

**A
Mathematica
Approach
to
Calculus**

by

John T. Gresser

Preface

The last decade of the twentieth century was an exciting and interesting time for the study of mathematics. At the dawn of the new millennium, mathematics has undergone a technological change, which will have a profound effect on the way mathematics is studied, understood and used in the future. What is driving all this change is the introduction and widespread availability of computer software programs that allow one to compute and manipulate mathematical symbols on a computer screen in much the same way that these symbols are computed or manipulated with pencil and paper. Several such programs, generally referred to as Computer Algebra Systems (CAS) or symbol processors are commercially available. One of the most elaborate of these is the program called *Mathematica* that will be used in this manual.

One of the strengths of *Mathematica* is that it is so easy to use. There are of course, some special language peculiarities that one must learn before *Mathematica* can be used to solve problems, but these are held to a minimum. To a large extent, *Mathematica* mimics the ordinary language of mathematics. With no more than an hour or two worth of effort, one can begin to use *Mathematica* to solve mathematical problems. Virtually all of the words used in mathematics are commands or data structures in the *Mathematica* language. Think of a mathematical word, and you will likely find it in *Mathematica*, with the first letter of the word capitalized, but otherwise spelled exactly as it is in English. If not, you will certainly be close enough to find something in *Mathematica*'s great on-line help, which will be discussed shortly.

With *Mathematica*, there is no longer a need to focus attention on the difficulties of carrying out a mathematical computation. The user becomes the director of the production, the creative force that drives the solution, while *Mathematica* does all the work. Problems that were previously inaccessible because of computational complexity, can now be solved using *Mathematica*. Just the freedom to concentrate on the creative aspects of a problem rather than the computational aspects, makes some problems much more approachable.

Mathematicians, indeed, all scientists, are naturally skeptics. Their first reaction to any claim is to ask for a proof, and then they are inclined to look for mistakes and counter examples when they study the work of their colleagues. It is exceedingly important that students of science and mathematics acquire this attitude of skepticism. Is the answer reasonable? Is it correct? Is it unique? Are there easier or alternate ways to get the answer? Is there a counter example? These are questions we should always ask, but in the past there was such a

heavy price to pay to answer these questions, that students typically became accustomed to accepting as correct the outcome of any computation or argument. Understandably, few of us were willing to verify the correctness of an answer with yet another lengthy calculation.

With *Mathematica*, on the other hand, it is usually easy to establish the correctness of a solution by an alternate calculation of some kind. Consequently it is now easier to be a skeptic, and it is just as important to question the validity of a calculation now as it ever was before. *Mathematica*, computers, and human beings are all capable of giving inappropriate answers, incomplete answers, misunderstood answers, or even wrong answers. *Mathematica* provides an excellent opportunity for students of science and mathematics to question their work and to acquire, in the process, this all important skeptical attitude at an early stage in their careers.

Computer Preliminaries

Mathematica is available for Macintosh, Windows, and Unix operating systems. Generally speaking, *Mathematica* needs approximately 16 MB RAM to operate, though it can operate on somewhat less. Even more RAM is clearly beneficial. As you begin to demand more of *Mathematica*, consider the benefits of giving *Mathematica* more RAM to operate, if it is available on your computer.

This manual is written for ***Mathematica 4.1 for the MacIntosh***. *Mathematica* for the **Windows operating system is similar enough** that very little difficulty should be experienced in modifying this manual to the Windows operating system. Rather than risk confusion, however, by discussing both operating systems, this manual is written entirely in terms of just one operating system.

We assume a passing familiarity with a computer using one of these operating systems. Excellent on-line tutorials are available on most computers. Two hours or so spent on such a tutorial should be sufficient preparation for anyone with no computing experience. In particular, we assume that the reader knows how to: 1) open up applications and files, 2) create folders and files and save files, 3) select, click, double click, and drag, 4) use menu bars.

Calculus and Mathematica

This manual is not meant to be a self-contained calculus text, but rather a supplement to a regular calculus text. Topics are covered in a way that is meant to parallel the sequence of topics in a fairly typical calculus text. New *Mathematica* commands are discussed as they arise in the process of solving problems in calculus.

In chapter 2, the primary focus of attention is to learn basic *Mathematica*, by using it to do problems in algebra, trigonometry, and introductory differential calculus. Starting with the integral in Chapter 3, emphasis shifts back to the study of calculus, where it remains for the rest of the manual. As you will see, *Mathematica*, as involved as it may be, is also straightforward enough that it will come to be understood quite naturally, while attention is paid primarily to calculus. It takes some time, however, to adjust to this merger of mathematics, computers, and *Mathematica*. Chapter 2 is meant to nudge us gently in this

direction. Introductory differentiation is well suited for more intense *Mathematica* activity, but the opportunity to use *Mathematica* in this way will come soon enough.

There are several advantages to be gained by postponing the use of a Computer Algebra System until the second calculus course, and this manual is ideally suited for such a program. Hand held graphing calculators are still a mainstay in the mathematical curriculum, and the first calculus course provides a good opportunity to get better acquainted with this excellent tool. By the second course, students who are continuing their study of mathematics are usually ready for more serious work, and their numbers are small enough to make computer lab work more manageable.

Of course, the best way to learn *Mathematica* is not to read about it, but to use it, and so, to a certain extent, this manual is a list of problems. Many of the problems at the end of each section are meant to provide routine experience in using *Mathematica* commands. **Routine problems from your main calculus text can also be used as *Mathematica* practice problems.** You can, with such problems, **look up their answers in the appendix of your main text** to verify that *Mathematica* commands are being used correctly.

Problems are, however, also designed to foster an attitude of skepticism and experimentation. The importance of adopting a skeptical scientific attitude has already been discussed. Answers are, by design, not provided in the back. **How do you know** that you have the correct answer? **Is there another** solution to that equation that is being solved? Is there an interesting feature to a graph under consideration that is **too small to be seen** in a window or that occurs outside the window being used? Problems in this manual sometimes create **unexpected, incomplete or occasionally wrong answers**. In the first few problem sets, you will usually be alerted to look for unexpected results, but eventually, such warnings will not be supplied. **Whenever possible or appropriate, you should supply evidence that your answer is correct.** One of the goals of this manual is to develop a good, skeptical, scientific attitude.

Mathematica can be used to gain a deeper understanding of calculus, by focusing complete attention on an issue of calculus rather than on a computation. Some of the problems are designed with this in mind.

Without question, *Mathematica* can be used to enhance problem solving skills. Just imagine the creative freedom that you will have, when you can focus all of your energy on ideas rather than computations. *Mathematica* makes more substantial and interesting mathematical problems accessible, and this may be its most important contribution to mathematical education. One should strive to get as much experience as possible in doing problems of this sort. Problems designed to improved problem solving skills are included in the exercise sets along with all of the other problems.

Many of the exercise sets have additional problems, labeled “**Projects**” that are somewhat more involved. They range in difficulty from being just longer and more interesting versions of ordinary problems, to being quite difficult. They should be accessible without outside background reading. These problems are designed to enhance problem solving skills, by making use of not only current topics under discussion, but, occasionally, a wide variety of previously discussed topics as well. At least some of them are presented in a playful way and are meant to be enjoyed as well as to be instructive.

These projects, however, should be tackled with some discretion as well. Using *Mathematica* for the first few times is an interesting, but very different way of doing mathematics,

and it takes some time to get accustomed to it and to take advantage of all the opportunities *Mathematica* presents to the user. When you are ready to take on the issue of “putting it all together,” you are encouraged to work on the projects that appear after the exercise sets at the ends of the chapters.

Some Closing Thoughts

It is the nature of computer languages that certain words, phrases, sentences, and punctuations have to be read quite carefully, symbol by symbol. We have made an attempt to emphasize all such structures with **bold face** type. **When you encounter these items, read them with great care.** In addition, whenever a *Mathematica* command is introduced for the first time, it also appears in bold face type.

Explanations of new commands are given just once, when they are first introduced, so if chapters or sections are skipped, it would help to skim over the material that was skipped, looking for, and reading the material on the introduction of new commands. Since new commands appear in bold face type when they are first introduced, this should be relatively painless.

While *Mathematica* is fairly easy to learn, it is a wide ranging language with hundreds of commands effecting many subjects of mathematics. No attempt has been made to make this manual a complete study of *Mathematica*. **You are encouraged to explore on your own and to seek help frequently with *Mathematica*'s on-line help.** Adopt the attitude that **any “mathematical or logical word” under consideration is a word that can be found in some form, somewhere in *Mathematica*.**

Exercises should be presented in an organized, thoughtful and readable manner. Remember that *Mathematica* is a **very good word processor**, and it should be used frequently in the exercises. Projects, in particular, should be done with great care paid to presentation. Think of a project as a term paper or as a report that is going to your supervisor at work. The same matters of presentation should play a role in creating a paper on any subject—including mathematics.

A pencil and paper strategy session can be a useful way to start a *Mathematica* work session. It should be a strategy session, however, and not a complete pencil and paper solution, unless, of course, such a solution is desired. Remember that *Mathematica* will do all of the computations for you. Think of a pencil and paper strategy session as a flow chart, where steps are organized, notation is devised, and so forth.

And finally, above all, remember that *Mathematica* was created by human beings. Human beings make mistakes. *Mathematica* makes mistakes, infrequently, perhaps, but mistakes nevertheless. The same is true for all other software packages. *Mathematica* is under continuous improvement, but it will always have some potential for error. Human beings are, of course, a more common source of mistakes. A small mistake in an input statement can have enormous consequences. Even without an input line mistake, a *Mathematica* computation can be misinterpreted or used improperly by us with grave consequences.

Work with *Mathematica* should be a partnership between a human being and a machine, not a thoughtless ride on a machine. The way to avoid wrong answers is to know mathematics well enough to see the warning signs when mistakes have been made or when

Mathematica is misbehaving. **Make it a practice to verify that answers are correct, especially if an answer “looks” doubtful.** Frequently, answers can be verified very quickly.

In order to keep the main story line simple, brief, and readable, this manual will not always participate in this practice of verifying answers. Make it a habit, when reading this manual, and when doing your own work to **check *Mathematica*'s performance.**

Acknowledgments

I am indebted to the staff of Prentice Hall for its help in preparing this book. Special thanks are offered to Sally Yagan, Editor-in-Chief, College Mathematics; Eric Frank, Calculus and Precalculus Editor; Dawn Murrin, Associate Editor for Calculus and Precalculus; Aja Shevelew, Editorial Assistant, College Mathematics; and Wendy A. Perez, Production Editor. Finally, special thanks go to all of my students, past and present and future, who were, are now, or will be a part of my technology-based calculus program. Their cooperation and sense of humor kept and will continue to keep this project moving forward.

— — — ★ ★ ★ — — —

Solving problems with *Mathematica*, especially more involved problems, can be a rich and rewarding mathematical experience. I hope that this manual and its problems meet with your approval. My e-mail address is included below, because good text books are a community effort, and your comments and suggestions for improvements would be greatly appreciated.

John T. Gresser
Department of Mathematics and Statistics
Bowling Green State University
Bowling Green, OH 43403
jgresse@bgnet.bgsu.edu

To Pamela, Morgan, and Nathan
for their patience
during this writing project

and

to Captain Ralph
off on another adventure.

Contents

1	Some Preliminaries	1
1.1	<i>Mathematica</i> Notebooks	1
1.2	Opening Old Notebooks	4
1.3	Frequent Saving	5
1.4	<i>Mathematica</i> 's On-Line-Help	6
2	Basic Mathematica	9
2.1	<i>Mathematica</i> as a Calculator	9
2.2	Assigned and Unassigned Names	14
2.3	Functions and Expressions	17
2.4	Algebra	20
2.5	Plotting Graphs	24
2.6	The Differential Calculus	26
2.7	Solving Problems in <i>Mathematica</i>	29
2.8	Exercise Set	36
3	Basic Integration	41
3.1	Area and the Definite Integral	41
3.2	Integration and The Fundamental Theorem of Calculus	45
3.3	Integration by Substitution	49
3.4	Numerical Integration	52
3.5	Exercise Set	56
4	Applications of Integration	59
4.1	Sample Problems	60
4.2	Exercise Set	67
5	The Transcendental Functions	71
5.1	Inverse Functions	71
5.2	Logarithmic and Exponential Functions	76
5.3	The Inverse Trigonometric Functions	83
5.4	Exponential Growth and Decay	88
5.5	The Hyperbolic Functions and Their Inverses	90
5.6	Indeterminate Forms and L'Hôpital's Rule	92
5.7	Exercise Set	94

6	Techniques of Integration	99
6.1	Integration by Substitution Revisited	101
6.2	Integration by Parts	104
6.3	Reduction Formulas	106
6.4	Inverse Trigonometric Substitutions	107
6.5	Rational Functions	109
6.6	Improper Integrals	111
6.7	Exercise Set	116
7	Sequences and Series	119
7.1	<i>Mathematica's</i> Limit Command	120
7.2	Sequences	122
7.3	Series	129
7.4	Positive Term Series	131
7.5	Alternating Series and Absolute Convergence	135
7.6	Power Series	138
7.7	Exercise Set	142
8	Plane Curves and Polar Coordinates	149
8.1	Conic Sections	149
8.2	Parametrically Defined Planar Curves	156
8.3	Polar Coordinates	162
8.4	Exercise Set	170
9	Vectors and Analytic Geometry	177
9.1	Vectors, Addition, and Scalar Multiplication	177
9.2	The Dot and Cross Products	181
9.3	Lines and Planes	185
9.4	Surfaces in Space	188
9.5	Exercise Set	197
10	Vector-Valued Functions	203
10.1	Curves in Space	203
10.2	Motion in Space	208
10.3	Geometry of Curves	211
10.4	Exercise Set	213
11	Multivariate Differential Calculus	219
11.1	Functions of Several Variables	219
11.2	Limits with Two or More Variables	223
11.3	Partial Derivatives	225
11.4	Directional Derivatives and the Gradient	232
11.5	Optimization	241
11.6	Exercise Set	258

12 Multiple Integration	267
12.1 Double Integrals	267
12.2 Applications of Double Integrals	276
12.3 Triple Integrals	278
12.4 Applications of Triple Integrals	284
12.5 Exercise Set	290

To The Teacher

A Mathematica Approach to Calculus is intended to be used as a supplement to a main text. There is a significant change in philosophy between Chapter 2 (basic *Mathematica* learned by a quick tour through introductory differentiation) and the rest of the manual where the focus is more on calculus than on software. This matter was discussed in the Preface as well.

Whether *Mathematica* is introduced in the first course or the second, students still need time to adjust to the rigors of merging mathematics, computers, and *Mathematica*. There are certain advantages to postponing the introduction of *Mathematica* until the second course (a more mature audience is one reason), and this supplement is ideally suited for such courses. By working their way through Chapter 2, students can quickly learn basic *Mathematica* and adjust to this computer environment, by reviewing familiar topics from the previous course. While students work on Chapter 2, somewhat on their own, a regular second calculus course can proceed almost at a normal pace. In a fairly short amount of time, students will be ready to use *Mathematica* in their second course activity.

If *Mathematica* is introduced in the first course, then this adjustment period must still be built into a course. One way to do this is to depend initially on a more traditional development of calculus, and then slowly introduce *Mathematica* into the course as it progresses. For such a program, Chapter 2 may well be an acceptable way to start a *Mathematica* based calculus program. Certainly it lacks an in depth study of the elementary properties of the derivative, but it allows calculus to proceed at a fairly normal pace, and it gives students time to adapt to a computer based program.

The mathematical community has responded to the issue of calculus reform in a variety of ways. The reform process is still evolving, but it does not appear to be evolving towards one uniform way to teach calculus. Nevertheless, it is hard to deny that computers could (or should) play at least some minimal role in calculus, if not a maximal role, in all of these schemes. Time, unfortunately, is a major player in this game.

Calculus classes have typically been packed with so much material that there has been little or no room left to add new material. If a computer laboratory component is added to a calculus class, certainly classroom time has to be made available to deal with this addition. It is probably true that some topics have to be dropped, or the level of expectation has to be relaxed on some topics, in order to make room for the additional demands made by computer activity. Therein lies the problem. How do we make time for computer work in calculus?

Time plays a role in another way as well. Mathematicians are extremely busy people. Adding a computer component to a calculus class means taking time out of an already packed work schedule to learn new software, and more important, to learn how to use it in a class room setting. The influence of a computer can produce an overwhelming change in the way mathematics is presented, and this can create a demand for more time than a teacher is able to provide for a course. How are mathematicians suppose to make time for

all of this change?

A software manual may supply all of the necessary computer background, and a main text may supply all of the pertinent mathematical material. If, however, there is nothing connecting the two books, then a teacher has to spend significant class room time and energy to explain the connection. A student may ask, “Why did I do this problem on a computer? I pushed a button and got this answer. Now, what’s the point?”

This manual will certainly not eliminate these problems, but, hopefully, it can soften them somewhat. I believe that it is possible to include a fairly significant computer component to a calculus class with only minor alterations to whatever method is being used to teach calculus. More dramatic reform might be encouraged by some, but then again, each strategy has its advantages and disadvantages.

This manual is mainly about calculus. It contains a fair amount of intuitive—hopefully readable, and student oriented—mathematical material in addition to *Mathematica* material. These same ideas may well appear in a main calculus text, but they also appear in the manual to provide a connection between mathematics and *Mathematica*. Student questions such as those raised above may be answered by reading the manual. If so, then a teacher can spend most of the class room time on strictly mathematical ideas. (*Mathematica*, however, frequently offers a slick way to present an idea.) The computer component of the course can then be handled with a command to, “Read the manual, and do problems”

Some classroom lab time is probably essential. At my home institution, Maple is used in calculus, rather than *Mathematica*, but the situation would be much the same, if *Mathematica* were used instead. In my 5 credit (introductory Maple) second calculus course, the first 3 days of the semester are spent in the lab devoted exclusively to Maple. This is a very intense time for my students, but they are fresh, they are not yet burdened by the demands of their other courses, and a great deal of Maple is learned in three days. After that, classroom time is devoted almost entirely to mathematics, with slightly less than one day per week spent in the lab for the rest of the semester. Maple frequently enters into classroom discussion, but usually in a mathematical way.

In my 3 credit follow-up Maple based, multivariate calculus course, no class room time is spent in the lab. My classroom activity, is almost exclusively mathematical. Maple issues are almost always discussed in some mathematical context. Students are told to read the manual, and to do certain problems. I make myself available on a regular basis for lab activity outside of class.

I confess that my students get frustrated and need human encouragement. Much of this frustration is a communication problem they have with computers. Human beings can understand content, sometimes in spite of what is said to them. A computer (lacking a human ability to interpret) can only understand exactly what it is given. But this is a plus, is it not? What a great way to enforce precise thinking and communication!

Except for the computer component, calculus at my home institution is taught in a very traditional way. A traditional book (read big) is used, and this approach has left a clear impact on the structure of this manual. As you can see, topics are covered in much the same way they would be covered in a traditional calculus course.

In spite of the traditional bent, this manual should also be usable in courses that are more reform oriented. Computers are, after all, one of the principal features of this movement. As a supplement, this manual can simply be regarded as a list of topics and problems, and

so the choice of topics and their order of presentation does not have to be strictly adhered to.

In fact, regardless of the main text you use, and whether your approach to teaching calculus is traditional or more reform oriented, you may wish to avoid a cover to cover study of this manual. A cover to cover study would be nice, but it would require somewhat of a commitment to the *Mathematica* based approach. If material is skipped, new *Mathematica* commands that are introduced in the skipped material will be missed. It is reasonably important to become acquainted with most of these commands, even if material is skipped, because they are introduced, in detail, only once. New *Mathematica* commands appear in bold face type, when they are first introduced, so it should be easy to skim through the skipped material and pick up the new commands. If a command is missed in this way, it will be noticed later on, when the command is used again. If the command use is not clear from context, it should be a simple matter to find the introduction of the command in the missed section, or look it up in *Mathematica's* on-line help file.

In my courses, students need to spend at least two or three hours per week outside of class in the lab in order to finish lab assignments. They are permitted (almost encouraged) to discuss mathematics with fellow class members. So much learning takes place during these sessions, that I am only slightly concerned about how much of their work is entirely their own. In order to make their hard work a bit more tolerable, I assign, whenever possible, the problems in this manual that have been written in a playful way. Students are quick to pick up the playful spirit, and grading their lab assignments can be an enjoyably funny experience.

Lab assignments are turned in over the campus network and graded electronically. It takes a while to adjust to grading in this way, but there are some real advantages. My grading practices with Maple could be used with *Mathematica* in much the same way. I would open a blank Notebook to use as my comment file, and then open student lab assignments, one at a time. (It helps to change the font setting on this comment-notebook to produce text that looks very different from the fonts used on student lab assignments.) I write all of my comments on this comment-notebook and then paste comments onto student lab assignments. Because of the similarity of student mistakes, it takes little time before the same comments are being copied and pasted into several different student work files, and this speeds up the grading process considerably. This practice also encourages detailed comments, since I know they only have to be typed once.

I hope this manual becomes a useful addition to your calculus program. I have included my e-mail address in the Preface. Your comments would be appreciated.