Three
Review of the Tools

The selection and development of appropriate mathematical tools occupied a critical position, not only in the early stages, but throughout the course of the project. Although a wide and growing range of advanced mathematical software applications exists, relatively few such tools were considered appropriate for the purposes of this study. Since the research was aimed at mathematical learning situations spanning the secondary school years, the vast majority of mathematical software tools were rejected, since most were designed for senior secondary, post-secondary or even professional mathematical applications, with little consideration given to students of lesser mathematical capability and experience.

The choice of appropriate software was governed principally by three criteria:

(1) Interface

(2) Cost, and

(3) Mathematical functionality.

These criteria are listed in order of the importance accorded to them when selecting tools for use within this study. The review which follows examines examples of three major software types - algebra, graphing and number tools - from these perspectives. In this way the eventual choice of tools may be better understood, and the particular strengths and weaknesses of each appreciated.

Algebraic Tools for Teaching and Learning

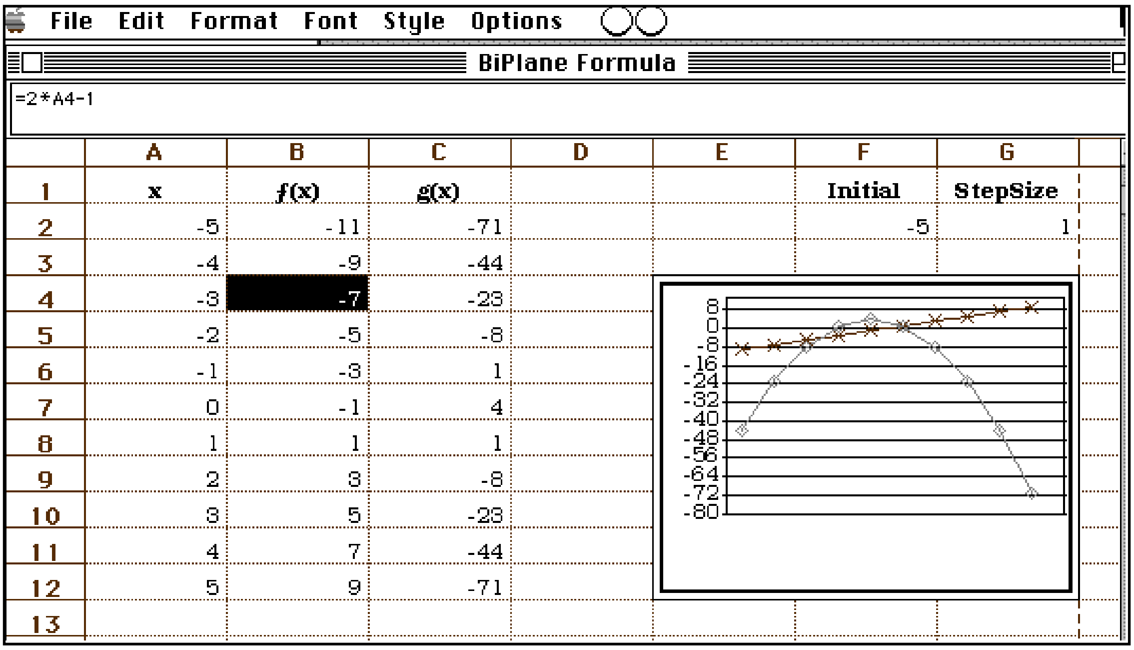
The symbolic language of mathematics was designed over time as a means of expressing the complex and powerful ideas and processes associated with the activity of *doing mathematics*, and only incidentally with the related activities of *teaching* and *learning mathematics*. Its very elegance and efficiency may in many ways serve to obscure fundamental understandings on the part of those with limited mathematical experience and, indeed, to deny entry to the “uninitiated”. When coupled with the extra demands of entering and interpreting mathematical text using a computer, additional burdens are placed upon learners who often already find difficulty enough with mathematical syntax and symbolism.

The majority of algebra and graphing software packages adopt what is, for them, the simplest approach, requiring the user to enter mathematical statements using “command-line syntax”, borrowed from computer programming. Thus, entry of an expression such as

$$\frac{1 + 2x}{\sqrt{1 - x^2}}$$

would require the user to type **(1 + 2 * x) / sqrt (1 - x^2)**. Such an entry mode not only demands that the user learns additional syntactical commands and conventions, but denies access to the important visual cues by which mathematical notation is most easily recognised and interpreted (Kirshner, 1989). While those with extensive mathematical experience may not be unduly inconvenienced by such demands, those with lesser experience, and especially those first learning the conventions of algebra, might be expected to be significantly disadvantaged.

Figure 3.1: BiPlane 2.0: A typical spreadsheet format



This problem is further exacerbated in the case of spreadsheets, in which the symbolic variable is replaced by reference to a cell location, such as A3, or even A3. An algebraic formula, such as

$$\mathbf{3x^2 - 4x + 1}$$

when adapted to a spreadsheet assumes a form such as

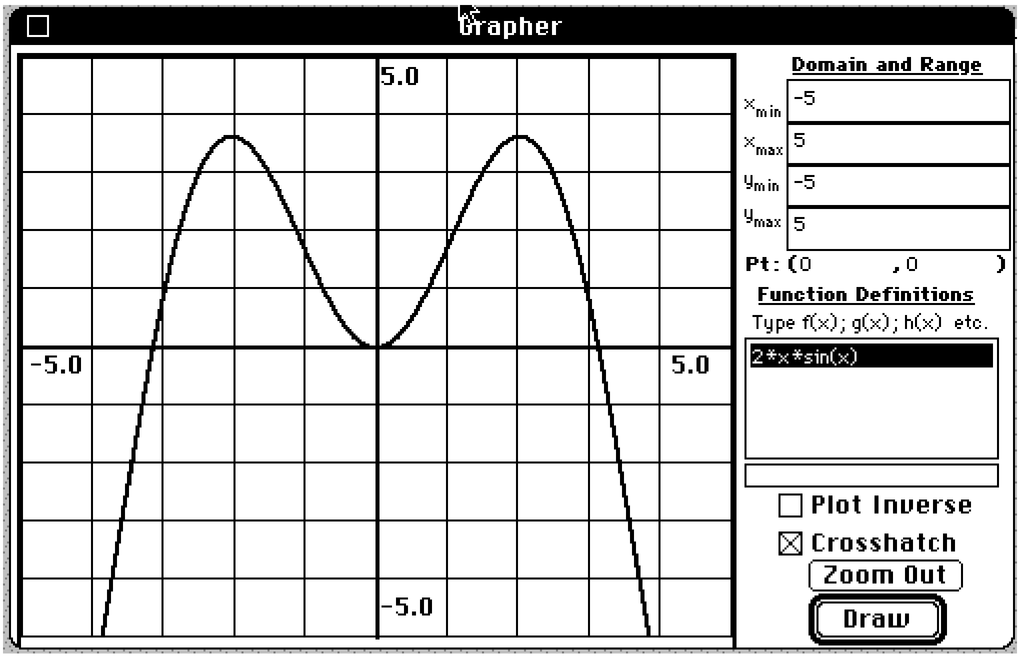
=3*A3^2-4*A3 + 1

in which the leading "=" sign indicates the commencement of a formula, and variable references point to an entry in a particular cell, A3 (see Figure 3.1). Asp, Dowsey and Stacey note the potential confusion for some students in the use of a symbol ("=") already burdened with several mathematical functions (1993b, p. 90). Some spreadsheets offer a degree of relief for this problem by allowing a cell location to be entered automatically by simply clicking on the cell, using the mouse. In this way, students do not need to enter the cell location in symbolic form, but must still specify all operations required.

This version of algebraic formatting appears to serve as both strength and weakness in defining the role of the spreadsheet as a tool for algebra learning. Although the entry and formatting of algebraic statements can be difficult and confusing, the reference to individual cell locations with distinct numerical values encourages students to perceive of variables as dynamic processes (with multiple numerical values) rather than the static placeholders so often associated with the use of symbols such as “x” and “y”. The unique capacity of the spreadsheet to expose the numerical bases for algebraic forms and expressions offers some encouragement for teachers to persevere with their use in algebra learning. It seems likely, however, that a simple “table of values” utility (in which an algebraic formula is entered in the more usual form, and features such as initial value, step size and number of steps may be controlled by the user) offers many of the advantages of the spreadsheet without the more difficult interface problems. For this reason, one of the first utilities developed for this study using *HyperCard* was a *Table of Values* tool, in which it was sought to offer the advantages of the spreadsheet in a simpler format. Designed to accompany a *HyperCard*-based graph plotter (adapted from one by Dr Khoon Yoong Wong of Murdoch University), the table of values was perceived as an important representational tool which encouraged users to perceive of algebraic forms as defined in terms of numerical processes. Dynamically linked with the graph plotter, it was seen as important for students to learn to move freely among symbolic, graphical and numerical forms.

Spreadsheets hold the honour of being the oldest mathematical software tools available for microcomputers (at least in schools), and have certainly been the among the most ubiquitous (since most school computers from the earliest days provided access to integrated software packages, traditionally offering word processing, spreadsheet and database capabilities). The difficulties associated with their use, coupled with the traditional nature of school algebra (dominated by a focus upon algebraic “objects” as opposed to the “process” approach offered by these numerical tools), have served to minimise the impact of spreadsheets upon mathematics learning situations, and may even have helped to contribute to what appears from this study to be a fairly widespread view among teachers of computers as being, for the most part, incompatible with school mathematics (Messing, 1994).

Figure 3.2: Grapher 3.61s: A typical graph plotter



If spreadsheets have largely failed to excite teachers of mathematics with their potential, the same cannot be said of graph plotters. Since spreadsheets were designed as tools for business rather than learning, the first true mathematical software available for school computers consisted of tools for plotting functions. Now increasingly available and affordable in hand-held form, the evidence of this study indicates that teachers of mathematics appear very comfortable with this application of computer technology. Participating students and student teachers had little difficulty in using graphing tools, supporting the view that these tend to sit comfortably “alongside” existing practice, as opposed to applications such as spreadsheets and computer algebra tools, which appear to critically confront such practice. Indeed, as discussed in the study which follows, while physical factors such as interface, cost and capabilities may at first appear to be the principal stumbling blocks for the use of mathematical software in schools, it is likely that there are deeper political aspects related to mathematical tool use which provide a far more influential barrier to their implementation in schools.

As specialist tools for mathematics teaching and learning have been developed over the past decade, increasingly attempts have been made to adapt these to school situations. The problem of interface has been confronted by different applications in a variety of ways. While most retain the one-dimensional format of BASIC programming, some have developed a two-dimensional approach, in which exponents are raised and multiplication is implicit. Such programs include free and shareware algebra tools for the *Macintosh* platform such as *Mathmaster 2.21* and *CoCoA 1.0c*, which both support the entry of numerical exponents using the option key, a feature adopted in the *HyperCard*-based *MathPalette*, developed for this project. This simplified entry process makes such tools readily accessible for younger students, and provides many of the important visual cues which aid in interpreting mathematical statements.

Extending this two-dimensional format are programs such as *ANUGraph*, *Milo™ 1.00* and *Theorist* (all on the *Macintosh*). These programs support the entry of mathematical expressions using menus, templates and palettes, from which mathematical forms may be chosen without the need to learn additional commands or syntactical conventions peculiar to the computer. The palette which became the basis for the *MathPalette* was based upon these principles, allowing entry entirely from visual cues and immediately producing full two-dimensional mathematical formatting. Coupled with simplified keyboard entry (such as the use of the option key for exponents and subscripts, and the “up” and “down” arrows for numerators and denominators of fractions), the palette allows quick and easy entry for both experienced and inexperienced users. Further, by converting the mathematical expression into “text-file” format at the same time, the *MathPalette* allows the user to access other software tools, within which the expression may be “pasted” using the usual *Macintosh* commands. Thus, the *MathPalette* was designed to serve as a common “front-end” for a range of available tools, and to further encourage exploration and the use of multiple representations.

The mathematical tools now available to teachers vary widely in their capabilities, their formats and, most of all, in their costs. The most expensive program considered (*Theorist* from Prescience) costs over $500 (although the Student Edition, selected for use as a principal tool in this project, costs only $105); the least expensive (*MathMaster* and

CC3) are free to be copied and distributed, with a nominal fee requested if the application is to be retained and used.

The most functionally extensive of current mathematical software, *Mathematica* (Wolfram Research), was rejected early. This was not only because it was too expensive (available at between $200 and $300 for the Student version) but also because it will not run on the type of hardware schools are likely to have available. Requiring over 6 megabytes of RAM to operate effectively, its demands are too great for the models most likely to be found in even better equipped school computer laboratories.

Similarly, *Maple V* (Brooks Cole Publishing) was considered and then rejected. Although capable of running (slowly) on the minimal machines likely to be available to schools, and offered in an affordable Student Version (at $150), the interface of this program, similar to that of *Mathematica*, was considered too difficult to support its use across the secondary years. Both packages offer extensive arrays of mathematical commands (*Maple* offers over 1400 commands), but expressions must be entered in one-dimensional form, and specific commands and syntactic forms are required. It was felt that such programs add significantly to the burden faced by students in learning algebra.

The various capabilities of these programs have been summarised in a table (Table 3.1). Applications from only two computing platforms - Macintosh and MS-DOS were considered for the project, since these offered the greatest range of possible software and appeared most prevalent in schools.

Software Summary

	Theorist	Derive	Calculus T/L II	Math-Master	xFunctions 2.2	Math-Palette
Platform	Macintosh	DOS	Macintosh	Macintosh	Macintosh	Macintosh
Cost	$105	$200	$105	Free	Free	$25
Algebra :						
Simplify	✓	✓	✓	✓	-	•
Factorise	✓	✓	✓	-	-	-
Solve equations	✓	✓	✓	•	-	✓
Substitute	✓	✓	✓	✓	✓	✓
Graph						
2 dimensional	✓	✓	✓	•	✓	✓
3 dimensional	✓	✓	✓	-	✓	-
Polar	✓	✓	✓	-	-	-
Parametric	✓	✓	✓	-	✓	✓
Calculus						
Differentiate	✓	✓	✓	-	✓	✓
Integrate (def.)	✓	✓	✓	-	-	-
Integrate (indef)	✓	✓	✓	-	✓	✓
Presentation:						
Text capabilities	✓	-	✓	-	-	•
2D notation	✓	•	✓	✓	-	✓
Other Options:						
Table of Values	✓	•	✓	-	✓	✓
Exact Arithmetic	✓	✓	✓	-	-	•
Complex Arithmetic	✓	✓	✓	-	-	•
Inequalities	•	•	•	✓	-	✓
Matrices	✓	✓	✓	-	-	-
Statistics	✓	✓	✓	-	-	-

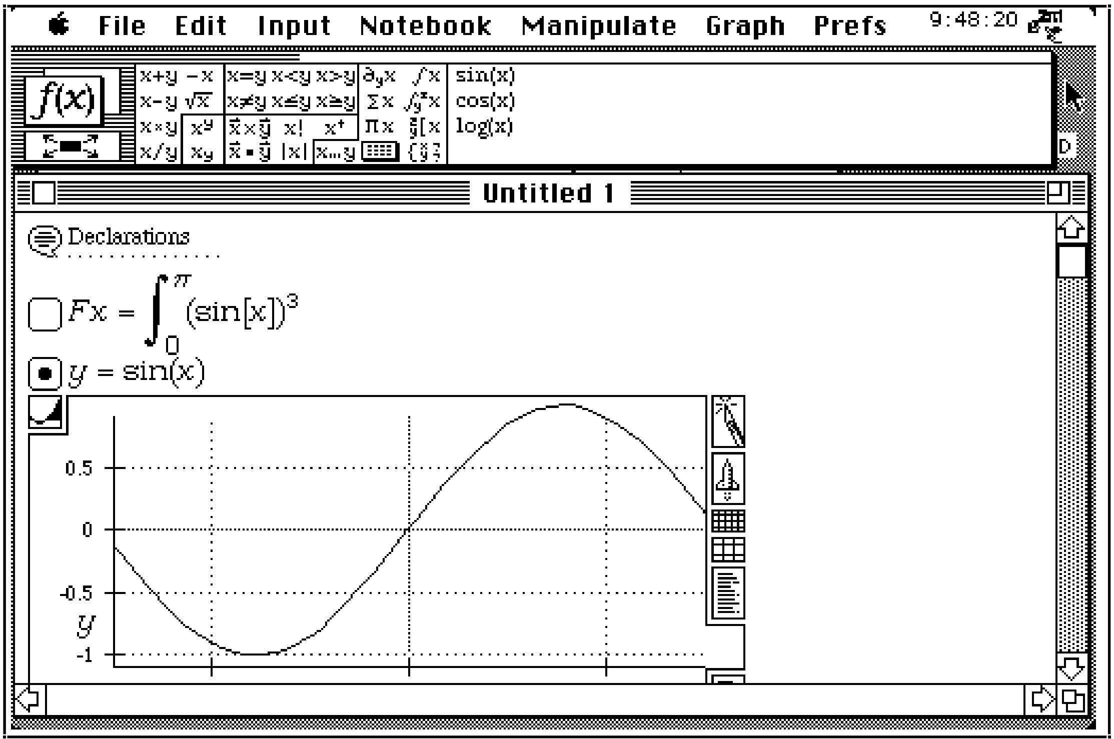
• This indicates that this feature is present in a limited way.

Selected Algebra Tools

Theorist: Almost unique in this field, *Theorist* is a Macintosh program which fully utilises the graphical interface to provide true mathematical notation in a reasonably transparent format. Almost as powerful and extensive in its rule base as *Maple*, this program appears to be a most appropriate package for teaching and learning in secondary schools. Fully menu-driven, with extensive calculus and algebraic capabilities, as well as two and three dimensional graphing and animation, mathematical expressions and equations may be entered by simply pointing and clicking at an available palette, relieving students of the need to learn additional syntactical conventions and commands in order to enter mathematical forms. Files, called “notebooks”, offer a mixture of text, graphics and mathematical forms, allowing the creation of interactive worksheets and exercises by the teacher, and annotated responses and solutions by students.

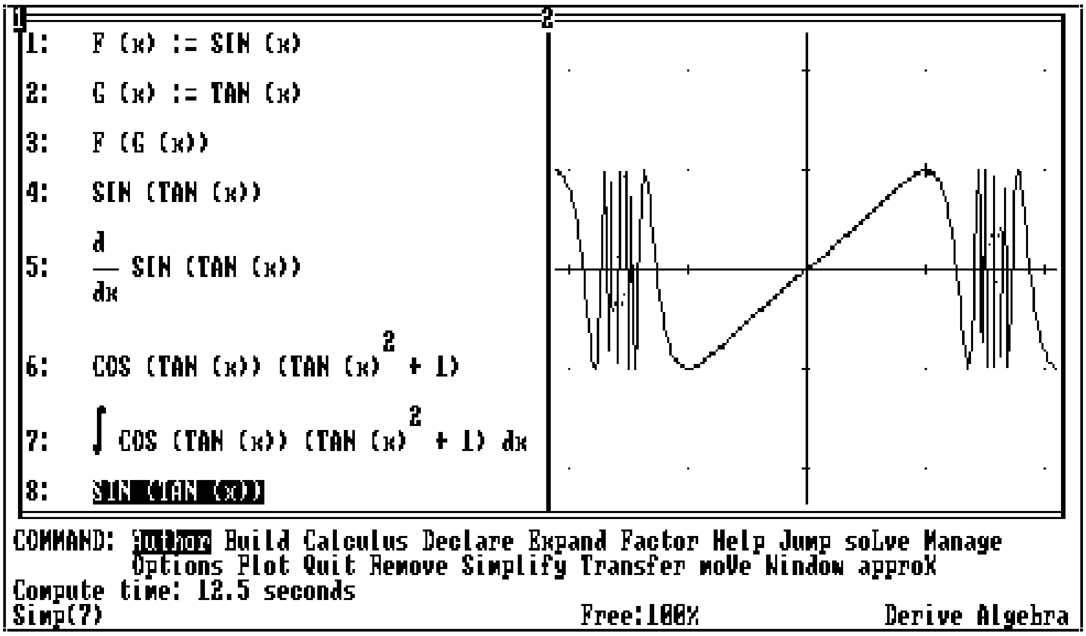
The ability of this program to manipulate terms, solve equations and substitute values into expressions using the graphical interface of the Macintosh is quite unique. Algebraic terms may be relocated by simply “dragging” with the mouse, allowing, for example, equation-solving which physically emulates methods commonly used which involve transferring terms across the equality. Graphs in both two and three dimensions (including relations such as conic sections) are simple to create and edit, and may also be manipulated by hand (dragged and rotated); these may also be animated with ease to produce a moving picture which can be used to illustrate the effects of variable changes. This program offers unprecedented control by the user over the various mathematical representations available - symbolic, graphical and tabular. It was adopted as the preferred tool for the current project. Three copies of the Student Edition were donated for the purposes of this study by the publishers (Thomas Nelson Australia), allowing it to be made available to those cooperating in the gathering of data.

Figure 3.3: *Theorist*



Derive: The MS-DOS equivalent to *Theorist* appears to be *Derive*, the successor to *muMATH* (which was the first serious attempt at computer algebra for personal computers). *Derive* is fully menu-driven, extensive in its mathematical capabilities, and presents mathematical output correctly. Input, however, must be entered in “linear” format, with the advantage of implicit multiplication (enter only 2x - 3, not 2*x - 3 as you need to do with *Maple V* and several others), and simplified (ALT key) commands for √, π, *e* and *i*.

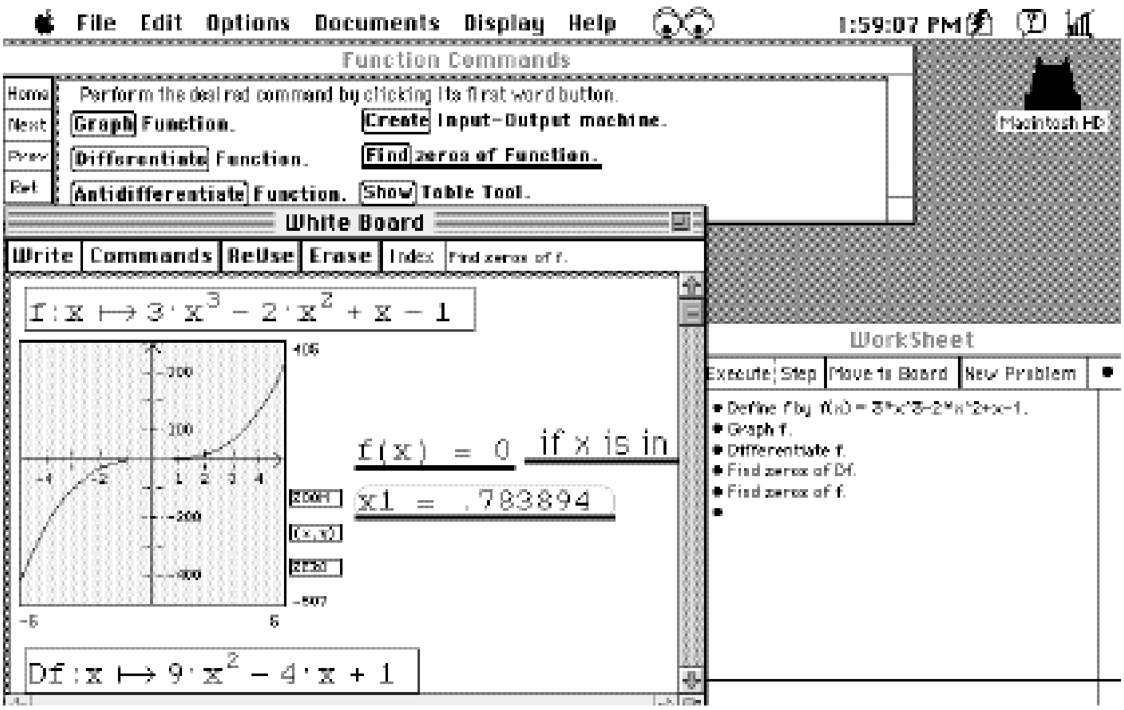
Figure 3.4: *Derive*



While not possessing the ability to freely mix text with graphics and calculations, *Derive* does give the user the option to operate up to eight different screens at the same time. It will carry out all the mathematics required up to University level, and yet is intuitively easy to use. Although expensive for individual copies (around $200), network and Lab prices are more affordable at $1195 for 10 copies networked, and $1395 for a Lab pack. Like *Theorist* on the Macintosh, this was chosen as the recommended tool for MS-DOS users in terms of ease of use and mathematical power.

Calculus T/L II: This powerful computer algebra package offers a “point and click” interface which accesses *Maple*’s algebraic “engine”. Available only on the Macintosh platform, it includes an extensive array of tutorial files, in addition to complete algebraic capabilities and two- and three-dimensional graphing. *Calculus T/L* is structured to support the inexperienced user: selecting any object on the screen provides access to the range of operations and functions which are appropriate to that object. In this way, students confronted by a "blank page" are supported in terms of their possible options, and the uncertainties often associated with problem solving and computer use are minimised.

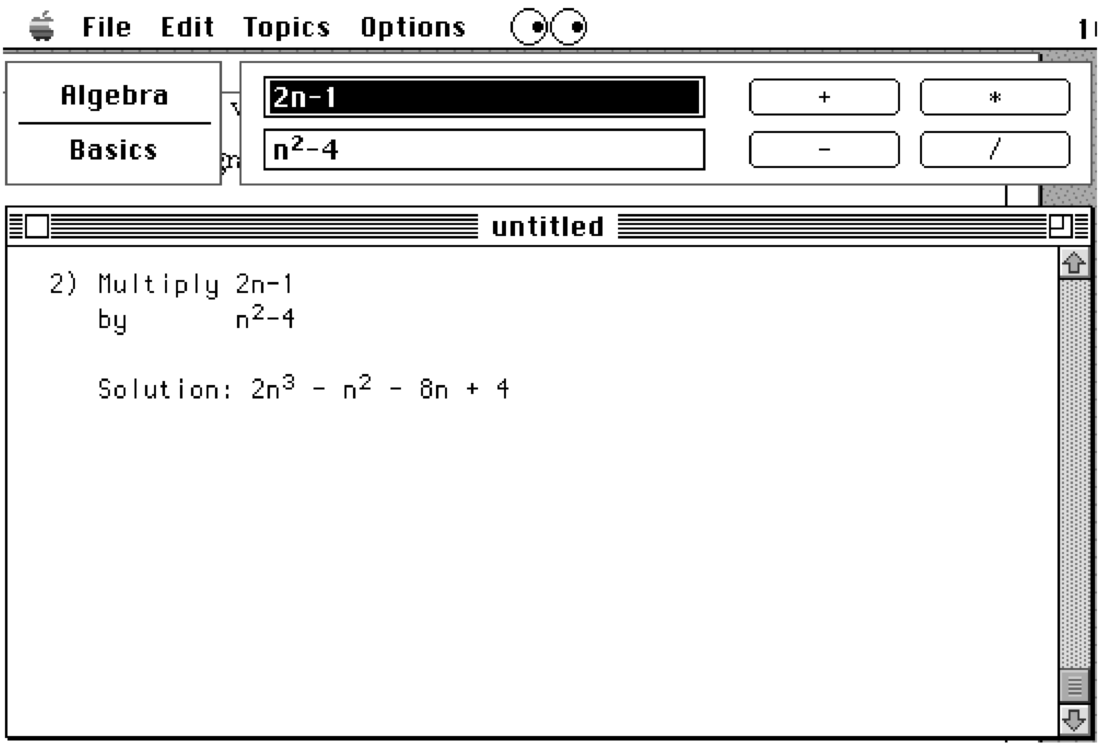
Figure 3.5: *Calculus T/L II*



Respondents using the various software tools found this to be both easy to use and extensive in its mathematical capabilities. It became the preferred option for some, even over a program such as *Theorist*, with its superior formatting and presentation.

MathMaster 2.21 : Originally released in 1987 as "shareware", this program was withdrawn from circulation by the author, who did not wish to continue upgrading it to newer Macintosh versions and models. After some correspondence, he was persuaded to allow it to be distributed freely for educational purposes and, in particular, to be used for the current study. This is an excellent program for basic algebra and co-ordinate geometry from junior to senior years. Its capabilities include operations on numbers and polynomials (from +, -, x and ÷ to greatest common factor and lowest common denominator), simplification of algebraic expressions, solution of linear equations and inequalities (both algebraically and graphically), co-ordinate geometry (graphing and solving linear equations and inequalities in two variables, as well as finding equations given intercepts, slope, points and so on).

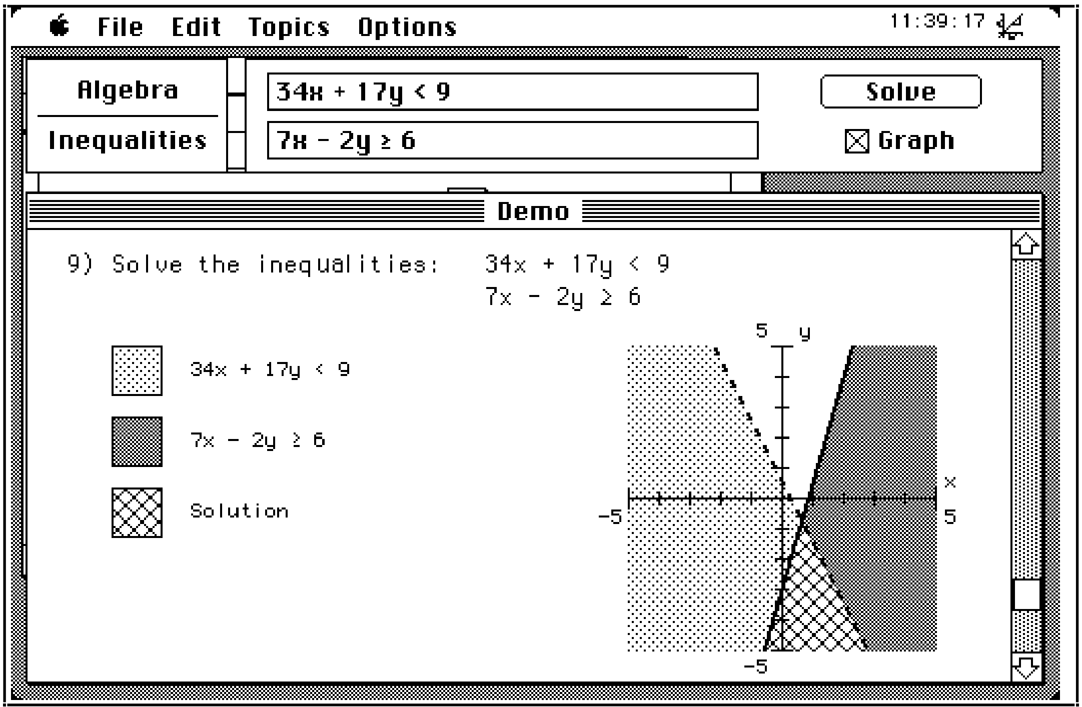
Figure 3.6: *MathMaster 2.21*



The Macintosh interface is used to advantage (including the ability to place exponents in position by using the option key with the required number). Thus an expression such as $\mathbf{3x^4y^2 - 2xy^3}$ may be entered quickly and directly. Worksheets may be created, and the results are presented with appropriate comments to describe the process involved.

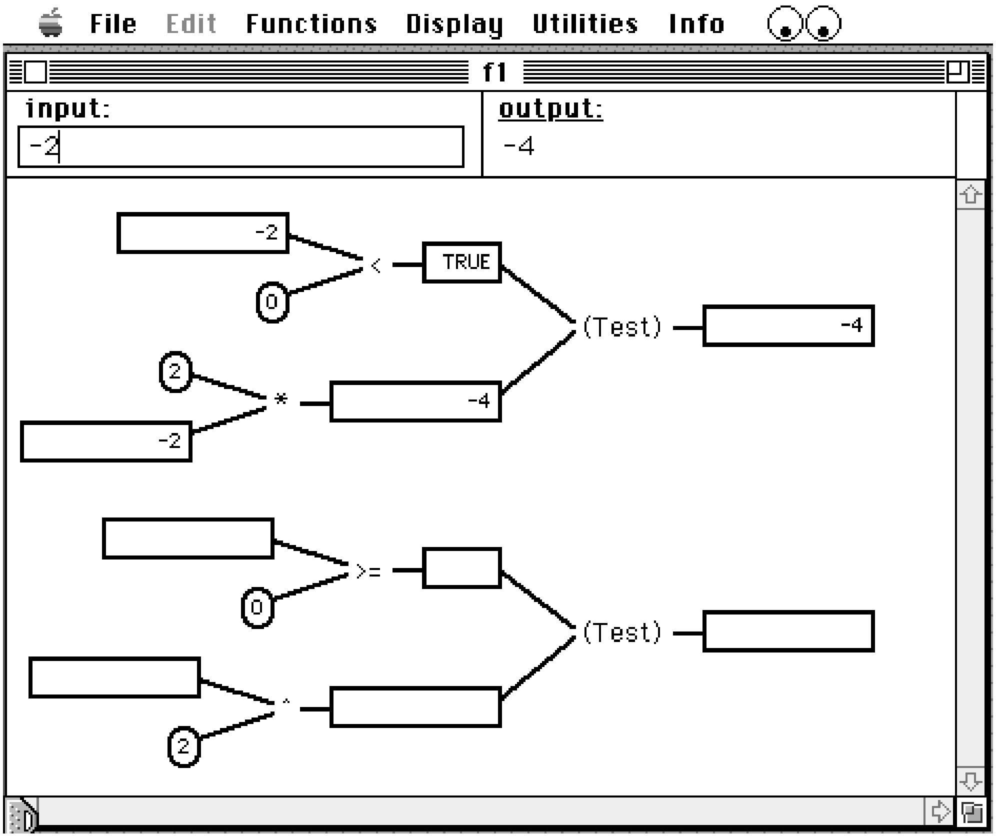
Although limited in its mathematical functionality in comparison with those programs described above, *MathMaster* offers an appropriate interface and basic algebraic capabilities which make it an attractive tool for schools which often cannot afford site license versions of the commercial programs.

Figure 3.7: *MathMaster*: Solution of Linear Inequalities



Additionally, *MathMaster* offers some particular advantages over the more powerful algebra tools. Algebraic expressions, for example, may be entered term by term, with students selecting the appropriate operations which will link them. This forces students to “reconstruct” algebraic forms from the visual printed original, which may often be perceived superficially. The coordinate geometry and equation solving features, too, provide a quick and convenient means for verifying solutions and exploring alternative forms.

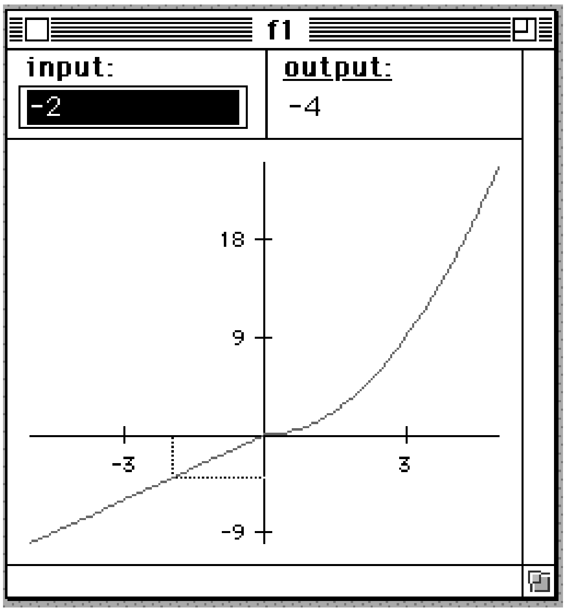
Figure 3.8: *xFunctions 2.2*



xFunctions 2.2: This multiple representational tool offers extraordinary versatility in a program freely available for educational purposes. Functions may be entered in three forms - as expressions (using the usual linear form, but allowing "split-domain" or "piece-wise" functions to be easily defined), as graphs (where the user actually creates the graph by clicking and dragging on a pair of coordinate axes), and as a table of values (entering the function values as x- and y-coordinates). Once entered, functions may be viewed in any of five representations: as expressions, graphs, tables, input-output boxes or as "diagrams", illustrated below. Figure 3.8 displays the "diagram" for the function defined by $y = 2*x$ for $x < 0$ and $y = x^2$ for $x \geq 0$. This unique ability to expose the process of a given function appears to offer much in

assisting students to develop versatile understandings of function concepts.

Figure 3.9: *xFunctions* Graph

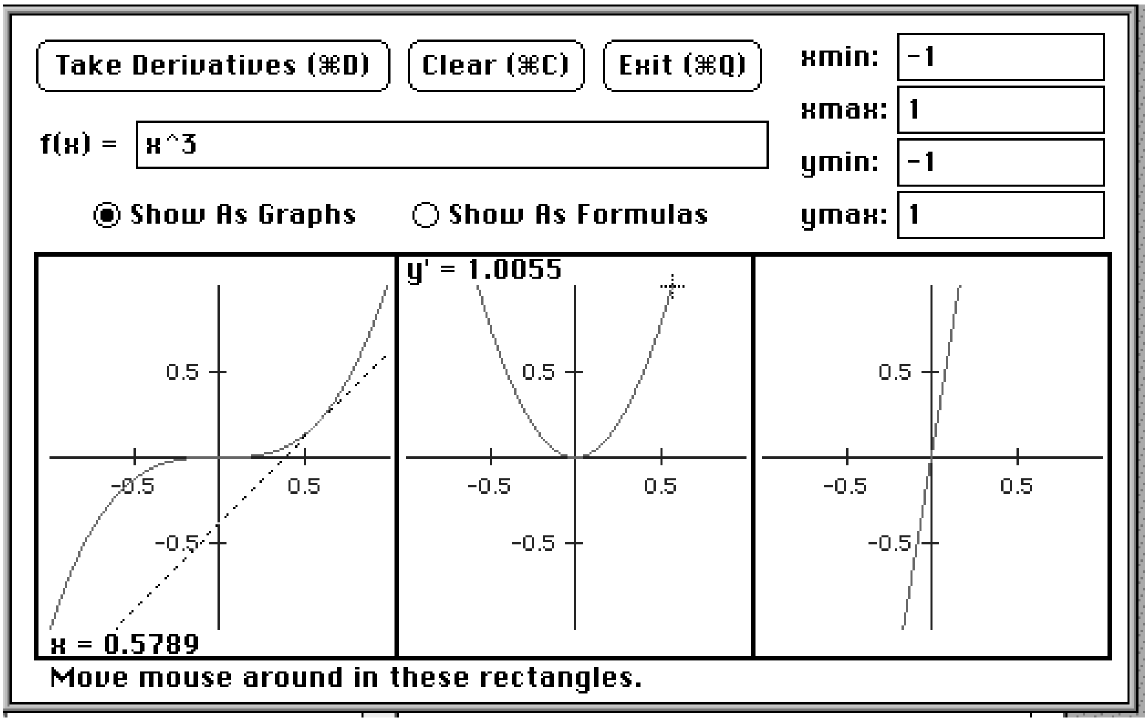


The graphing capabilities of this program are fast, clear and effective, allowing ready access to this important representation, while at the same time encouraging them to go beyond and explore other forms of the function under consideration.

In addition to the input and output capabilities of this tool, a range of mathematical features are also available, including three-dimensional plotting, animation of functions, derivatives, areas under curves, parametric forms and differential equations. This is an exceptional package for teaching and learning function concepts at all levels.

The *Derivatives* option illustrated in Figure 3.10 is typical of the capabilities of this program as a teaching and learning tool. In addition to the function, first and second derivatives are displayed, and tangent lines and points are available interactively using the mouse. Additionally, first and second derivatives are available in symbolic form, if desired.

Figure 3.10: *xFunctions* Derivatives Option



One of the few problems associated with this program is that of entry of mathematical forms. Once again, the user is forced back to one-dimensional input. It was as a particular response to this problem (and the desire to make use of the capabilities of this and other suitable programs) that consideration was given to developing a "front-end" - a program which would allow mathematical expressions to be entered easily without use of specialised syntax or commands, and then capable of accessing other software tools and "pasting" the expression in. The result was the *MathPalette*.

Conclusion

While a number of other software tools were utilised during the course of the project (including the calculator, *PCalc*, the tutorial package, *Are*

You Ready for Calculus?, the unique interactive geometry package, *Cabri-Geometrie, CoCoA* (a commutative algebra tool which offers useful facilities for expanding and simplifying polynomials, substitution and fraction capabilities) and the version of *LOGO* developed by the University of California Berkeley) the packages described above served as the principal algebraic tools for data gathering from students, teachers and student teachers over the two years of this phase of the project. When combined with the *HyperCard* modules developed for the study, participants were provided with both context and available tools for exploring algebraic ideas and skills, and the means by which such interaction might be observed. The modules, forming the package *Exploring Algebra*, are described in the following chapter on the research design. The development of the supporting multiple representational tool, *The MathPalette,* in response to interactions of participants with available tools throughout the course of the study occupies a significant part in the subsequent description of data gathering and results,