 0-86238-312-9.

[75] A.J.P. Watkins. *Usando Derive - Fundamentos da Matematica.* editora Grypho, 1993.

[76] C. Werner. *Algebraprogramme und Tabellenkalkulation im Unterricht.* Hölder-Pichler-Tempsky, Wien, 1993. pp. 271, ISBN 3-209-01608-9.

[77] J. Zizi. *Derive, Maple et Mathematica.* Editions Archimede, Argenteuil/France, 1993. pp. 303, ISBN 2-90-9028-13-5.

3.2 Papers and Reviews

[1] E. Ahonen. Mechanics. *Bulletin of the Derive User Group*, 9,10:10–14,12–14, 1993.

[2] Anoniem. Meer ruimte voor wiskundebegrip. *Computers op School*, 5:22–24, 1993.

[3] Anonymous. Fun with Math. *Radio-Electronics Magazine*, 1990.

[4] H. Appel and R. Brand. Joint Venture Cabri-Derive. *Bulletin of the Derive User Group*, 4:18–22, 1991.

[5] H. Appel. Derive im M-Unterricht. *Bulletin of the Derive User Group*, 5:18–21, 1992.

[6] B. van Asch. Wiskunde leren met Derive – Boekbespreking. *CAN Newsletter*, 9:17–18, 1992.

[7] K. Aspetsberger. Using Derive in Analytic Geometry. In J. Böhm, editor, *Teaching Mathematics with Derive, Proceedings of the KREMS '92 conference*, pages 21–28. Chartwell-Bratt, 1993.

[8] K. Aspetsberger. Derive for Students at the Age of 17 and 18. In H. Heugl and B. Kutzler, editor, *Derive in Education — Opportunities and Strategies, Proceedings of the KREMS '93 Conference*, pages 159–168. Chartwell-Bratt, 1994.

[9] B. Bärzel. Taylorreihenentwicklung mit Derive. *Mathematische Berichten im Unterricht*, 6, 1991.

[10] B. Bärzel und P. Drijvers. Matrizen mit Derive. *Mathematische Berichten im Unterricht*, 5, 1993.

[11] B. Bärzel. Taylor Series Expansion. In J. Böhm, editor, *Teaching Mathematics with Derive, Proceedings of the KREMS '92 Conference*, pages 57–62. Chartwell-Bratt, 1993.

[12] J., Berry, T., Graham, and A. Watkins. Integrating the Derive Program into the Teaching of Mathematics. *International Derive Journal*, 1(1):83–96, April 1994.

[13] J. Böhm. Finanzmathematik mit Derive. *Bulletin of the Derive User Group*, 1,2:13–22,9–17, 1991.

[14] J. Böhm. Solving Odes Using Derive's ODE1.MTH. *Bulletin of the Derive User Group*, 2,3:22–28,15–33, 1991.

[15] J. Böhm. How to write a Demo-file. *Bulletin of the Derive User Group*, 4:30, 1991.

[16] J. Böhm. Selfmade Scales. *Bulletin of the Derive User Group*, 5:28–30, 1992.

[17] J. Böhm. Riemann at Random. *Bulletin of the Derive User Group*, 7:17–21, 1992.

[18] J. Böhm. Maurer–Roses. *Bulletin of the Derive User Group*, 7:24–27, 1992.

[19] J. Böhm. True Riemann Rectangles. *Bulletin of the Derive User Group*, 8:18–20, 1992.

[20] J. Böhm. Trigonometry. *Bulletin of the Derive User Group*, 9:15–17, 1993.

[21] J. Böhm. Newton-Raphson's Chaos. *Bulletin of the Derive User Group*, 12:10–13, 1993.

[22] A. van den Bogert. Differentiëren en Derive. *Euclides*, 68:3–5, September 1992/1993.

[23] R. Castelletti. Esperimenti didattici con un programma di Computer Algebra. *Bulletin of the Derive User Group*, 7,8:28,15–17, 1992.

[24] J.-C. Chuan. Playing-cards Shuffling with Derive. *Bulletin of the Derive User Group*, 12:24–25, 1993.

[25] P. Drijvers. Computeralgebra en wiskundeonderwijs. *Nieuwe Wiskrant, Tijdschrift voor Nederlands Wiskundeonderwijs*, 23–26, June 1991.

[26] P. Drijvers. The Use of Graphics Calculators and Computer Algebra Systems: Differences and Similarities. *International Derive Journal*, 1(1):71–82, April 1994.

[27] D. Dyer. The Bisection Method and Derive. *Bulletin of the Derive User Group*, 11:11–13, 1993.

[28] K. Eames. Mathematical Induction with Derive. *Bulletin of the Derive User Group*, 12:26–32, 1993.

[29] A. Engel. Eine Vorstellung von Derive in Didaktik der Mathematik. *Didaktik der Mathematik*, 18, 1990.

[30] T.A. Etchells. True Riemann Rectangles. *Bulletin of the Derive User Group*, 8:15–18,21–22, 1992.

[31] T.A. Etchells. Computer Algebra Systems and Students' Understanding of the Riemann Integral. In J. Monaghan and T. Etchells, editor, *Computer Algebra Systems in the Classroom.* Center for Studies in Science and Mathematics Education, University of Leeds, 1993.

[32] K. Fuchs. Logic with Derive. *Bulletin of the Derive User Group*, 7:12–16, 1992.

[33] L. Gilligan. Learning Visually with Derive. *International Derive Journal*, 1(1):19–36, April 1994.

[34] P.E.J.M. Gondrie and G.A.T.M. van Alst. Praktikumbundel Derive, '91-'92. *CONCAVO Publikatie*, 1, 1992. pp. 12.

[35] E. Graham. Using Derive's 3D-Plot Facility: A Case Study in Mathematical Modelling. *Teaching Mathematics and its Applications*, 10(4):159–162, 1991.

[36] E.A. Herman. Derive, A Mathematical Assistent, ver. 1.22 – A review. *American Mathematical Monthly*, pages 948–959, 1990.

[37] D. Hodgkinson. A Module on Sequences and Series. *Bulletin of the Derive User Group*, 4,5:23–29,9–17, 1991 and 1992.

[38] D. Hood. Digital Filter Design. *Bulletin of the Derive User Group*, 11:14–21, 1993.

[39] A. Hoogewijs. Derive, een steun bij het wiskunde-onderwijs. *Wiskunde en Onderwijs*, 18:205–222, 1992.

[40] M. Horbatch. Solving Physics Problems in the Classroom with Derive 2.0. *Computers in Physics*, 1990 and 1991.

[41] M. Horbatch. Physics in the Classroom. *Bulletin of the Derive User Group*, 9,10:4–10,6–11, 1993.

[42] M. Hunter, P. Marshall, J. Monaghan, T. Roper, and G.T. Wain. Using computer algebra systems with younger students. In J. Monaghan and T. Etchells, editor, *Computer Algebra Systems in the Classroom.* Center for Studies in Science and Mathematics Education, University of Leeds, 1993.

[43] M.K. Hurd. A Comparison of the Use of a Computer Algebra System and a Spreadsheet to Teach the Newton-Raphson Method. In J. Monaghan and T. Etchells, editor, *Computer Algebra Systems in the Classroom.* Center for Studies in Science and Mathematics Education, University of Leeds, 1993.

[44] H.-J. Kayser. Trickfilme mit Derive. *Bulletin of the Derive User Group*, 6:10–12, 1992.

[45] U. Keusen und H.-J. Kayser. Bilder und Filme mit Derive. *Mathematische Berichten im Unterricht*, 3, 1993.

[46] W. Köpf and A. Ben-Israel. Integration mit Derive. *Didaktik der Mathematik*, 21:40–50, 1993.

[47] W. Köpf. Zur Berechnung der trigonometrischen Funktionen. *Preprint A/16-93*, 1993.

[48] W. Köpf. Ein Elementarer Zugang zu Potensreihen. *Didaktik der Mathematik*, 21:292–299, 1993.

[49] W. Köpf and A. Ben-Israel. The Definite Nature of Indefinite Integrals. *International Derive Journal*, 1(1):115–131, April 1994.

[50] A. Kozubik. Independent Replicated Experiments. *Bulletin of the Derive User Group*, 10,11:15–18,22–23, 1993.

[51] B. Kutzler. Derive – The Future of Teaching Mathematics. *International Derive Journal*, 1(1):37–48, April 1994.

[52] J.M. Lopes and J.C. Tamames. Word Processing and Derive Revisited. *Bulletin of the Derive User Group*, 11:24–25, 1993.

[53] J.M. Lopes. Minimization of a "Flat Function". *Bulletin of the Derive User Group*, 13:24–26, 1994.

[54] D. Mackie. A Chemical Reaction. *Bulletin of the Derive User Group*, 7:9–11, 1993.

[55] E. Marinell. Regression Lines. *Bulletin of the Derive User Group*, 10:15–18, 1993.

[56] A. Marlewski. Rational Collocation. *Bulletin of the Derive User Group*, 8:10–14, 1992.

[57] J. Monaghan. Using a Computer algbra System to teach Quadratic Functions. In J. Böhm, editor, *Teaching Mathematics with Derive, Proceedings of the KREMS '92 Conference.* Chartwell-Bratt, 1993.

[58] J. Monaghan. On the Succesful Use of Derive. *International Derive Journal*, 1(1):57–69, April 1994.

[59] J.W. Notenboom. Wiskunde leren met Derive – Boekbespreking. *Nieuwe Wiskrant, Tijdschrift voor Nederlands Wiskundeonderwijs*, pages 45–47, April 1993.

[60] R.P. Offerijns. Derive, een wiskundige tovenaar. *HCC Nieuwsbrief*, 123, December 52 1989.

[61] D.H. Olwell. Four Uses of Derive in the Instruction of Calculus at the United States Military Academy. In Z.A. Karian, editor, *Symbolic Computation in Undergraduate Mathematics Education, MAA Notes 24*, pages 83–90. Mathematical Association of America, 1992.

[62] N. Pitcher and M. Johnson. Two Underground Tunnels. *Bulletin of the Derive User Group*, 5:25–27, 1992.

[63] S. Pozzi. Algebraic Reasoning and Computer Algebra Systems: Freeing Students from Syntax? In H. Heugl and B. Kutzler, editor, *Derive in Education — Opportunities and Strategies, Proceedings of the KREMS '93 Conference.* Chartwell-Bratt, 1994.

[64] W. Pröpper. Von der Binomial- zur Normalverteilung. *Bulletin of the Derive User Group*, 10:26–30, 1993.

[65] S. Repo. Understanding and Reflective Abstraction: Learning the Concept of Derivatives in the Computer Environment. *International Derive Journal*, 1(1):97–113, April 1994.

[66] C.E. Reuther and J.R.V. Domingos. Extended Fourier Series. *Bulletin of the Derive User Group*, 9:18–26, 1993.

[67] A.D. Rich and D.R. Stoutemyer. Inside the Derive Computer Algebra System. *International Derive Journal*, 1(1):3–17, April 1994.

[68] N. Roorda. 'n zakjapanner is goedkoper. *Dutch Personal Computer Mazine*, pages 177–179, 1991.

[69] J.V. Royle. Inverse Laplace transforms. *Bulletin of the Derive User Group*, 4:15–18, 1991.

[70] E.A. Sawada. Extending Algebra Concepts with Technology. *Bulletin of the Derive User Group*, 12:14–19, 1993.

[71] G. Scheu. Entdeckungen in der Menge der Primzahlen mit Derive. *PM*, 34 (Heft 3), 1992.

[72] G. Scheu. Triangle of Pascal. *Bulletin of the Derive User Group*, 9:27–28, 1993.

[73] H. Scheuermann and W. Eichenauer. The Derive Screen on the TV-screen. *Bulletin of the Derive User Group*, 11:28–30, 1993.

[74] Schnegelberger and Wynand. Derive für den Analysisunterricht in der Sekundarstufe II. *Beiträge zur Mathematik*, 1990.

[75] Schönwald. Zur Evaluation von Derive. *Didaktik der Mathematik*, 19, 1991.

[76] F. Schumm. Über die Verwendung von Derive zur Erzeugung von Wertetafeln mit Hilfe der VECTOR-Anweisung. *Bulletin of the Derive User Group*, 1:9–12, 1990.

[77] F. Schumm. Berechnung und graphische Darstellung der Cornu-Spirale mit Derive. *Bulletin of the Derive User Group*, 2:7–8, 1991.

[78] F. Schumm, and J. Böhm. Wordprocessing and Derive. *Bulletin of the Derive User Group*, 2:18–21, 1991.

[79] F. Schumm. With iterates to the Chaos. *Bulletin of the Derive User Group*, 3,4:18–21,8–14, 1991.

[80] B. Simon. $200 Algebra/Symbolic Manipulation Program Takes on Mathematica. *PC Magazine*, page 26, September 1989.

[81] B. Simon. Four Computer Mathematical Environments. *Notices of the A.M.S.*, 37:861–867, 1990.

[82] B. Simon. It's Not Just for Mainframes Anymore. *PC Magazine*, August 1992.

[83] D. Sjöstrand. CAS and Spreadsheets (MS Excel). *Bulletin of the Derive User Group*, 13:13–23, 1994.

[84] R.W. Southward. Finding a Gradient. *Bulletin of the Derive User Group*, 5:21–24, 1992.

[85] D.R. Stoutemyer. Crimes and Misdemeanors in the Computer Algebra Trade. *Notices A.M.S.*, 38:778–785, 1991.

[86] D.R. Stoutemyer. Derive's Impact on Mechanics. *Bulletin of the Derive User Group*, 6:19–31, 1992.

[87] G. Wain. Some Technical Problems in the use of Derive with school Pupils. *International Derive Journal*, 1(1):49–55, April 1994.

[88] A.J.P. Watkins. Introducing Calculus with Derive. In J. Böhm, editor, *Teaching Mathematics with Derive, Proceedings of the KREMS '92 conference*, pages 1–20. Chartwell-Bratt, 1993.

[89] A.J.P. Watkins. A new approach to mathematics for engineers. *Int.J.Math.Educ.Sci.Technol.*, 24:689–702, 1993.

[90] T. Weth. A Lexicon of Curves. *Bulletin of the Derive User Group*, 11,12,13:26–28,8–9,10–12, 1993.

[91] F. Wiesenbauer. Titbits in Algebra and Numbertheory. *Bulletin of the Derive User Group*, 13:27–29, 1994.

[92] V. Winkelmann. Didaktische Beschreibung mathematischer Software am Beispiel Derive. *Beiträge zur Mathematik*, 1989.

[93] H. Wunderling. Nonstandard Analysis. *Bulletin of the Derive User Group*, 6:13–16, 1992.

[94] E. Zott. Maximum and Minimum Values. *Bulletin of the Derive User Group*, 12:20–24, 1993.