Classification and analysis of some computer software packages for teaching Mathematics

by

Fhatuwani Samuel Netshapala

Submitted in partial fulfillment of the requirements for the degree

Master of Science: Mathematics Education

in the Faculty of Natural & Agricultural Sciences

University of Pretoria

PRETORIA

October 2001

Acknowledgments

I would like to express my sincere gratitude to my supervisor, Professor J.C. Engelbrecht and my co-supervisor Dr A.F. Harding, for their tireless guidance, patience and encouragement in completing this dissertation.

Many thanks to my work colleagues for their support throughout. Special thanks to my wife Jennifer and my daughter Vhulenda for having endured my divided attention.

Summary

Classification and analysis of some computer software packages for teaching Mathematics

by

Fhatuwani Samuel Netshapala

Supervisor	:	Professor J.C. Engelbrecht
Co-Supervisor	:	Dr. A.F. Harding
Department	:	Mathematics and Applied Mathematics
Degree	:	Master of Science: Mathematics Education

There exists, at present, a large number of software packages that are specifically designed for the teaching of mathematics. Yet many schools, especially in rural areas, have not been exposed to the new technological developments. Using a mathematical software package is therefore an unknown concept for many teachers. It is our belief that as time progresses, more schools will become aware of the possibility of using technology and software for teaching mathematics. Teachers who embark on such a route will be in need of knowledge and guidance. This dissertation aims to address such needs. In this dissertation, the advantages and disadvantages of using software packages in teaching mathematics are discussed, a classification of software packages is developed and a number of the software packages are listed and described. We do not attempt by any means to give a complete list of available mathematical software packages, but we do, however, give some examples in each of the categories.

My personal experience

I started school in 1978 when I was seven years old. I attended a school in one of the villages around Thohoyandou in the Northern Province of South Africa. The first time I touched a computer was in 1991 in my second year of BSc (Education) at the University of Venda for Science and Technology. The first thing that came to mind was to press the keyboard. Unfortunately the computer was dead. Nevertheless, I was so proud and happy about the experience that I took a picture of the computer. It was a memorable experience for me.

I managed to graduate even though I was at a disadvantaged university with poor facilities and infrastructure. I now see that by not having had experience of computers in my studies, I have missed a lot. Our lessons were very theoretical and we had to visualise everything mentally. In mathematics, we used to draw graphs by hand. Some were complicated and seemed a waste of time. Some were even impossible to draw, especially those in three dimensions. In my opinion we wasted time evaluating definite integrals and solving complicated systems of linear equations without computers.

These days a large number of school learners are at a great advantage because they are able to use computers, even before Grade 1. They are also able to do many complicated tasks in mathematics using computers. Their lessons are no longer imaginary and they can draw almost any graph with the aid of the computer.

CONTENTS

Chapter 1

INTRODUCTION

Computers have become a part of everyday life. They have taken an important position in our lives and probably will continue to do so in future. It is not a question of whether computers should be used in teaching, but rather how. In today's society, it is irresponsible not to expose learners to computers. We have to prepare learners for the life in which computers will play a very important role.

Computers have an important role to play in the classroom, just as they have in many other areas of daily life. They can, for example, be effectively used to demonstrate a concept or technique in class by means of an appropriate computer software package. But the instructor has to acquire the necessary software package and the knowledge of its use. In order to make a decision on which software package to acquire, the instructor requires a description of how a package can be used in the classroom and how effective it is likely to be. Information on usage can be obtained from individual distributors, but information on how to successfully implement it in the classroom is not so easy to come by. A compilation and classification of available software packages and how they can be used would thus be of valuable use. One of the aims of this dissertation is to make a contribution in this direction.

In this dissertation, instructors refer to both teachers at secondary schools and lecturers at colleges and universities. Learners refer to students at secondary schools, colleges and universities.

Computers can be used in many ways in the classroom. They can, for example, be used to teach. facilitate the study of traditional content-area topics. provide opportunities for learning how to use technology in a mathematical content or to supply learners with general purpose tools for performing academic tasks more efficiently. According to Grabe [12. p12]. "these distinctions are similar to what is called the tutor. tool and tutee model". A computer application could be designed specifically to teach learners in the role of a tutor. Computer programmes are then directly responsible for instruction. More general-purpose applications. such as word processing. are designed to help learners function more productively in the role of a tool. They make academic tasks such as writing and calculating easier. When functioning in the tutee role. learners instruct computers by writing programmes. Discussion in this dissertation will be based on the computer being used as a tutor.

Not all instructors are in favour of using computer software packages and such instructors are quick to mention disadvantages. Some of these opinions are influenced by prejudices and fears and are not always objective. Similarly. other instructors are so much in favour of using software packages that they are reluctant to admit that there are disadvantages. It is true that advantages and disadvantages are valid. but it is probably also true that the advantages outweigh the disadvantages. Computers are more appropriate for some uses than others. and are more suited to some teaching styles than others.

In the second chapter. we will list some of the advantages and disadvantages of the use of computers in the classroom for both instructors and learners. We will construct a classification of packages according to the style of teaching they provide in the third chapter. Each category supports a different classroom strategy. This categorical identification will help to illustrate the point that not every strategy is supported by every software package. Instructors need different software packages to accomplish different objectives associated with a given curriculum. No single software package can address all classroom needs.

Classifying software packages by category of use is a step towards learning how to apply the right package to a given task. The first category consists of those packages that can be used when the instructor is in control of the class. This is followed by categories of problem-solving packages. graphical packages. tutorial packages. drill and practice packages. game packages. computer algebra systems and of simulation packages. We list a number of available software packages for each of these categories and give a short description of each and the level at which they are suitable. A more detailed description of some of the software packages is given in the appendix. We also discuss the way in which each category of packages can be integrated into the teaching process as well as advantages and disadvantages to both instructors and learners.

Integrating computers into the teaching process means using the power and ability of the computer to aid learning. Instructors must identify what the learning goals and objectives are and then identify the appropriate software packages which could assist them to accomplish these goals. This sounds simple. but in practice it is considerably more difficult to achieve. Decisions on what to do. how to do it. and the task of convincing other instructors to do it can be formidable. Integrating computers into the teaching involves change. It changes the ecology of the school and affects funding. teaching methodology. curricula and timetables to name but a few changes. This integration does not refer to computer literacy or computer awareness. It refers to using the computer where it is the best medium to support learning goals. The entire school community of learners. parents. instructors and administrators have to accept that computers are part of everyday school life.

Before integrating computers into the teaching process. instructors must introduce chosen computer programmes to learners. Learners need a demonstration and hands-on instruction to make them feel comfortable and to develop the motivation for tackling assignments that require a software package. Towards this end. instructors could prepare instructional handouts explaining key features of the programme. Instructional videos could also be used when it is desirable for learners to learn independently.

One should keep in mind when introducing a new software package. that it may be necessary to have a thorough knowledge of the four stages of instruction similar to when introducing a new topic in class. Grabe [12, p79] suggests that, "these stages are: presentation of information, initial guidance as the learner struggles to understand the information, extended practice to provide fluency, and the assessment of the learner's learning". These stages are intended as a general description of components of instruction, and are not specific to instruction delivered using a computer software package. However, awareness of these stages of instruction can be used to identify and differentiate purposes of different categories of software packages. Awareness of these stages can enable instructors to integrate a particular software package into the teaching process at the right time.

Chapter 2

Advantages and disadvantages of using computers in the mathematics classroom

2.1 Introduction

The introduction of computers into education can be both advantageous and disadvantageous for both instructors and learners. Although introduction of computers is often a positive experience. there are also problems and dangers in using computers in the classroom. Any instructor who considers the introduction of software packages into the classroom should be well aware of these. and should also be able to distinguish between valid concerns and prejudices.

2.2 Advantages of computers

2.2.1 Visualisation.

Williams [41. p1] says that. "computers offer a potential for visualising complex and abstract processes". Thus. by using computers. learners can visualise mathematical concepts through graphical representations. This may enable learners to understand concepts better. which could in turn lead to their feeling successful in learning the course material. and hence develop a positive attitude towards mathematics. Walker [34, p652] is of the opinion that, "visualisation of mathematical phenomena increases the rate of assimilation, learners acquire basic knowledge more quickly and develop mathematical intuition". Successive graphs in animation style can improve understanding further. It is said that a picture is worth a thousand words. It is now easy to see what has before, in many cases, only been spoken of. The presentation of information in a pictorial form may leave learners free to put their own structure on that information, so that they can learn it in their own way.

2.2.2 Calculations and manipulations

Learners can make complex calculations quickly and accurately using computers. The burden of tedious calculations then is no longer a threat. This is especially true in processes where a computer can do impressive successive calculations in virtually no time. In situations where a lot of standard calculations have to be done before learners come to a conclusion, the use of computers can be very effective. Even if these calculations can in principle be carried out by hand, the fact that they are time-consuming and may lead to calculational errors may be demotivating, and may obscure the real aspects of the problem. Where previously, the concept often got lost because it was obscured in calculations, this no longer needs to be the case. Quoting Kelman [20, p3], "learners can probe problems, store and retrieve information, test out solutions and calculate results". According to [45, p1], "the power to compute rapidly, to graph relationships instantly, and to systematically change one variable and observe what happens to other related variables help learners become independent 'doers' of mathematics".

2.2.3 Patience

Instructors often work under restrictions such as large groups and a work overload. They could thus find it difficult to grant repeated explanations. There is a danger of learners taking this as a personal rejection which could in turn affect their self-confidence. On the other hand, the learners may not want another long explanation, but may want more time to explore and assimilate the new concepts.

The learners may also not want to admit that they still do not understand. In this regard. the computer has all the time and patience necessary. Computers. contrary to instructors. are available all the time and at any time the learner wishes to work. They are never tired and never 'lose their temper'. The learners may repeat the same activity many times and so improve their understanding. Engelbrecht [11] says that. "with its infinite patience the computer can assist in bringing home the use of certain techniques". It is. of course. necessary to have the right software for the circumstances.

2.2.4 Logical thinking

Bajpai [3. p83] suggests that. "one approach towards using the computer requires learners to write their own programmes". This can be a useful exercise as it encourages learners to think logically. and reinforces their understanding of the method or algorithm. One only develops an understanding of an algorithm or method when one has to write a programme for it. This forces one to take ownership of the method and to instruct the computer on how it should 'think'. It is also true that the skill of programming has become less common because of all the excellent software packages available. These packages are good for saving time. but nevertheless. the value of programming should not be underestimated.

2.2.5 Individual instruction

Engelbrecht [11] states that. "computers can handle each learner individually". This increases the rate of learning by allowing learners to proceed at their own pace. This may promote self-confidence because it gives the learners a feeling of control over what they are learning. Instructors are able to determine the weak points of the learners and pay attention to the specific weak points of the learner concerned.

2.2.6 Enhanced problem-solving skills

According to Dubinsky [9], "the use of computers as a thinking aid and an intellectual tool enriches learners' mathematical exploration, facilitates learners' growth of mathematical understanding, and improves their problem-solving skills and concept development". When solving problems with computers, learners may be given a chance to focus on the process of problem solving instead of focusing on the answer. Learners may be able to explore problems and not simply obtain answers. By exploring they may reveal a range of solutions rather than a single answer. Quoting Bishop [6, p243], "access to numerical computation tools enhances learning and extends the problem-solving power of most learners".

2.2.7 Enjoyable learning

A computer is seen by many learners as a source of entertainment. This may be because there are so many computer games that attract learners. Most of the learners do indeed seem to find pleasure in using the computer. Ruthken [27, p279] maintains that, "learners demonstrating high computer motivation find that computers make learning more enjoyable, and will spend more hours at a computer to complete a task. They enjoy testing out new ideas on the computer".

2.2.8 Fear of exposure

Any verbal discussion normally requires good eye contact, a necessary social skill, between the parties involved. Ball [4, p74] is of the opinion that, "learners who do not feel confident about the topic being discussed could feel exposed and so could feel threatened by too much eye contact". Since the screen presents images around which the discussion is centred, the problem of eye contact between the instructor and the learner no longer exists. Learners can approach the topic without fear of exposure.

2.2.9 Improved understanding of geometry and trigonometry

Geometry is traditionally a difficult topic to master. The use of appropriate software packages could aid in the process of understanding. Ball [4, p26] says that, "excellent software packages are available to help learners master geometry and trigonometry by moving and changing angles, recognising and combining shapes to appreciate properties". Such packages offer the opportunity for experimenting which makes the subject more alive. This is of course also true for many other topics in mathematics.

2.2.10 Immediate feedback

When working problems using pen and paper, learners may make repeated mistakes before discovering that they have problems. Learners may get immediate feedback when using a computer and mistakes may be immediately brought to their attention. This may keep learners from wasting a lot of time by avoiding repetition of the same mistake.

2.2.11 Remediation

Engelbrecht [11] states that, "the computer can do a thorough, objective diagnosis and remedial teaching of specific learning problems in a scientifically dependable manner". The computer can do so by letting learners repeat the same activity several times, at the learner's own pace and in his/her own time.

2.2.12 Utilisation of instructor time

Engelbrecht [11] is of the opinion that, "the computer relieves the instructor of the burden of correcting numerous work problems, and gives him/her chance to respond to learners immediately". This gives the instructor the opportunity to utilise his/her time more effectively by attending personally to learner problems. This also frees the instuctor to do the important work of human interaction.

2.3 Problems, dangers and disadvantages of computers

2.3.1 Cost

According to Klooster [22. p1]. "education budgets consist mainly of salaries. building and administration costs. with little expenditure being made on equipment". It is true. therefore. that schools may not have sufficient funding to set up an effective computer lab. It is also true that the issue of cost can be used as an excuse for not embarking on computer-assisted instruction. This excuse could hide a dislike of technology or inexperience. There may be a way out on the issue of cost. Schools may embark on fund-raising. If members of the community are aware of the importance of computers. they may be willing to assist.

2.3.2 Threat to novices

Williams [41. p2] is of the opinion that. "computers may be threatening to novices". Instructors who are exposed to computers for the first time may feel threatened if they find themselves in a situation in which they have to use computers in their classrooms. They know that they may make mistakes at any time and may not like such an embarrassment. Because of this. they may decide to abstain from using computers.

2.3.3 Lack of free dialogue

When learners are working on the computer. there may be a lack of human interface. and hence no free dialogue between the learners themselves. Walker [39. p487] maintains that. "there is no free dialogue between the learner and the computer". The computer is unable to answer questions from the learner who asks for explanations on difficult points. In this case. the presence of the instructor is necessary in order to clear up any misunderstanding between the learner and the computer as well as to overcome difficulties in understanding concepts.

2.3.4 Dependence

Learners may become dependent on computers. They may see the software package as an end in itself rather than as tool to be used to further their understanding of the topic. They may not want to do any task in the absence of the computer. Burns [8. p1] is of the opinion that. "computers may also eliminate basic skills such as spelling. simple arithmetic. and grammar".

2.3.5 Lack of documentation

There is a large number of computer software packages available on the market. According to Williams [41. p2]. "it is time-consuming for instructors to discover what is available and to have to choose packages that they feel are worth using". It may be possible that software packages selected by instructors do not have clear operating instructions for use in the classroom and do not run on available computers. To minimise this problem. instructors should consult the relevant people with information on how to operate a particular software package. Instructors can. for example. consult software companies or other instructors. It may be useful for instructors in a particular area to exchange information about packages they have used.

2.3.6 Substitutes for books and instructors

Computer packages should not be seen as substitutes for books. Williams [41. p2] maintains that. "if learners are required to read and analyse substantial texts. they should be given a book". Many learners may not prefer spending hours reading text on the computer screen. When the computer is used in the teaching process. one must always remember that the computer is a piece of equipment. an aid. and that the role the computer plays in the learning process is always of secondary importance to the real aim of learning. Quoting Engelbrecht [11]. "the task to familiarise the learner for the first time with concepts still remains the instructor's privilege and duty. and must under no circumstances be substituted by a computer system".

2.3.7 Distraction

Walker [36. p425] argues that. "if the computer comes in action too early. it may distract learners' attention from the mathematics and it may result in an attitude of pushing the button to get the answer". So. instructors should integrate software packages in the right way at the right time. When novice software developers design software packages. they may use multiple media elements simultaneously in the belief that more variety leads to better learning. The danger here is that some elements may act as distractors. For example. if learners are presented with concurrent text. audio and video elements. they may not know which media element to pay attention to. Some learners may concentrate on video and some on audio. which may distract their attention from the aim of the software package.

2.3.8 Impersonal

Computers are impersonal. Burns [8. p1] maintains that. "there is a lack of physical interaction between the learner and the computer". Learners may get frustrated when working with the computer if the need arises to discuss a specific point with another learner. Learners may feel alienated because computers cannot meet their emotional needs and cannot motivate them to love learning.

2.3.9 Inexperienced instructors use it wrongly

When instructors have to use computer software packages for teaching. they need to gain experience similar to what they need when using any other aid. If they are inexperienced. they may not know when to integrate a particular software package in the classroom. They may. for example. rely on software packages for passing all the information they need on to their learners. which is typical of the wrong usage of computers in the classroom. Computer software packages need an experienced instructor for them to be used effectively.

Chapter 3

CLASSIFICATION OF SOFTWARE PACKAGES

3.1 Introduction

In this chapter, we construct eight different categories of instructional softwares and discuss the characteristics of each of these categories. There is a large and growing number of computer software packages available to support the teaching and learning of many topics in mathematics. These packages can be used in a variety of ways, and a classification should be useful. One simplistic way of attempting to classify software packages is to label them as good or bad. One of the problems in doing this is that it is difficult for instructors to agree on what makes a good software package, and the classification is then subjective. Another way of classifying software packages is by the topics they teach. The problem here is that while many software packages are designed to teach just one topic, some panoramic packages support the learning of a large range of topics. It is difficult to give a complete list of these. A different way of classifying software packages is by the style of learning they support. This is how software packages in this dissertation are classified. One package may provide a game to be played, for example, while another may best be used by instructors to illustrate points they want to make while teaching.

3.2 Exposition packages

Exposition packages are used by the instructor to demonstrate or explain an idea or a technique to the whole class. The use of these packages stimulate mathematical discussions in the classroom between instructors and learners, and between learners themselves. A sufficiently big monitor or data projector is required so that all learners can see clearly. Here the computer is used as a visual aid. In this category, the instructor may ask learners many questions, and he/she remains in control of the sequence and the rate of teaching. The instructor decides on what idea is to be presented next, how and for how long it needs to be discussed before the next idea is introduced.

Table 3.1. Examples of exposition packages.

Package	**Level**	**Description**
Spring and string	Tertiary	Ordinary differential equation software.
Complex numbers	Tertiary	Evaluate complex expressions.
Graph theory	Tertiary	Allows learners to experiment with undirected graphs.
Integral	Tertiary	Computes the numerical value of a definite integral by various techniques.
Sequence	Tertiary	Creates sequences and shows values of successive terms numerically and graphically.
Lepack	Tertiary	Sets realistic problems involving the solution of a large system of linear equations.

Table 3.1. Continuation.

Package	**Level**	**Description**
Moves	Secondary	Teaches geometrical transformation-rotation, translation, reflection and enlargment.
Singos	Secondary	Provides a model that explains the meaning of sines and cosines of all angles.
Solve	Secondary	Explains the solution of linear equations.
Discrete mathematics	Tertiary	Discrete mathematics package.
Phaser	Tertiary	Differential equations package.
Macmath	Tertiary	Solves first-order differential equations.
MATT graphical display package and differential equation solver	Tertiary	Solves differential equations through graphics.
Maclogic	Tertiary	A first-order logic software package.
Solving quadratic equations	Tertiary	Supplements the teaching of the concept of solving quadratic equations.
EZQ	Tertiary	Solves first and higher-order differential equations and systems of equations.
Math 246 **programmes**	Tertiary	Ordinary differential equations software.
Matrix calculator	Tertiary	Manipulates matrices.

Table 3.1. Continuation.

Package	Level	Description
Matrix master	Tertiary	Inputs. stores. inserts or deletes a row of a matrix. It performs elementary row operations.
Ordinary differential equations	Tertiary	Solves first and second-order differential equations.
Differential equations graphics package	Tertiary	Solves first and second-order differential equations.

To integrate this category of packages into the teaching process. the instructor must first teach the content in class using the traditional method. and can then use a computer software package to further illustrate it. Alternatively. the instructor can use the package from the start to introduce a concept. Here the instructor keeps full control of the educational programme. The instructor can use the package for any concept that he thinks can best be illustrated by the package. By being exposed to visual illustrations of the mathematical concepts. learners may be able to understand these concepts better. This may lead them to feeling successful in learning the course material. and thereby cultivates a more positive attitude towards mathematics. According to Walker [39. p487]. "the capability of seeing the results obtained on the screen is considered to be an evolutionary step in the strategy of learners' learning. Learners can see the cause of error. even in an erroneous action. and this can lead them to another presentation of the solution".

Walker [36. p419] maintains that. "introducing these packages into the teaching process requires a considerable re-arrangement of the classroom activities and requires a lot of time". There is the danger that instructors may not want their jobs usurped. and since these packages aim to mimic human teaching activities this is an understandable fear. Another danger is that instructors may have little time or incentive to keep up to date with developments in computing, and feel unable to make proper use of these packages. Graham [13, p3] is of the opinion that, "instructors do not want their normal routines disrupted by, for example, the need to be responsible for the scheduling of the use of these packages".

3.3 Problem-solving packages

Quoting Wiebe [40, p17], "a problem is a situation demanding resolution for which there is no immediate solution, and problem solving is a process of searching for and finding solutions to problems". Problem solving consists of moving from a given initial situation to a desired goal situation. A different way of saying this is that problem solving is a process of designing and carrying out steps to reach a goal. It serves as a preparation for the real world because it teaches learners the necessary skills for dealing with problems they are bound to be confronted with. These packages help learners to search for a solution to a problem. Kelman [20, p5] maintains that, "problem-solving packages help learners to explore problem situations, keep track of what they have learnt and find patterns in what remains to be done". They enable learners to attack a problem using different strategies. Problem-solving packages create practice situations that help learners to gain skills in the process of selecting a strategy for solving a problem.

Table 3.2. Examples of problem-solving packages.

Package	**Level**	**Description**
Geometric supposer	Secondary	Does geometrical constructions.
Math pac	Secondary	Provides review and interactive problem solving in algebra.

Table 3.2. Continuation.

Package	Level	Description
Visicalc	Secondary	Software for topics such as functions and equations.
TK solver	Secondary	Solves somewhat complex problems. This is an automatic problem solver.
Mind over matter	Secondary	Helps develop problem-solving skills.
Wizard	Primary	Gives learners practice in basic mathematics.
Guesstimation	Primary	It provides practice in numeration and estimation.
Mathpert	Tertiary	Allows learners to focus on correct strategy of problem solving in calculus and algebra.
Geometric constructor	Tertiary	Does geometrical constructions.

Problem-solving packages may be used when learners have difficulties in selecting the best strategy to solve a problem. Learners are given a new-found freedom to explore. to test strategies. and to play. all of which are at the heart of problem solving. According to Kelman [20. p3]. "problem-solving packages teach learners to think. and hence prepare them for the real world". They enable learners to work with challenging and complicated problems. Learners engage in a kind of problem solving in which the process. not the product is paramount. They may come to recognise that there are often many correct answers for a given problem. The correct answer becomes less important than the process of exploring the content of the solution.

Instructors can create complex problem -solving environments in which learners can act more like real-world problem solvers. They enable learners to solve the standard homework problems in less time. thereby allowing the instructor to cover more material or the same material in more depth. Williams [42, p2] says that, "problem-solving packages reinforce the need for understanding mathematical concepts and mastering computational skills". They enable learners to develop and apply problem-solving strategies to geometry, logical reasoning, classification, measurement, fractions and decimals, and other mathematical content. Strategies include using trial and error, making an organised table, drawing a diagram, and looking for a pattern.

Howard [18, p1] says that, "when using problem-solving packages, learners enjoy solving word problems easily and dependably. They may become happy campers in the forest of problem solving. Instructors may also gain the advantage of a logical, consistent approach to word problems that applies across the curriculum and at all grade levels". These packages enable learners to experience success in solving problems. Learners may thus gain confidence in doing mathematics and develop persevering and inquiring minds. They may also grow in their ability to communicate mathematically and develop a point of view about what it means to learn mathematics and to solve problems in mathematics.

3.4 Graphical packages

Kelman [20, p95] says that, "mathematics education has for some time been dominated by numbers while graphs and geometric interpretations have played a secondary role". This is because very few instructors are used to or familiar with visualising mathematical concepts by geometric interpretations. Graphical packages promise to change all that. For this to happen, mathematics instructors as well as their learners, need experience with graphical packages. A graphical package is a package that contains a variety of tools which allow learners to create and design pictures and graphs on the screen, and to print them. There are now a variety of graphical packages available for special applications that may be of interest to instructors and to learners of mathematics. Packages that display data easily on the screen and that graph mathematical functions can be used in a variety of ways in mathematics classrooms.

Table 3.3. Examples of graphical packages.

Package	Level	Description
Turtle graphics	Secondary	Draws complex designs.
SIGAD	Tertiary	For learning three-dimensional geometry.
Locus	Secondary	Draws loci on the screen.
Graph	Secondary	Draws graphs.
Function graph plotter	Secondary	Draws graphs.
Build	Secondary	Draws pictures composed of cubes.
Algebra and function plotter	Secondary	Helpful in the study of functions.
Interpreting graphs	Primary and secondary	Gives practice for matching line graphs to description of events in the physical world.
Taylor	Tertiary	Graphs the first 20 Maclaurin polynomials.
Polar	Tertiary	Plots polar equations of the form $r = r(t)$. t is an angle.
Quick-Graph	Tertiary	Enhances the teaching of mathematical functions.

Table 3.3. Continuation.

Package	Level	Description
Gyro graphics	Tertiary	Provides interactive real time animated 3-dimensional graphing of mathematical objects: Surfaces, space, curve, vectors and vector fields.
Geometer's sketch pad	Secondary and tertiary	Offers dynamic visual images to help learners' understanding of the topic.
Graphic calculus	Tertiary	A graphical programme numerically based.
Master grapher	Tertiary	Explores fundamental concepts related to 2-dimensional functions.
Plot pak	Tertiary	Graphs discontinuous functions and straight line asymptotes for functions.
Calculus-Pad	Tertiary	Displays, manipulates and performs operations on a variety of functions in calculus.
Number forms	Secondary	For exploring and learning mathematics.

One way of integrating graphical software packages into the teaching process is to do it after presenting information. The instructor first teaches the content in the traditional way. Learners are taught to first draw all kinds of graphs and then to use a computer to check these or to do more difficult ones. Another way of integrating graphical software packages arises when there is a need to save time or if some graphs are impossible to draw by hand or look complicated on the board, such as graphs in three dimensions. This is probably the most common way of using it.

Instructors may use graphical packages to create pictures that model mathematical concepts. Yet another way of using computer graphics is for instructors to illustrate proven theorems with specific examples. and also to use accurate pictures to suggest possible new results to prove. Instructors can encourage learners to put visual images of mathematical phenomena on the computer for study. manipulation and exploration.

Graphical packages allow learners to position and size rectangles. ovals and other regular shapes. draw irregular shapes using a mouse. fill closed shapes in desired colours or shadings. and to draw lines of various thickness and colours. Learners can use the graphic capabilities of computers in a wide range of ways. ranging from traditional graphing activities to informal geometric exploration to sophisticated three-dimensional design. Learners can also visualise problem elements by using two-dimensional drawing programmes. For example. learners can draw a geometry problem which asks them to calculate the angles at which a ladder of a given length leans against a house at various heights. Using a more sophisticated three-dimensional drawing programme. learners can draw three-dimensional figures. manipulate them. and solve problems involving such shapes. An example of this is LOGO. a high level computer language with a powerful graphics component called turtle graphics. It can be used to teach programming to elementary high school learners. geometric and algebraic concepts to high school mathematics learners and vectors to high school and tertiary learners.

Graphical packages enable instructors and learners to visualise geometric and trigonometric relationships. These are normally difficult to do without the package. According to Walker [34. p640]. "in the special classroom equipped with the trichromic projector. it is possible to expand the image appearing on the computer screen on a large wall screen. This allows the use of graphic facilities of the computer to show. in a limited time. a number of different examples". Learners are freed from the drudgery of sketching and erasing lines and curves on graph paper. and they can then focus on concepts behind graphs and formulae instead.

Quoting Vockel [32. p66]. "automated plotting of functions makes it possible for learners to solve many more problems than would otherwise be possible in the same amount of time". This increase in the efficient use of academic learning time may lead to a better understanding of topics under consideration and to a more productive application of principles to higher order problem solving. Graphical packages enable learners to explore mathematical concepts. thereby. shaping their intuition. and teaching them to enjoy the experimental side of mathematics as well. When learners are able to plot related graphs and see the effects whenever variables are changed. they may gain a deeper insight into the relationship between functions and their graphs. Graphical packages may thus enable learners to visualise what the instructor is saying quickly. Geometry packages enable learners then to experiment with figures. arriving at conjectures that may be proved in the end.

The use of geometrical figures with the accuracy that is offered by graphical packages enhances the understanding of mathematical concepts being presented. Kimmins [21. p2] states that. "three-dimensional graphical packages allow learners to view solids from different perspectives. thus helping learners to see models they had difficulty visualising otherwise and at the same time improving their spatial visualisation capabilities".

3.5 Tutorial packages

To function as a tutor. the computer must be programmed by people who are experts both in programming and in the subject matter. Here the computer presents subject material and the learners respond. The computer evaluates the responses. and from the results of the evaluation. determines what to present next. At its best. the computer tutor keeps complete records on each learner. Tutorial packages attempt to teach new information and new conceptual understanding. A tutorial lesson starts by presenting information which guides the learner to an understanding of the information and ideas presented. This is followed by a series of questions. When learners successfully answer these questions. the computer presents more information. When they are unsuccessful in responding to the questions. the computer either gives a hint or a repetition of previously stated details followed by more questions. Tutorial packages function in a role similar to an instructor explaining information or concepts to the learner. They present new information. and attempt to teach the material through dialogue with the learner. The computer provides information in small segments on the screen. At least some of these screens prompt learners for a response. The computer's next step depends on the learner's response. The best computerised tutorial packages employ a strategy called branching programmed instruction. Possible branching options include:

- supplying the learner with the next piece of information in the learning sequence.
- sending the learner to a remedial branch of the programme to permit review or to explain the information in a different manner, and
- allowing the learner to seek help by getting definitions, clarifications or other information necessary to solve a problem.

Table 3.4. Examples of tutorial packages.

Package	Level	Description
Mastering mathematics: decimals	Secondary	Presents information to learners and then asks a question about that information.
Word-problem tutor	Secondary	Solves word problems.
Vectors and scalars	Tertiary	Helps learners understand properties of scalars and vectors.
Geometry concepts	Secondary	Provides practice in identifying and spelling geometric terms.
Equations	Secondary	Solves algebraic and linear equations.
Integer	Primary	Teaches addition. subtraction. multiplication and division of positive and negative numbers.

Tutorial packages may be integrated into the lesson at the beginning by presenting information. According to Grabe [12. p80]. "tutorial packages can be integrated during the second stage of instruction to guide learning as learners struggle to understand the information". This can be given in the form of exercises that cover the content presented in class.

Vockel [32. p35] says that. "tutorial packages provide immediate and interactive feedback which is often difficult to provide in a written format". By presenting one screen at a time. the computer can sometimes focus attention more precisely on a single concept. By using tutorial packages. learners tend to work for longer and with more interest. Tutorials can more precisely tailor the rate of progress and the content of presentations to the needs of the individual learner. Immediate adaptations in instruction can be made. and learners can interact with the tutor. Quoting Stephen [29. p17]. "with well- designed software packages. the computer tutor can easily and swiftly tailor its presentation to accommodate a wide range of learners' differences".

There is a danger in using these packages. Sometimes the screen is not big enough to contain all the information needed to present an idea. Segmenting concepts onto separate screens can be distracting. Tutorials are limited. not only by the imagination of the designers. but also by the computer technology itself.

3.6 Drill and practice packages

Packages that fall under this category are those that give learners practice in various areas such as operations with real numbers. number sequences and factorising. They are designed to provide practice in concepts and skills learners have already learnt. Such packages are usually designed to pre-test learners. and on the basis of these results. provide the appropriate level of practice. Most drill and practice packages randomly generate mathematics problems and then check if the answer is correct. Learners are provided with problem sets. presenting a sequence of skills to be practised. These packages are designed to exercise arithmetic computations or algebraic manipulations. There are also factoring packages that exercise skills in solving equations. They give learners extensive repetitive work that reinforces concepts previously taught.

Table 3.5. Examples of drill and practice packages.

Package	Level	Description
Algebra blaster	Secondary	Pre-algebra and algebra software.
Addition and subtraction of story problems	Secondary	Helps learners practice addition and subtraction.
Beginning mathematics -word problems	Secondary	Solves word problems in addition, subtraction, division and multiplication.
Mastering mathematics	Secondary	Provides drill and practice using real numbers.
Stuckybear word problems	Primary	Motivates learners to work at solving word problems.
Integer fast facts	Primary	Teaches addition, subtraction, division and multiplication.
Magic grid	Secondary	Provides learners with an opportunity to practice equation-solving skills.
SERGO	Primary, secondary and tertiary	A comprehensive mastery learning package.
Graph-calc	Tertiary	Designed for pre-calculus and calculus classes.

Table 3.5. Continuation.

Package	Level	Description
Algebra drill and practice	Secondary	Offers drill and practice in solving linear equations and systems of linear equations in two or three variables.
Problem	Tertiary	For any exercise in which learners must calculate numerical results from numerical data.
CAMI maths	Secondary	Covers all aspects of secondary school syllabus under algebra, trigonomerty, geometry and graphs.

Grabe [12, p92] is of the opinion that, "drill and practice packages should not be used to introduce new concepts in mathematics as they have a too narrow approach to helping learners understand new material". The learners' initial exposure to academic facts or skills is often insufficient for an adequate level of mastery. Extended study may be required before facts or skills can be considered mastered. This is where drill and practice packages can be introduced. They are developed to meet the needs of the third stage of instruction, the extended practice stage. They are, thus, appropriate after learners have advanced past the guidance phase of instruction.

According to Grabe [12, p93], "the exact proficiency learners should develop varies with the type of content. For factual information, the expectation is that learners will be able to effortlessly retrieve the information from memory. With procedural skills, the expectation is that learners will be able to perform quickly, smoothly, and with few errors. In cases where learners are not able to do so, drill and practice packages may be integrated".

Smith [28, p2] is of the opinion that, "drill and practice packages reinforce learners learning by correct answers and they are able to identify those areas in which their knowledge is inadequate". Instructors are able, by keeping track of learners' records, to determine topics in which the learners need further training. Quoting Jones [19, p1], "instructors can modify the content, using drill and practice, to fit the lesson they are teaching". For example, a good drill software for spelling allows the instructor to input learners' current spelling list or particular words with which particular learners have difficulty. Repeated practice may help learners to internalise skills to make them automatic so that they can apply them to real-life situations.

Some packages often include branching features that establish an appropriate starting point for learners and avoid unnecessary repetition of previously mastered concepts. The level of difficulty is automatically adjusted during operation. This keeps learners from becoming bored and from needless repetition or frustration arising from material being too difficult for their present level of understanding. This reinforcement motivates learners. According to Walker [37, p735], "lessons based on drill and practice packages are usually more self-paced than traditional ones, and allow learners to control the level of explanatory detail required". Explanatory feedback from errors enables learners to learn from their own mistakes.

Smith [28, p2] says that, "drill and practice packages sequence problems". With written problem sheets, the sequence of problems learners work on is determined in advance. Learners become bored by repetitive exercises and testing problems that they have already mastered. They profit from working on easier problems before proceeding further. Problems are characterised according to level of difficulty. Learners may begin to work at level 1 and move up to level $n+1$ when they have demonstrated mastery of level n, perhaps by doing five problems correctly in succession, or by meeting some other criterion of success. The computer keeps track of individual learners' histories so that they can re-enter the system where they left off. The diagnostic abilities of some of these packages are a facet where the computer is often better than the instructor. Even a very good instructor can hardly determine accurately what the exact abilities of each of his/her learners are in every subdivision of the subject.

Overusage of drill and practice packages may have a negative effect, however, producing boredom rather than excitement. This could lead to rote learning. Drill and practice packages may sometimes be boring, yet, for many learners, they become much more tolerable when lessons provide self-paced problems, and when creative graphics, animation, and sound are effectively employed for information. The same techniques may however be distracting for learners or may reinforce incorrect responses to problems.

3.7 Game packages

Instructional activities are categorised as games when the activities emphasise competition and entertainment. If the activity has a winner or a loser or focuses the learner on competing against established records or standards, the activity has game-like qualities. Educational games are generally drill and practice packages spiced up by using a fantasy format and having scoring options. Many packages are designed to give learners practice in problem solving as well as in particular subject areas, concepts and skills. They test cognitive skills and motor reaction time in a game-like atmosphere. Games that require logic or strategy play an important role in developing thinking skills which are crucial to learning. These packages are normally aimed at the primary and secondary phases of education.

Table 3.6. Examples of game packages.

Package	**Level**	**Description**
Minus mission	Primary	Helps learners to practice subtraction facts.
Green globs	Primary	Enables learners to write equations to create graphs that pass through certain points.
Number munchers	Secondary	Learners practice arithmetic skills by munching numbers.
Laser math	Secondary	Practices addition. subtraction and multiplication.
Math invaders	Secondary	Provides practice in basic addition. subtraction and multiplication.
Darts	Secondary	Designed to teach fractions.
Multiploy	Secondary	Provides a game to be played with addition. subtraction. or multiplication problems.
Songwriter	Secondary	Provides an interesting perspective on music and mathematics education.

Grabe [12. p97] maintains that. "educational games can be integrated at the beginning of the lesson in order to provide interesting ways to initiate related areas of study". This may also be done to activate existing knowledge and stimulate learners' interest in more traditional work that follows. Game packages provide an opportunity for learners to go beyond drill and practice of arithmetic skills. Learners can develop insights into number theory and problem solving. Computers keep records from one game to the next. and display high scores and the name of the current record holder.

According to Grabe [12. p98]. "game packages may be used to enhance competition and co-operation". Instructors may use educational games to reward an achievement. For instance. the instructor can allow those learners who always get their work done first to spend time with computer games. These packages can be used for the development of knowledge and assimilation, recall, reproduction, computational skills, mathematical attitudes, personal traits or interest. Individuals and small groups of learners can play these games.

3.8 Computer Algebra Systems

Computer algebra systems are powerful computer software packages that do symbolic as well as numerical and graphical mathematics. The appearance of computer algebra systems has had a significant impact on, in particular, the curricula of tertiary institutions. Computer algebra systems manipulate a formula symbolically. For example, they can be used to expand, factorise, find root or simplify an algebraic polynomial. These packages do symbolic differentiation, integration and find solutions to equations. Computer algebra systems have the capability to evaluate symbolic solutions, approximate numerical solutions, and produce graphically generated approximate solutions. They use commands similar to mathematical notation, which permit learners to clearly or easily draw functions and their derivatives, find roots of functions and derivatives, and obtain numerical graphical output. Computer algebra systems used in learning mathematics have good graphic capabilities and help facilities.

Table 3.7. Examples of computer algebra systems.

Package	Level	Description
Derive	Tertiary	Performs numeric and symbolic computations in algebra and calculus.
Mathematica	Tertiary	Performs symbolic integration and differentiation of functions and much more.
Maple	Tertiary	Performs symbolic integration and differentiation of functions and much more.

Table 3.7. Continuation.

Package	Level	Description
Matlab	Tertiary	Powerful package for solving problems in linear algebra and many more.
Mathcad	Tertiary	A numerical processor.
True basic	Tertiary	Finds the derivative of a function.
GAP	Tertiary	A system for computational discrete algebra with particular emphasis in group theory.
Reduce	Tertiary	Designed for general algebraic computations.
Felix	Tertiary	Designed for computations in and with algebraic structures and substructures.
Magma	Tertiary	Designed to solve computationally problems in algebra. number theory and combinatorics.
Macsyma	Tertiary	Solves systems of equations and much more.

According to Walker [36. p425]. "computer algebra systems can be integrated in class after the theory has been covered". One of the reasons for the introduction of computer algebra systems into the class is that learners lose time in getting started with real calculations. This distracts their attention from the problem to be solved.

Computer algebra systems enable the instructor to ask open-ended questions with the confidence that learners will attempt to find the answers. For example. the instructor may ask about the integral of expressions of the form $(a + bx + cx^2)^q$. Questions like this are beyond the capabilities of first year learners who do not have access to computer algebra systems. In modelling problems. there is the opportunity to concentrate on setting up the problem and interpreting the solution. while leaving the solution of the mathematical equation to the packages. Instructors of undergraduate courses are. thus. able to enhance their teaching by using computer algebra systems.

When drawing graphs in calculus. much time is spent in finding the derivative of the function and in the determination of critical points. By the time learners get to the point where it is possible to use the information obtained to sketch the curve. a lot of work has already been done. The learners rush through the main part of the exercise with the feeling that important parts have already been done. This is a poor use of learners' time. The use of these packages force learners to concentrate on the topic of the assignment. The removal of the manipulation phase of solving the problem forces learners to concentrate on processes of setting up the problem and analysing the results of the manipulation to solve the problem. The amount of manipulation involved in developing many mathematical concepts gives learners the false idea of the importance of the manipulative section of the development process. With computer algebra system packages. more time can be spent on the actual idea of the concept.

The use of a computer algebra system provides great opportunities for allowing learners to experiment. analytically as well as numerically. without the tedium of complicated algebraic manipulation. Burn [7. p62] reflects that. "although the need for a grasp of the basic manipulative skills will always remain. the use of computer algebra systems will open up many more courses to learners whose algebra is relatively weak. and will enable them to follow mathematical courses avoiding frustrations caused by elementary errors of manipulation".

Quoting Smith [28. p60]. "for many learners there are only two problem-solving skills. These are finding a suitable worked example to mimic. and carrying out algorithmic computation. To be effective solvers. learners need to consider alternatives. to experiment. to conjecture and test. and analyse results". Shifting the burden of computation to a computer algebra system makes time available for learners to concentrate on how to approach a problem. consider alternatives and experiment. It may also make time available for concentrating on concepts and processes rather than on the mastery of algorithms. According to Smith [28. p32]. "computer algebra systems allow learners to gain experience in combined use of symbolic. numerical approximation. and graphical methods for solving problems".

The value of the computer algebra system as an educational tool is dependent on the type and objectives of the task for which it is to be used. According to Smith [28. p40]. "using a computer algebra system for a homework assignment to differentiate or integrate a collection of functions from a standard calculus text would be a waste of time. yield a negative learning and lead to the 'button pushing robot' syndrome". These packages may effect changes in computational skills. For instance. there may be a decrease in learners' abilities to recall and carry out many standard algorithms. such as those for factorising and integrating.

3.9 Simulation packages

Before jet pilots are allowed to pilot certain planes for real. they are required to practice in a flight simulator. a computer-driven environment where the pilot sits in a realistic cockpit and is put through training exercises. The pilot manipulates the cockpit controls and the computer produces reactions simulating those that the pilot would experience if he/she were actually in flight. According to Kelman [20. p48]. "a simulation is a representation of a real-world situation in which a specified number of factors can change. as they do they produce other changes throughout the simulated world". Simulation is the modelling of reality to understand and solve problems. Simulations are designed to highlight the relationships between particular variables in the real world. They provide controlled learning environments that replicate key elements of real-world environments. A simulation's focus on a limited number of key elements provides a simplified version of the real world that allows the learner to learn a topic or skill effectively. A simulation is designed so that actions the learner takes within a simulated environment produce results similar to those that would occur in the actual environment.

Table 3.8. Examples of simulation packages.

Package	Level	Description
Sell apples, plants, lemonade	Primary	Simulates businesses that sell apples. plants and lemonade.
Furs, nomad, sumeria, voyageur	Primary	Encourages learners to apply their knowledge of four basic operations with rational numbers.
The Oregon trail	Secondary	Teaches basic operations and logical steps in problem solving.
South dakota	Secondary	Requires learners to make farming decisions involving mathematical skills.
Monte-Carlo	Secondary	Enables learners to introduce statical variation into simulation models.
Explorer metros	Secondary	Designed to give learners experience in using estimation.
Survival math	Secondary	Provides learners with the opportunity to apply and integrate arithmetic skills in realistic situations.

Grabe [12. p83] says that. "simulations can be integrated before formal presentation of new material to stimulate learners' interest. to activate what learners already know about the topic. and to provide a concrete example to relate to the more general discussion that follows". These can also be integrated after learners have been exposed to a new topic. In this approach. simulations allow learners to attempt to transfer what they have learnt to an actual application. and so perhaps reveal misconceptions. Computer simulations provide practical alternatives to certain hands-on activities requiring expensive equipment. dangerous materials. trips outside of the school or other time-consuming processes.

These packages do provide opportunities to apply mathematical skills in interesting and practical settings. The simulation approach attempts to place the learner in a real-world situation in which choices are presented. where he/she must make decisions and then face the consequences of those decisions. Some of the simulations place learners in a problem-solving environment and teach them to apply mathematics to real-life situations. The packages allow learners to see entities that may be invisible or difficult to see when working with real-world problems.

The simplification allowed by a simulation package helps learners to focus on critical information or skills and makes learning easier. Simulations also allow learners to observe phenomena that are not normally visible. and to control processes that are not normally controllable. They put learners in control of situations with which they would seldom be allowed to experiment under any other circumstance. In certain situations. simulations provide quality experiences at a reasonable cost.

Simulation packages set up a world that is governed by a limited set of rules. Thus. learners are given little information about the source or reason for these rules. Through interaction with artificial worlds. learners learn something about hypothetical thinking. but they do not develop any sense of what is required to construct a simulation package in the first place. It is also difficult to find optimal solutions using simulations. The only way to attempt to optimise using a simulation package is to make a change and run a simulation package to see whether an improvement has been achieved or not. and repeat. Large amounts of computer time can be consumed by this process.

Chapter 4

Conclusion

Chapter 4

Conclusion

Computers have proven to be powerful problem-solving tools that can also promote the understanding of mathematical concepts. For this reason the integration of computer-assisted instruction into the teaching and learning of mathematics should be encouraged.

The task of selecting appropriate software packages is not an easy one and obstacles may abound. For example. instructors could be restricted by stringent budgets that limit their choice. Or. the task of deciding could be given to a curriculum committee where there could be a difference of opinion as to what product is best suited. This could prove to be problematic. especially if the software package to be purchased has to be used for a variety of classes or for a long period of time. Another obstacle is. of course. a lack of knowledge concerning content and use of computer software packages. This can only be resolved by following a steep learning curve that requires time and patience. It is our hope that this dissertation may somewhat alleviate this task.

Important issues to consider when selecting a software package is firstly. whether the particular package is likely to assist instructors in teaching and learners in learning for their particular situation. In other words. the package should suit the course. Secondly. the instructor should consider his/her own style and goal when selecting a package. and to be clear on how the package will be integrated into the teaching process. Thirdly. instructors should feel committed to using the particular package and feel comfortable when using it.

Once the choice has been made and the package is at hand, the instructor needs to acquaint himself thoroughly with it. This can be done by trial and error if the instructor has an experience of other packages. It could also be done by working through a tutorial, a regular feature with most packages nowadays. Communication with experienced instructors, be it via e-mail or by attending conferences on computer-assisted instruction, is invaluable. Communication is especially useful in times of encountering seemingly insurmountable problems which can be resolved in no time at all by more experienced peers. Direct communication with software companies via the internet is another option. A final useful source of information on computer-assisted instruction is the rapidly growing collection of journals, specifically on this topic. An example here is "Journal of Computers in Mathematics and Science Teaching". This journal also carries useful software reviews. Instructors may subscribe to such publications themselves or encourage schools to make them available through the school libraries.

The way in which an instructor integrates technology into his/her mathematics teaching will depend on the available infrastructure. If only a few computers are available, the instructor could begin by using it for demonstration purposes. With more computers becoming available in time, learners could become actively involved.

Our experience is that having learners working on the computers in groups of up to four, works well. Learners in a group begin discussing the work and fruitful discussions are generated. The instructor should try to become involved in some of these discussions so that he/she can make contributions that encourage further and deeper thought.

In whatever capacity, computers should form part of our mathematics teaching. Computers are an integral part of everyday life, and we have the responsibility of preparing our learners for what they are going to be faced with in society.

The author hopes that this dissertation will assist schools that are not knowledgable about computer software packages for teaching mathematics.

References

(1) Abelson H. 1981. Turtle geometry. The computer as a medium for exploring mathematics. United States of America: The MFT Press.

(2) Appel M. 1987. A critique of CAI: The case of SERGO. *South African Journal of Education. 7(4): 278 - 282.*

(3) Bajpai A.C. 1988. Using computers to enhance the learning of mathematics. *International Journal of Mathematical Education in Science and Technology. 19: 83 - 85.*

(4) Ball D. 1996. Microcomputers in maths teaching. Great Britain: British Library Cataloging.

(5) Beasley J. Simulation. OR-notes. Electronic Publication at *http://graph.ms.ic.ac.uk/jeb/or/sim.html*

(6) Bishop A.J. 1989. Technology and mathematics education. A survey of recent developments and important problems. *Educational Studies in Mathematics. 20: 237 - 243.*

(7) Burn B. 1998. Teaching undergraduate mathematics. Singapore: Imperial College Press.

(8) Burns D. 1999. Computers and our society. Electronic Publication at *http://student.1ssu.edu/~dburns/presentation/ts1d001.htm*

(9) Dubinsky E. 1999. Computers and school mathematics reform: Implications for mathematics teacher education. *Journal of Computers in Mathematics and Science Teaching. 18(1): 31 - 45.*

(10) Dyckhoff R. 1990. CTI mathematics workshop 03. Computer packages for the teaching of discrete mathematics. Electronic Publication at *http://www.bham.ac.uk/ctimath/workshops/wdis.htm*

(11) Engelbrecht J.C. 1993. Mathematic education with the SERGO-system. University of Pretoria.

(12) Grabe M. 1996. Integrating technology for meaningful learning. United States of America: Library of Congress Catalog.

(13) Graham D. The design and implementation of a computer-assisted learning package to demonstrate the fetch-execute cycle. Electronic publication at *http://www.ulst.ac.uk/cticomp/davies.html*

(14) Harding R. 1996. Mathematical software packages. Electronic publication at *http://www.bham.ac.uk/ctimath/gateway/packages.htm*

(15) Hausler T. 1990. CTI mathematics workshop 03. Computer packages for the teaching of discrete mathematics. Electronic Publication at *http://www.bham.ac.uk/ctimath/workshops/wdis.htm*

(16) Hayward J. Division of mathematics. teaching and learning. Electronic publication at *http://www.maths.glam.ac.uk/maths/teach.html*

(17) Herng-Jeng J. 1999. SAL-mathematics-computer algebra systems-text deriv. Electronic publication at *http://sal.kachinatech.com /A/1/ TDERIV.html*

(18) Howard C. 1996. 21st century problem solving - Welcome page. Electronic publication at *http://www-hpcc.astro.washington.edu/mirrors/home.html*

(19) Jones D. 1999. Integrating technology in the classroom: Game/Drill software. Electronic publication at *http://www.crownpointpanthers.com /tech/drill.h*

(20) Kelman P. 1983. Computers in teaching mathematics. United States of America: Addison Wesley Publishing Company. Inc.

(21) Kimmins D. 1996. Making mathematics come alive with technology. Electronic publication at *http://www.mtsu.edu/-itconf/papers96/kimmins.html*

(22) Klooster J. 1998. Computers in the elementary school classroom. Electronic publication at *http://student.1ssu.edu/~jklooste/computers/s1d001.htm*

(23) Kutzler B. 2000. Introduction to Derive 5. The mathematical assistant for your PC. United States of America: Texas instruments. Electronic publication at *http://www.ti.com/calc/docs/derive5.htm*

(24) Maree J.G. 1997. Using CAMI mathematics in the classroom - A case study. *Journal of Education and Training. 18: 17- 29.*

(25) Martin E. 1996. Mathematica 3.0. Standard add-on packages. United States of America: Library of Congress Cataloging.

(26) Richard H. 1986. The geometric supposer: Promoting thinking and learning. *Mathematics Teacher 79: 418 - 422.*

(27) Ruthken K. 1998. Disentangling the nexus: Attitudes to mathematics and technology in computer learning environment. *Educational Studies in Mathematics. 36(3): 275 - 290.*

(28) Smith D.A. 1988. Computers in mathematics. The use of computers in undergraduate instruction. Library of Congress Catalog. Volume 9.

(29) Stephen B. 1997. Computer as a tutor. Electronic Publication at *http://www.albany.edu/faculty/swan/tutor.html*

(30) Steyn T.M. 1998. Graphical exploration as an aid to mastering fundamental mathematical concepts: An instructional model for mathematics practicals. University of Pretoria.

(31) Strotman A. 1995. The reduce computer algebra system. Electronic publication at *http://www.uni-koeln.de/DERIVE*

(32) Vockel E. 1989. The computer in the mathematics curriculum. United States of America: Mitchel Publishing. Inc.

(33) Walker D. 1983. Computer-aided teaching of three-dimensional geometry. *International Journal of Mathematical Education in Science and Technology. 14(2): 129 - 135.*

(34) Walker D. 1988. The impact of new technologies on teaching calculus: A report of an experiment. *International Journal of Mathematical Education in Science and Technology. 19(5): 637 - 657.*

(35) Walker D. 1992. A linear algebra course with PC-Matlab: some experiences. *International Journal of Mathematical Education in Science and Technology. 23(3): 421 - 428.*

(36) Walker D. 1992. Using PC-Matlab in the teaching of linear algebra. *International Journal of Mathematical Education in Science and Technology. 23(3): 413 - 419.*

(37) Walker D. 1992. An evaluation of computer-assisted learning in mathematics. *International Journal of Mathematical Education in Science and Technology. 23(5): 731 - 737.*

(38) Walker D. 1994. Mathematics teaching: The issue of quality. *International Journal of Mathematical Education in Science and Technology. 25(6): 823 - 830.*

(39) Walker D. 1995. A study of teaching mathematical concepts with computers. *International Journal of Mathematical Education in Science and Technology. 26(4): 473 - 488.*

(40) Wiebe J. 1993. Computer tools and problem solving in mathematics. Using technology in the classroom series. Oregon: Library of Congress Cataloging.

(41) Williams G. 1997. Introduction to educational technology. Electronic publication at *http://www.csu.edu.au/division/oli/celt/edtech/index.html*

(42) Williams K. 1999. Bradford problem-solving series. Electronic publication at *http://www.wkbradford.com/title/brps.html*

(43) *Journal of Computers in Mathematics and Science Teaching.* Charlottesville, USA: The Association of the Advancement of Computing in Education (AACE), an international, educational non-profit organization.

(44) Mathematics: UofA software programmes. Electronic Publication at *http://www.math.arizona.edu/software/azmath.html*

(45) Of special interest to teachers. Electronic publication at *http://www-hpcc.astro.washington.edu/mirrors/suremath/teachers.1html*

Appendix

Here we give a more detailed description of some of the packages that are mentioned in Chapter 3. We describe what these packages do and state the user friendliness where possible. The packages are listed alphabetically.

(1) Addition and subtraction of story problems. The problems in this package are presented in groups of five. In the first drill, the learner is asked to enter two numbers. The operation is provided. The learner then enters the answer and the proper label. In the next level of difficulty, the learner types in a word to complete the question and then computes and enters the answer. As the difficulty level increases, the learner is given fewer and fewer aids.

(2) Algebra blaster. Each topic involves three levels of work for learners. The first level is called 'study the steps' in which two examples are worked out in detail with an accompanying explanation of how the problem should be solved. For example, the rule of adding signed numbers may be explained. By using the arrow keys, learners may go back and review the previous step as many times as necessary. The second level is called 'build your skill'. When learners choose this level, they may tell the computer whether they wish to start at level 1 or where they left off. In this level, a prompt is available for learners who require assistance. If the problem is missed twice, the correct answer is provided. Learners may choose to retake any missed items at the end of the sequence. The third level, called 'solve it' is the most difficult. There are no prompts available at this level, but there are certain hints. The type of cursor tells learners whether a sign, numeral, variable, exponent or a fraction is expected. Blanks are provided so that learners know how many digits the answer should contain.

(3) Algebra drill and practice. This package also offers drill and practice in equations and inequalities involving absolute values, simplifying expressions with integral exponents, factorising linear and quadratic expressions, finding roots of quadratic equations, x-intercepts and the vertex of parabolic graphs. and solving inequality.

(4) **Algebra and function plotter**. This package allows learners to use the computer's graphics to aid their understanding of the effects of changing various parameters of a function. such as reversing the sign of a leading coefficient [32. p66].

(5) **Beginning mathematics-word problems**. This package makes good use of graphics. and aims to motivate learners and to provide positive reinforcement for correct answers. The problems are all based on the theme of gold miners making their way to the West in search of gold. The addition/subtraction disk has a series of six units that build in progression. Each unit consists of five problems for learners to solve. The multiplication/division disk has six units that parallel the addition/subtraction ones exactly. The disks provide a wide variety of ways to practice solving word problems. Since learners can start at a simple level. they can enjoy success and not be frustrated. The last section of each disk allows learners to make up their own problems.

(6) **Build**. The programme starts with one cube already drawn on the screen. Learners build pictures by instructing the computer to place a brick next to the brick last placed. The position of the new brick can be to the right. to the left. up or down and in or out. One activity is to investigate the various different ways of instructing the computer to draw an eight-brick cube [4. p80].

(7) **Calculus-Pad**. This is a superior plotting software package which learners can use to learn operations on functions in a regular calculus course. It is designed so that learners can do experiments with functions quickly to increase their understanding.

(8) **CAMI maths**. CAMI is a drill and practice package that randomly generates a multiple of examples for learners to work through. CAMI is not a multimedia package as it does not make use of sound and animation.

The only graphics are graphs and diagrams that are directly relevant to the question on the screen at that moment. It does not have an adaptive mechanism to adjust the difficultly level. CAMI frees learners from dependence on the instructor. and it makes it possible for learners to share ideas and work co-operatively in solving problems. Since the role of the package is to promote mastery of a sharply focused syllabus and set of techniques. it is errorless in that each problem has only one correct solution. The motivation in the package comes from getting the correct answer. This is marked with a green tick. and a word of praise. The package builds confidence and mastery of the fundamental mathematical techniques. reducing fear and feelings of alienation [24. p20].

(9) **Darts**. Balloons appear on the number line and learners guess where the balloons are by typing in numbers. Each time they guess an arrow shoots to the specified position. The arrow pops the balloon when the guess is correct [20. p289].

(10) **Derive**. Derive offers symbolic algebra and powerful graphics. It is an excellent tool for teaching and learning mathematics. Derive eliminates the drudgery of performing long and tedious mathematical calculations. While Derive takes the burden of doing algorithmic parts of solving a problem, learners can concentrate on the mathematical meaning of concepts. Instead of teaching and learning boring technical skills. instructors and learners can concentrate on the exciting and useful techniques of problem solving. Derive algebraically simplifies. expands and factors expressions. Equations and systems of non-linear polynomial equations can easily be solved. yielding their real and complex solutions [23. p1].

(11) **Differential equations graphics package**. This package includes programmes for linear first and second-order differential equations. Euler approximation. graphical comparisons of Euler. tangent fields. logistics differential equations. Frobenius series. and Fourier series. The linear first and second-order programmes allow construction of non-zero right-hand-

side from a menu of components. All other programmes have this feature, and allow the learner to enter the desired expression directly. The Euler approximation programme allows equations of the form $y' = f(y)$ and plots several approximations for different numbers of intervals. The tangent field programme plots trajectories for initial positions entered from the keyboard. The Frobenius programme allows input of the first 5 terms and plots 5 approximations. The Fourier programme plots 3 approximations [28, p70].

(12) **Discrete mathematics**. This package is user-friendly and requires no knowledge of programming. According to Smith [28, p44], there are eight routines in this package. These are:

- **Truth table**. This routine generates truth tables for logical statements entered by the learner. These tables have up to four variables and a number of logical connectives and($\wedge$), or($\vee$), not($\rceil$), implies($\Rightarrow$), and if and only if($\Leftrightarrow$). Learners can enter a single statement and get its truth table or enter two statements to get a truth table for each, and a statement as to whether one logically implies the other.

- **Venn diagrams.** This routine enables learners to enter up to three set names and to request venn diagrams of intersections, unions, set differences and complements. Learners can also enter the number of elements in each of the sets, and request that the venn diagram shows the corresponding number of elements in some combinations of the sets.

- **Counting**. This routine enables learners to calculate combinations, permutations, partitions (the number of ways of dividing n things into cells of specified sizes), boxes (the number of ways of arranging n identical objects in j boxes) and the Pascal triangle.

- **Recursive**. This routine allows learners to define a function recursively and request the value $f(n)$ for any natural number n for which the function is properly defined.

- **Graphs**. This routine contains options for drawing graphs and digraphs specified by learners' input of vertices and edges.

- **Binary**. This routine provides a graphic demonstration of a binary search and a binary tree for the learner's list of words.

- **Sorting**. This routine demonstrates bubble sort, demonstration sort, and a quick sort for a random list of numbers.

- **Equation**. This routine solves systems of linear equations with up to eight unknowns.

(13) **Equations**. This package is designed for introductory algebra courses when learners are learning to solve linear equations of the form $AX + B = C$. Instructions are sequenced so that learners practice steps, in the proper order to solve equations. Since problems are randomly generated with A, B and C chosen as nonzero integers, a wide variety of problems are available. After each equation is presented, the learner must choose whether to add or subtract the same number from each side of the equation. After simplifying both sides, the learner then chooses whether to multiply or divide each by the same number. The operations of multiplication and division are interchangeable to the extent that the learner may choose to divide, say by 2 or multiply by $\frac{1}{2}$, in order to finish solving a problem.

(14) **Explorer metros**. In this package, learners explore a space colony and encounter situations requiring them to choose one of three alternatives based on a given metric length, mass capacity or temperature. Learners must complete the journey within a time limit. One particularly useful feature of this package is the modification option which allows instructors or learners the opportunity to redesign the encounters and the material covered. The package varies the elements of the simulation each time it is run so that individuals or small groups will enjoy playing the game several times. Use of graphics and on-screen documentation are also a plus [20, p289].

(15) EZQ. This package has very good graphic capability. Functions are built up by filling in blocks that take the exponential of a number or multiply two results or integrate a result. Depending on the amount of memory available, 32 to 96 blocks can be used at one time. Learners need to think about the proper sequence of blocks to solve a given problem. The sequence of blocks can be saved on a disk and used later. The package can be used by the instructor to work out a problem. The instructor can save the problem on a disk, and either show it in class or give the disk to learners [28, p70].

(16) Function graph plotter. This is another graph-drawing package that provides similar facilities and that can be used in similar ways as **Graph**. It draws graphs quicker. It also has a useful facility that enables learners to input a function of the form $y = 2x + a$, for example, and then draw a set of graphs by continually incrementing the value of a [4, p62].

(17) Furs, nomad, sumeria, voyageur. Learners attempt varying tactics, for example, different numbers of furs to be sold to maximise profits. In nomad, learners apply some geometry skills to read a map. In voyageur, learners make decisions on what to take on their imaginary voyage depending on the cost and practicality. Sumeria deals with starving people and for this reason some instructors are reluctant to use it [32, p106].

(18) Geometry concepts. This is a four-part package.

- **Identification game**. This part provides practice in identifying and spelling geometric terms. A figure appears on the screen after the learner correctly identifies it. A screen of information about figures is displayed. Points are awarded based on the number of incorrect guesses the learner makes in naming the figure.

- **Data-retrieval utility**. This part allows the learner to select a figure from the mathematical data bank. A screen of information about the figure is then presented.

- **Quiz machine**. In this part, learners are asked a series of multiple-choice questions.

- **Constructions**. This part presents five basic constructions in an easy-to-follow fashion. Learners may work along with the constructions step by step or watch the sequence for review. The following constructions are presented:

- perpendicular bisector of a segment,

- angle bisector,

- perpendicular to a line through a point not on the line,

- perpendicular to a line through a point on the line, and

- inscribed square.

(19) Geometric constructor. This package allows learners in college-level geometry to perform a number of elementary constructions using points, lines, circles, bisectors, perpendiculars and parallels. Putting these concepts together enable learners to create every possible Euclidean construction. This package includes nine conic-plotting options which allow learners to accumulate as many lines, circles and conics on the screen as they wish.

(20) Geometric supposer. Learners specify the procedure at first. For example, give me triangle ABC. Bisect AB, and label the midpoint as D. Bisect BC and label the midpoint as E. As learners type in instructions, the constructions they have defined appear on the screen. Now that they have defined their constructions, they have the whole construction accessible as a procedure. They can ask for a new triangle ABC, and the supposer will repeat learners' constructions. This permits learners to test conjectures on a new figure quickly [28, p3 and 26, p415].

(21) Graph. This package has an option which enables the instructor to enter the equation of the graph to be drawn without displaying the equation on the screen. The instructor can enter an equation. get the programme to draw the graph. and then invite learners to suggest the equation of the graph being displayed. Alternatively. the instructor can use the option in which the computer selects an equation of the graph to be drawn. In this case. both the instructor and learners share the problem of having to decide which graph is being displayed [4. p60].

(22) **Graph-calc**. Learners can manipulate graphs and equations in much the same way they would manipulate numbers with a calculator. The package consists of three components:

- **Graph**. In this component. learners graph functions in rectangular coordinate. parametric equations. equations in polar coordinates. a function and its derivative. and a function and its integral. Any number of curves may be plotted on the same graph for the sake of comparison.

- **Conic**. In this component. learners graph conic sections and compute pertinent data such as foci. vertices. and asymptotes. Two curves may be graphed simultaneously.

- **Checker**. This component enables learners to check steps in evaluating expressions. solving equations. solving identities. and finding derivatives and antiderivatives. Evaluated integrals may also be checked. This component does not work out the problem for the learner. but helps to isolate the step where the error was made. With this feature. learners are encouraged to make up problems on their own for extra practice. because they are then able to check the accuracy of their work [32. p105].

(23) **Green globs**. This game combines fantasy. challenge and curiosity in an intrinsically mathematical activity. Learners are challenged to destroy thirteen green globs displayed in a coordinate grid. Learners write equations to create graphs that pass through the globs. thereby obliterating them. Points are scored for hitting the most globs with the smallest number of equations. The computer stores the ten highest scores for the game along

with the series of equations that the record holders wrote to achieve the scores. Curious challengers may recall and display records of the record setting games to improve their own winning strategies [20. p58].

(24) Guesstimation. Learners are asked to estimate distances on a number line. order numbers. and use negative numbers. A wide variety of problems allow learners to practice with integers. fractions. decimals. and absolute values. Learners are encouraged to make an estimate before any actual measurement is made.

(25) Gyro graphics. This package allows learners to input their own functions and then creates the graph in seconds. complete with labeled axes and automatic scaling. It also allows learners to automatically add tangent and acceleration vectors to space curves and tangent planes to surfaces.

(26) Integer. Learners may choose two to nine practice problems. A number line is available by typing 'N'. and a marker on the number line can be moved back and forth using '<' and '>' keys. Typing 'R' allows learners to review multiplication and division rules. After two incorrect responses. the computer shows the correct solution [32. p106].

(27) Integer fast facts. This game-like drill package has three versions. In the solo version. the learner gets ten problems per round. Learners may play over and over again while trying to improve their scores. The team version may be played by teams of two to five learners. The team score is the total of the individual scores to encourage learners to co-operate with one another to achieve higher point totals. In the competition version. learners compete against one another. In any of the three versions. learners may choose addition. subtraction. multiplication. division or assorted problems. Learners can repeat the same type of problems or change to another type at the end of the round. The game is a good motivator since the computer keeps track of both the learner's highest score and most recent score for as long as the same kinds of problems are repeated.

(28) Laser math. When the learner enters the answer to the problem presented, and then presses the 'F' key, the computer fires at a phantom cruiser. A correct response destroys the enemy, but an incorrect response results in losing a token to the enemy's torpedo attack. Learners earn points and are rewarded with promotions to higher ranks as their scores increase. All answers to problems range from -5 to $+5$. There are separate units on addition, subtraction and multiplication [32, p108].

(29) Lepack. The package includes algorithms for linear equations, eigenvalues and applications, and has facilities for dealing with arrays. It was initially used to support the teaching of computational mathematics. Its use has now been extended to second year engineering mathematics level [15, p2].

(30) Locus. This package can be used to present ideas to a class about loci or about situations involving loci. Using this package, learners can specify one or more fixed objects to be drawn on the screen with reference to which loci are to be drawn. Each fixed object is either a point or a line segment. For instance, if there is just one fixed object, consisting of one point, the locus drawn will be a circle [4, p60].

(31) Maclogic. The rationale for this package is to ensure familiarity with first-order logic. It covers proof theory. It does not implement specialised methods such as truth tables and venn diagrams [10, p3].

(32) Macmath. This package allows the learner of the Euler, improved Euler or fourth-order Runge-Kutta methods to solve first-order equations for one, two or three-dimensional systems of differential equations. Other routines find eigenvalues and eigenvectors for matrices by the Jacobian method, locate and identify bifurcation points for autonomous planar systems, and compute partial sums of Fourier series [28, p69].

(33) Macsyma. This is a powerful mathematical software package. It contains facilities for algebra, linear algebra, calculus, numerical analysis, on-line help system, and versatile graphics. It includes 1,300 executable demonstrations, easily accessible at many points in the help system. Recent mathematical improvements include enhanced speed in solving linear and algebraic equations, stochastic mathematics and better evaluation of special functions. This package can be used to differentiate, integrate, solve equations of the form $f(x) = 0$, solve systems of linear equations, and perform matrix manipulations [17, p1].

(34) Magic grid. Learners must solve nine different equations selected from a tic-tac-toe-like grid. Each equation requires information from a previously solved equation. Help is given in a friendly manner each time an incorrect response is returned. The answer is supplied after two wrong choices. When Magic grid is completed, a summary of equations solved, with and without help, is provided, as well as the opportunity to do another Magic grid [20, p295].

(35) Mastering mathematics. This package consists of seven content disks and three utility disks. Titles of the content disks suggest the subject matter covered in each. Problems are sequenced according to difficulty levels and may involve operations with up to three digits. Concepts of regrouping and working with real numbers are gradually introduced. One advantage of this package is that instructions on the various disks are very similar, so learners do not waste a lot of time becoming familiar with a new set of commands each time a new disk is used. Concepts are also built in logical progression. Since each content disk has a management option, the instructor can turn graphics on or off, set up the printer, change speeds to review programmes, and review learners' performance. The three utility disks consist of diagnostic and management systems and a worksheet generator. The diagnostic disk enables instructors to test learners on concepts covered in the seven content disks, to establish mastery levels, and to select the number of problems to be given on diagnostic tests. The worksheet generator allows the instructor to create worksheets covering some or all of the material on content disks. The management utility permits the instructor to monitor the progress of learners on each of the content disks.

(36) Mathematica. This package contains a large number of built-in functions. It includes a collection of standard add-on packages that define many additional functions in areas such as algebra, calculus, graphics and discrete mathematics. Mathematica provides many functions for working with polynomials, and standard add-on algebra packages extend that capability. Mathematica has extensive built-in support for calculus including integration, differentiation, differential equation solving and limits. Calculus packages extend this capability by providing Fourier and Laplace integral transforms. Discrete mathematics is concerned with enumerable mathematical structures, such as are studied in combinatorics and graph theory. Graphical packages offer log, polar, and three-dimensional plots [25, p1].

(37) Math invaders. The basic aim is for learners to destroy space invaders. By typing a correct answer and touching the space bar, the learner fires at the enemy. If the answer is incorrect, the enemy continues to move forward. There are never more than three enemies approaching at one time. Learner motivation is further enhanced by promotion to various ranks. The speed factor setting allows learners to compete with themselves as they try to improve the speed with which they can recall the basic facts. Learners who miss the same problem three times are given the answer by the computer. Learners must then touch the space bar and re-enter the correct answer to help in learning the fact [32, p115].

(38) Math pac. Major topics include the arithmetic of algebraic expressions, factorising quadratics, solving linear and quadratic equations, graphing, and solution of linear and quadratic problems. The package can be used as an independent tutorial for learners whose algebra skills are barriers to success in other courses.

(39 Mathpert. This programme is designed to eliminate road blocks by actively helping learners to solve any problem correctly. It allows learners to focus on the correct strategy of problem solving. Mathpert is available in three forms: Module 1 algebra only, module 2 includes algebra and pre-calculus, and module 3 which is the complete package of algebra, pre-calculus and calculus. Algebra topics include linear functions, systems of linear equations, polynomials and polynomial functions. Pre-calculus topics include logarithmic and exponential functions, complex numbers, trigonometry and matrices. Calculus topics include limits, differentiation and integration.

(40) **Math** 246 **programmes**. According to Smith [28, p71], this package consists of a menu driven selection of 4 programmes. These are:

- **Directional fields**. This programme allows a choice from a menu of five differential equations and plots the direction field. Learners can then sketch a trajectory with the mouse. The real trajectory is plotted through the chosen point.
- **Damping**. This programme allows learners to pick values for the damping and the magnitude and frequency of the forcing function for a vibrating string which is shown while the graph is drawn.
- **Systems**. This programme solves systems of the form $\frac{dx}{dt} = ax + by$, $\frac{dx}{dt} = cx + dy$. The eigenvalues are computed and the tangent fields and trajectories are drawn.
- **Phase plane.** This programme allows learners to pick a list of differential equations and plot the $x - t$, $y - t$ and $x - y$ graphs.

(41) **Matlab**. Matlab stands for matrix laboratory. Matlab is an interactive system in which the basic data element is a matrix that does not require dimensioning. This feature allows learners to solve many numerical problems easily. It has a routine that solves a system of linear equations. The main drawback of this package is that it gives only the final solution. It does not give the intermediate steps, which instructors are interested in for the purpose of demonstrating the Gussian elimination procedure. The package is however user-friendly [35 and 36].

(42) **Matrix calculator**. Learners begin by entering matrices into the programme and giving them specific names. Then, when prompted by the programme. learners simply type in the operation they wish to perform and the names of the matrices involved. Since the programme works only with integers and rational numbers. all answers are reported as exact. The package allows learners to perform the following operations: addition. multiplication. row operations. reduction to reduced row-echelon form. and scalar multiplication. Learners can also find the inverse. determinant. transpose. adjoint. characteristic polynomials. eigenvalues and vectors of matrices.

(43) Matrix master. This package consists of two disks and a short manual which enable instructors to demonstrate or learners to perform the following operations on matrices: input. store or retrieve. change individual entries. insert or delete a row or a column. elementary row operations. elementary column operations. sums and averages. transpose. determinants. inverses. add. subtract. multiply. juxtapose horizontally. vertically or diagonally. evaluate polynomials at matrices. characteristic polynomials. eigenvalues. rank. solve linear problems on equations.

(44) MATT graphical display package and differential equation solver. In this package. problems are solved by entering instructions. The package can solve first-order differential equations and systems of first-order differential equations. It can be used to solve higher-order equations by converting them to systems of first-order equations [28. p69].

(45) Mind over matter. One section of this package involves determining which letter is missing from an alphabetical list. At the easiest level. only one letter is omitted. At more difficult levels. several letters are left out. Learners may choose to have letters given in alphabetical order or in reverse alphabetical order. Another section of the package presents a numerical sequence and asks the learner for the missing number or numbers. The numerical sequence may be in ascending or descending order. The beginning problems involve counting and skip counting while the advanced ones challenge learners with obscure patterns.

(46) Minus mission. A blob of green slime drops towards a robot at the bottom of the screen. Learners try to find the slime by entering the answer to a subtraction fact and firing at the slime. Successful hits are recorded in a large slime at the top of the screen. Misses are also recorded. If the slime reaches the robot without being destroyed, the robot disintegrates. Learners may choose from several options at the beginning of the game. There are nine skill levels, and the run time may be varied from one to nine minutes [20, p54].

(47) Monte-Carlo. This method was developed for the design of early nuclear reactors in order to calculate the area of an irregular shape. The method consists of placing an irregular shape on a suitable grid and throwing darts at random so that an estimate of the area of the shape is given by [(number of darts landing on shape) / (total number of darts thrown)] × (total area of grid). The more darts that are thrown the better this estimate will be [5, p2].

(48) Moves. The programme draws axes on the screen and displays a flag. The learner can specify the transformation to be used and the flag can then be transformed. It can be used to discuss properties of different transformation with learners. For example, if different translations are presented on the screen, learners can observe that a translation just moves the flag a bit, that the size and shape of the flag do not change and that the flag is still the same way up. This programme is more useful when the instructor is in control of the situation and is guiding the course of learning. It encourages questions and observations from learners. It is very effective in connection with learners' work that involves drawing and transforming shapes using squared paper [4, p41].

(49) Multiploy. This is an arcade-type game in which learners try to shoot down problem ships before their base is destroyed. The game may be played with addition, subtraction, multiplication, or division problems. There are three levels of difficulty. Learners may choose to have problem ships drop at normal speed or at 'lightening speed'. The game lasts until all the problem ships have been destroyed or until the answer base is destroyed. Learners may change levels of difficulty or operations to be practised at the end of the game. The highest score is retained for each level and operation until the computer is turned off [32, p125].

(50) **Number forms**. This package is for exploring and learning mathematics. Learners can move a number line on the screen and pull off from that number line a series of blocks, 'number forms', which display the number. Learners can then add, subtract, multiply, or divide numbers as the blocks visually perform the same operations. Number forms, thus, provide the bridge between concrete manipulatives and abstract symbols [20, p293].

(51) **Number munchers**. This game allows learners to practice basic arithmetic skills by munching numbers while avoiding the evil troggles. This game can be played by two learners as a team. One may concentrate on the screen and call out the answer to the other who acts as a typist. For a second round the team may exchange positions. In this way learners practice both mathematics and keyboarding.

(52) **Ordinary differential equations.** This package is designed for use with the book "Ordinary differential equations with numerical techniques." It is a set of programmes that implement 17 numerical methods to solve differential equations. These include Euler, improved Euler, Runga-Kutta for first and second-order equations, Milne, Hamming predictor-corrector methods, Bessel methods, 3 series methods and 3 methods for systems of equations. When a method is selected, the learner is asked to enter a differential equation. The programme graphs the direction field and the solution of an initial value problem. It can overlay graphs of solutions to the same differential equation with different initial conditions [28, p70].

(53) **Phaser**. This programme is packeged with a 224-page book titled "Differential and difference equations through computer environments." It has over 60 differential and difference equations built in, each of which has parameters that may be set. Just about any equation learners might wish to

look at is already available. but learners can define some equations. Among the plots available are direction field. trajectories. and phase planes. Three-dimensional plots are available for simultaneous equations, and these graphs can be changed by rotation about any axis [28. p69].

(54) **Plot pak**. This package includes a 26-page instructor's manual and 200 function files so that learners or instructors can quickly recall properly scaled and labelled graphs. Learners can study topics with visual support from powerful. but easy to use. graphic software.

(55) **Quick-Graph**. This package is intended to be used either for in-class demonstrations or by an individual learner. After the learner selects a class of functions. an expression containing cells for parameter will appear at the bottom of the screen. Values for the parameters may be entered into these cells and the resulting function graphed. Since the graph remains on the screen while values are entered. the learner can readily observe the effect of changing a particular parameter.

(56) **Reduce**. The capabilities of this package include expansion and ordering of polynomials and rational functions. calculations with symbolic matrices, analytic differentiation and integration. and many more. The main aim is to support calculations that are not feasible by hand [30. p1].

(57) **Sell apples, sell plants, sell Lemonade**. In each of the three packages. learners have a business which sells one of these products. Learners must try to maximise gains and minimise losses by balancing various factors. Teams of learners can be used to make decisions. They soon learn that decisions are not as simple as they seem. For example. weather. advertising, and the price of the item must be considered. Learners learn that decisions have consequences. They begin to appreciate real-life applications for mathematics [32. p30]

(58) **SERGO.** SERGO is an Afrikaans acronym for Centre for Computer Aided Instruction (Sentrum vir Rekenaarsgesteunde Onderrig). SERGO authors

are university mathematicians. The package ensures that learners will not be allowed to proceed to a new concept or technique if they have not proved that they have mastered the work. This is how a proof is given: the learner is confronted with a question which must be answered. If answered correctly, the next question is posed. If it is answered incorrectly, the learner is given another opportunity to answer the question. A correct answer to the second attempt allows the learner to proceed. If the question is again answered incorrectly, the correct answer is displayed on the screen, and before the learner is permitted to attempt the next question, he must correctly answer ten questions, in a row, similar to the one answered incorrectly. The system does not replace the instructor. It serves as a support to normal teaching [2, p279 and 11].

(59) SIGAD. This computer graphics system has been developed to help in the design of three-dimensional objects such as houses and cars. Sigad is an interactive graphics system for learning three-dimensional geometry and for learning how to visualise in three dimensions. It allows learners to define, for example, a plane relative to three points or relative to a point and a line or relative to two lines. It provides learners with information about positions of current figures. It also helps learners to learn how three-dimensional drawings can be represented. It is a very useful tool for visualising drawings which involve points, vectors, segments, lines and planes [33, p129].

(60) Singos. When the programme starts, a circle with a horizontal radius is displayed, and a box in the top-left corner of the screen indicates that the value of the angle is currently zero. If the learner enters an angle in degrees, the radius rotates slowly until it reaches a position corresponding to that angle. As it rotates, the number in the box indicating the angle changes continuously from zero to the angle specified. In this way, learners can gain familiarity with the way in which different positions of the radius arm represent different positive or negative angles. This programme provides a computerised set of trigonometric tables. It has the advantage that the value of the function is accompanied by a visual explanation. It can be used to discuss signs of trigonometrical functions of angles that are not acute. For example, to demonstrate the fact that as an angle close to zero changes, its sine changes faster than its cosine [4, p44].

(61) Solve. The learner enters a linear equation which is displayed on the screen. The learner can then transform the equation by performing the same operation on the expression on either side of the equal sign. Thus, this programme carries out manipulation for the learner. This leaves instructors and learners free to concentrate on the difficult task of deciding what the appropriate next step in solving the equation should be and the time available for experimenting with suggestions made by learners. Such experimenting may give learners considerable insight into the reasons why one operation is more appropriate than another as the next step in solving the equation. Learners can also discover whether their suggestions are appropriate or inappropriate by seeing the computer carrying them out [4, p45].

(62) Solving quadratic equations. The screen design, sequence and pedagogy of this package has been rated highly by learners. The package has the following features: Each step towards solving quadratic equations is built as a concept. For each concept there is an explanation screen and a sequence of questions leading to the solution of the problem. Learners may either choose to select the problem generated by the computer or define their own problem. At each step learners get feedback and assistance if responses are not correct.

(63) Songwriter. This package helps learners to learn and understand music by composing and editing music scores. The package plays music in tune through the external speaker as the screen scrolls the notes like a player piano. Learners are able to adjust the musical notes by adding, subtracting, multiplying, or dividing. The package has been carefully designed to be easy to use and to be fun to play [20, p295].

(64) South dakota. This package calls for learners to use mathematical skills to make farm management decisions. Learners must decide which crops to plant and in what quantities in order to make a maximum profit. Learners also choose whether to hire extra workers at an additional cost in order to increase production.

(65) Spring and string. This package consists of specialised programmes for two applications of differential equations. Both programmess have very good graphics. The spring programme allows the input of values for the mass. damping constant. spring constant. magnitude and period of a sine wave forcing function. and initial position and velocity. The programme displays the analytic solution and shows the spring moving up and down while the graph is traced out. The string programme has three options. The first option displays the partial sums of the Fourier series solution for one of five problems at a specified time. The second displays the Fourier series solution at various times. Learners control the number of terms used and the number of points displayed. The third option allows the d'Alembert solution [28. p71].

(66) Stuckybear word problems. In this package. each correct answer is followed by a short animation sequence in which the tally of correct responses is increased by one. In one of the cartoons the drips of an ice-cream cone activates the correct response counter. The combination of extrinsic reinforcement (from the animated packages) and intrinsic reinforcement (from solving interesting problems) motivates learners to work at the solution of word problems.

(67) Survival math. This package is a set of four simulations. All of these simulations are practical. well designed and challenging. The activities are an excellent way to measure learners' abilities to use their arithmetic skills and to work on tasks as a group. Because of the on-screen documentation and the worksheets provided. the first two simulations can be attempted by capable learners without much assistance. The latter two. however. will require instructor's organisation and coaching. Parameters are changed each time packages are run so that they can be used several times.

(68) TK solver. Learners using this software begin by listing different elements of the problem on different screens. One is for rules, one for variables, one for dimensional relationship units and a number of others for specialised applications. Once these screens have been filled out, learners can ask the programme to solve the problem for any variable. The package sorts through equations provided by the learner to determine which are appropriate and in which order they must be used to provide the variable value needed to solve the next equation [20, p71].

(69) Turtle graphics. Turtle graphics is a graphic component that is found in LOGO, a computer language. Turtle graphics allows the learner to move a cursor, often called a turtle, around the screen. The geometry based on this tool is called turtle geometry. The turtle can respond to a few simple commands: Forward moves the turtle, in the direction it is facing, a number of units. Right rotates it in its place, clockwise moves it some number of degrees. Back and left cause opposite movements. The number that goes with a command to specify how much to move is called a command input. In describing the effects of these operations, we say that forward and back change the turtle's position. That is the point on the plane where the turtle is located. Right and left change the turtle's heading. That is the direction in which the turtle is facing. The turtle can leave a trace of the places it has been. This is controlled by the commands pen-up and pen-down. When the pen is down, the turtle draws lines [1, p3 and 20, p96].

(70) Visicalc. This is a software for numerical analysis. It permits instructors and learners to explore problems while employing traditional mathematical skills and concepts. Instructors can introduce this software in topics such as functions, equations and general problem solving. The disadvantage of this software is that learners arrive at answers without even seeing the formulae that provide those answers. It is as if learners are not solving the problem, but the software is [20, p70].

(71) Wizard. In this package, learners choose input numbers and the computer provides output numbers. Whenever learners have tried enough input numbers to think they know the pattern. they may make a guess. The computer then gives four numbers as input numbers and has learners predict output numbers. Learners who answer correctly are rewarded with a treasure and allowed to continue the treasure hunt. Learners who respond incorrectly are allowed to try additional input numbers or a different problem. Trying a different problem allows learners. frustrated by a particular problem. to move on [32. p47].

(72) Word-problem tutor. The package is unique in that word problems are presented in two parts. Learners are required to choose the correct operation required for the problem. After the correct response. they are asked to complete the computation. After two incorrect answers. the computer gives learners the correct response with explanation. One strength of this package is that it provides a mixture of problems involving each of the four basic operations so that they can decide which operation is correct for each situation. Each disk contains eighty questions and has four levels of difficulty. Branching moves learners to higher or lower levels or suggests they repeat the lesson depending on the score attained [32. p114].