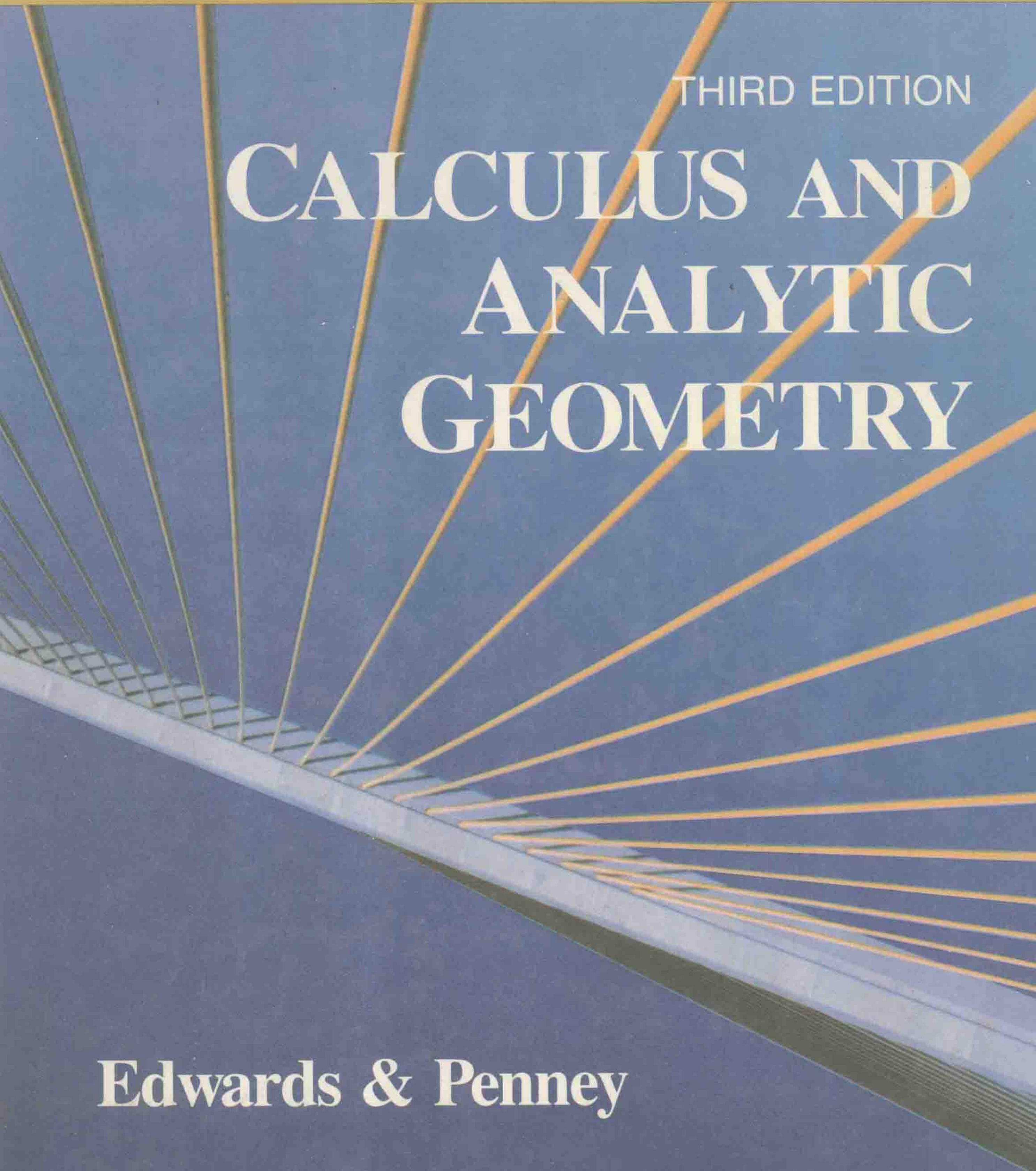
ANNOTATED INSTRUCTOR'S EDITION

Not for Sale to Students

THIRD EDITION

CALCULUS AND ANALYTIC GEOMETRY



Edwards & Penney

ANNOTATED INSTRUCTOR'S EDITION

Calculus and Analytic Geometry

THIRD EDITION

C. H. Edwards, Jr.
The University of Georgia, Athens

David E. Penney
The University of Georgia, Athens

PRENTICE HALL,
Englewood Cliffs, New Jersey 07632

Design: Maureen Eide/Florence Silverman
Illustrator: Ron Weickart/Network Graphics
Manufacturing buyer: Paula Massenaro
Production: Nicholas Romanelli

Calculus and Analytic Geometry, Third Edition
C. H. Edwards, Jr. and David E. Penney

A Division of Simon & Schuster
Englewood Cliffs, New Jersey 07632

Printed in the United States of America
10 9 8 7 6 5 4 3 2 1

ISBN 0-13-111253-8

Prentice-Hall International (UK) Limited, *London*
Prentice-Hall of Australia Pty. Limited, *Sydney*
Prentice-Hall of Canada, Inc., *Toronto*
Prentice-Hall Hispanoamericana, S.A., *Mexico*
Prentice-Hall of India Private Limited, *New Delhi*
Prentice-Hall of Japan, Inc., *Tokyo*
Simon & Schuster Asia Pte. Ltd., *Singapore*
Editora Prentice-Hall do Brasil, Ltda., *Rio de Janeiro*

To the Instructor

This, the Annotated Instructor's Edition, is intended as an additional teaching resource, particularly for the less experienced instructor.

This edition contains a number of marginal annotations (in red). The content and purpose of the annotations is quite variable. Some warn of possible student misinterpretations, some indicate ideas that deserve special or careful treatment, a few provide historical or other background information.

At the end of each section, this edition also provides a list of recommended problems. Indeed, there is a list of recommended odd-numbered problems (if you want the answer in the back of the book to be available to the students) and a list of recommended even-numbered problems (for which no answers appear in the text). The problems have been selected on the basis of covering the material of the section, providing a mix of drill and less routine problems, and avoiding problems that are too challenging. In many cases, the more interesting and more challenging problems are also indicated.

In addition to the annotated text, this Instructor's Edition includes opening essays on the following topics:

1. Calculus, Computing, and Writing.
2. Calculus: A Computer Laboratory.
3. Calculators: From Square Roots to Chaos.
4. Computers and Calculus (use of a graphics software package).
5. Calculus: A Core Course.

We hope that these essays will be of particular interest to instructors who are interested in reshaping the calculus curriculum, and in the possible use of calculators and computers in the teaching of calculus.

C.H.E. Jr / D.E.P.

Calculus, Computing, and Writing

Thirty-five years ago this fall I enrolled as a first-quarter freshman intending to major in engineering. In those days every college student took a one-year freshman English composition course featuring weekly themes that were graded quite savagely. This was followed by a year-long sophomore literature course in which weekly themes were again the principal mode of student expression.

In addition, I and my fellow engineering students that first fall quarter took a required course called "Engineering Problems" in which we learned how to operate a slide rule, how to check systematically the accuracy of our simple algebra and arithmetic, how to keep track of dimensions and cancel units correctly, and how to write a laboratory report explaining in plain English how our experiment had (or had not) worked. With a modern calculator in place of the slide rule, these are precisely the things—never mind the finer points of precalculus mathematics—that I would like all of my calculus students to have learned somewhere before appearing in my class.

I need not belabor the inferior preparation of today's students as compared with that of the students in those halcyon days when we ourselves sat on the other side of the lectern. Each of us has a pet suggestion as to the first corrective action that should be taken. But perhaps one could do worse than require that each student come to the campus for a pre-college summer quarter studying nothing but the same engineering problems text I studied in 1954: F. C. Dana and L. R. Hillyard, *Engineering Problems Manual*, fourth edition, New York: McGraw-Hill, 1947. There is probably a message in the fact that this fine text (first published in 1927) is no longer in print. Reading it in 1989, one is struck by the timelessness of such bits of wisdom as

- "The average student wastes much time and effort and gets lower grades than he should because of inaccuracy in figuring" and "the habit of [careless] inaccuracy saps confidence and leads to future difficulties" (page 7).
- Prominent among "time wasters are the formula worshipers . . . who spend more time hunting a magic formula than they would need to analyze the problem piece by piece using simple familiar methods and calculations" (page 9).
- "There is a close connection between slovenly thinkers and slovenly [writing]", and a close correlation between the appearance of the paper and the mental habits of the writer (page 11).
- The person "who resents difficulties, the quitter, the whiner, the leaner, the bluffer, gravitates sooner or later to a less demanding way [than engineering] of making a living" (page 13).
- "The ancient art of 'apple polishing', sometimes known by other names, is one that is sadly neglected in these modern times; . . . it is a way of getting acquainted with teachers" (page 20).
- (Under the heading "How to flunk out.") "It cannot be said too often that the ability to understand and solve problems *does not* come by

memorizing formulas . . . Formulas are not substitutes for thought, nor can they be used safely by blindly substituting data assumed to fit them" (page 27).

- "Do not make fine, faint, gray marks. They are sufficient reason for complete rejection of a paper by the checker even though the paper is otherwise acceptable . . . Bear down on the pencil . . . Do not scribble . . . Keep papers clean . . . Avoid crowding . . . Watch your spelling" (pages 46–49).
- "Use horizontal-bar fractions to show division". Inclined-bar fractions cause too many blunders (page 55).
- "To drop significant figures without good reason is a blunder that reduces the precision of the result. To record a string of doubtful figures beyond those justified is downright dishonesty" (page 130).
- "A working knowledge of the right triangle is of vital importance" (page 146).
- "The two principal operations [of calculus] are complementary . . . One is finding the rate of change of some variable; the other is determining the total change . . . Throughout the entire subject of calculus the student is really working with differential equations" (page 193).

Of course the book as a whole is more substantial and less moralistic than this arbitrary selection of quotes may suggest. But where can today's students find such still-needed advice so plainly stated? Certainly not in modern mainstream textbooks! My suggestion of a pre-college summer quarter prep course will surely go uniformly unheeded. So where in today's curricula can such instruction possibly find a place?

I happened upon an unexpected answer this past year when I set out to include a computer lab as a year-long feature of my calculus course. The daily homework I customarily assign was replaced with two- or three-week computer projects. Each project consisted of a collection of related problems with a common theme (such as Newton's method or Simpson's rule) for students to investigate in the computer lab, and then to organize their results in a written report. The very first project consisted of several problems like the following one, intended both for review of high school geometry and for familiarization with the computer and some simple programs that were provided:

> A 100-ft tree stands 20 feet from a 10-ft fence. Then the tree is "broken" at a height of x feet, so that it just grazes the top of the fence and touches the ground (on the other side of the fence) with its tip. Apply the Pythagorean theorem and the proportionality of corresponding sides in similar triangles to show that x satisfies the equation
>
> $$x^3 - 68x^2 + 1100x - 5000 = 0.$$
>
> Then use the tabulation and bisection programs to find the possible value(s) of the distance x accurate to 4 decimal places.

[The cubic equation above has three positive real solutions, but only two of them are the desired "possible values".]

The instructions for each project included the following sort of general directions.

> Write a mini-report describing your work on each problem. First explain in your own words the mathematical background to your numerical work. Then tell precisely what you did with a computer and how it worked out. Use complete sentences written in plain English. Recount both your failures and your successes, and describe any particular anguish or joy you experienced. Insert relevant computer results at appropriate spots in your narrative, carefully selecting the number of decimal places for the computer to print so as to highlight your results and conclusions.

As the work on each project progressed, I invited class discussion of issues raised both by their computer lab work *and* by their attempts to describe it in their reports. The students' questions brought to light a number of ground-level topics that are vital for numerical work in mathematics, but which seem not to fit in the standard calculus curriculum—accuracy and significant digits, dimensions and units, etc. I was prompted to deal more than previously with matters of good mathematical taste; for instance, why it might be good to show 6 decimal places if you're interested in 4-place accuracy, but not to show 16 places if 10 of them are meaningless (however willing is the computer to mindlessly print them out). In this way the project report emphasis served to flesh out the course and add a concrete dimension that often is missing from classroom mathematics.

As opposed to the mathematical and computational components, the *writing component* of this activity played a much more substantive role than I had anticipated. The "explain the mathematical background" direction seemed to call for conventional exposition, but the students were surprised by my insistence that they really *explain* their reasoning in complete (English language) sentences, and by my refusal to accept the typical student mathematics paper consisting solely of unexplained equations and computations. Perhaps some of them understood for the first time that an equation is not just an inert collection of marks on paper, but is itself a sentence that makes an assertion, whose meaning may well depend upon the context (which therefore must be made clear). Some clearly were shocked by my emphatically stated belief in a positive correlation between the orderly appearence of the paper and the orderly thought of the writer. I myself gained a new appreciation of the phrase "writing to learn mathematics" (though those of us who write mathematics books are perhaps the most notorious practioners of the principle). For a more systematic discussion of this concept and its implications, see George D. Gopen and David A. Smith, "What's an Assignment Like You Doing in a Course Like This?: Writing to Learn Mathematics", *The College Mathematics Journal*; to appear.

My instructions were deliberately evocative, but I was surprised that they stimulated a degree of student fluency that I have never before seen in student mathematics papers. Students are uncertain how to write well about mathematics (perhaps because they've seen so many examples of bad writing in their mathematics textbooks) but the direction to "tell what you did with the computer" called for them to describe *their own experiences*, and on a higher cognitive level than telling "what you did during summer vacation".

Students like to talk (and even to write) about themselves, and with the proper motivation can be quite loquacious. And though they doubt the usefulness of the perennial summer vacation theme, most can readily perceive the usefulness of learning to describe computer work—thus a new twist to the ageless question of what all this is good for!

The instruction to "recount both your failures and your successes" is particularly important. It seems to students that standard mathematical exposition deals solely with what "works", even though why it works may never be made clear (at least to them). A new concern with what does *not* work—and why it doesn't, and hence what might be tried instead—brings students closer to real mathematics than most of them have ever been before.

Much of the most eloquent writing came under the heading of "joy and anguish". Many students reported occasional feelings of antagonism toward the computer, and often described its refusal to cooperate in quite anthromorphic terms. One student insisted that his computer would not perform properly unless sworn at most viciously. Another described the exilaration he felt when his series summation program finally worked and produced correct sines and cosines—at that instant he exchanged winks with the previously inert portrait of Isaac Newton hanging on his dormitory room wall.

As the year progressed, I learned a few more tricks for inviting the feeling of active personal involvement on the part of the students. A third-quarter project on the 2-dimensional Newton's method included the following problem.

> You land in your space ship on a spherical asteroid. Your partner walks 1000 feet away along the smooth surface carrying a 10 ft rod, and thereby vanishes over the horizon. When she places one end on the ground and holds the rod straight up and down, you (lying on your stomach to rest from the arduous journey from Earth) can just barely see the tip of the rod. First determine the radius of the asteroid. Then determine, if you have enough spring in your legs to jump 4 feet high on Earth, how high could you jump on this asteroid. Assume that the earth is a sphere with radius 3960 miles having the same uniform density as the asteroid.

Several students took the bait and described in fanciful detail what had happened in the course of their "arduous journey from Earth", and why they consequently were so tired when they finally arrived on this lonely asteroid. One even cast his report largely in the form of a dialogue between himself and his "partner" (not unlike Galileo and Simplicio). Perhaps it is pertinent that this last student, who earned the class's top grade on this particular project, had begun the year terribly concerned about merely passing the required calculus course, describing himself as having always had trouble with mathematics.

In the end, I surely had learned no less than the students had. I had begun the year with the intent of including a certain sequence of computer topics as described in the manual *Calculus: PC Companion* (Prentice Hall, 1990). I ended with the view that, however interesting some of us may find computing in calculus, it's greater importance may lie in the fact that it provides a natural vehicle for "writing to learn mathematics" on the part of those many students who cannot (and probably should not) get this experience by writing theorems and proofs. I now realize that this was the primary benefit (and very likely the purpose) of those laboratory reports I labored to write back in engineering school. Some of the best and oldest lessons must be relearned continually, not only by our students, but also by we who have known them before.

C. H. E., Jr.

Calculus: A Computer Laboratory

In the calculus "core course" I taught last year the usual daily homework assignments were replaced with computer projects that constituted (from a grading standpoint) one-third of the course. For each assigned project the students had two to three weeks to do the necessary computer work and to write up their results in the style of a "lab report". The book

> C. H. Edwards, Jr., *Calculus: PC Companion*,
> Englewood Cliffs, N.J.: Prentice Hall, 1990

is the result of my efforts during the past several years to develop source material for use in such projects.

This manual consists of nineteen computer "lessons" that span the computational highlights of introductory calculus, from functions and graphs through infinite series and elementary differential equations. In addition, four supplementary "interludes" show how a combination of computational power and the elementary concepts of calculus leads to such exciting applications as fractals and the Mandelbrot set.

Using Computers in Calculus

The goal of the manual is to encourage the use of computing to enliven the teaching of calculus. It can be used either

1. As a computer supplement to a standard calculus course, or as the manual for a computer laboratory associated with such a course;
2. As the text for an experimental course on calculus with computing; or

3. For individual study by calculus students who are interested in computing.

However, it is specifically designed as a "computer supplement" that students can use in a standard calculus course, with as little or as much explicit involvement and supervision as the instructor may care to provide. Each lesson is constructed in such a way that a student should be able to work through it routinely, seated at a personal computer with the manual open.

Consequently, an organized official "computer laboratory" need not be associated with the course. On many campuses, personal computer clusters are distributed about the campus and are available for student use on a walk-in basis. In this setting computer assignments—ranging from daily homework to three-week projects—can be given, with the students expected to go forth on their own to find computers and to complete the work. The problem sets are intended to include ample material on which to base such assignments.

To Program or Not to Program?

Instead of writing their own programs, students following this manual will use the BASIC programs that are included on the MS-DOS diskette that accompanies the manual. Therefore no prior knowledge of programming is assumed. Lesson 1 begins by telling the student how to start from scratch by loading BASIC, and an appendix presents a brief BASIC tutorial for those who want or need it.

Most of the programs on the disk are listed in the manual itself. They are an integral part of the exposition, and hence are designed as much to be read and appreciated as to be used and enjoyed. In seeing how we instruct the computer to "do calculus," students learn how to do it themselves.

This approach is based on the principle that some sort of active student participation in computer work is required if learning is to take place. Consequently each program in the manual requires the student user to "do something" before and/or during its execution—if only to edit the line of the program that defines the function $f(x)$ for which Newton's method is to be applied to solve the equation $f(x) = 0$, and to "press any key" to see another iteration. By slightly altering existing programs so as to obtain desired results—perhaps only to change the number of decimal places displayed—even those students with no previous computing experience will gradually absorb the quite modest BASIC literacy that is needed. With this approach, I have found the question of student programming to be a non-issue, and the conscious teaching of programming to be (thankfully) quite unnecessary.

Nevertheless, I have much in common with those who feel that the mathematical benefits of computing are maximized if and when students can program for themselves the numerical algorithms of calculus. With an appropriate class, an enticing possibility would be the translation by students of our BASIC programs into nicely structured Turbo Pascal (especially if they've had advanced placement computer science) or into another dialect of BASIC, perhaps True BASIC or QuickBASIC for the Macintosh.

Range of Topics

The contents of the manual are indicated by the list of lesson titles:

LESSON 1: Algorithms, BASIC, and Computing
LESSON 2: Tabulation of Functions and Solution of Equations
LESSON 3: Bisection and Interpolation
LESSON 4: Graphs of Functions and Equations
LESSON 5: Iteration and Sequences
INTERLUDE: Successive Substitutions and the Mandelbrot Set
LESSON 6: Sequences and the Limit Concept
LESSON 7: Limits of Functions
LESSON 8: Numerical Differentiation
LESSON 9: Newton's Method
LESSON 10: Maximum-minimum Problems
INTERLUDE: Newton's Method and Computer Graphics
LESSON 11: Riemann Sums and Numerical Integration
LESSON 12: The Trapezoidal Approximation
LESSON 13: Simpson's Approximation
LESSON 14: Summation of Infinite Series
LESSON 15: Power Series and Transcendental Functions
INTERLUDE: Extrapolation, Archimedes, and Getting Something for Nothing
LESSON 16: Parametric Curves
LESSON 17: Partial Derivatives and Newton's Method
LESSON 18: Critical Points in Two Dimensions
LESSON 19: Differential Equations and Euler Methods
FINALE: Bounded Populations, Iteration, and Chaos

Beginning with Lesson 6 the sequence of topics parallels (roughly) those in a "core calculus" course—limits, derivatives, Newton's method, maximum–minimum problems, numerical integration, infinite series, multivariable calculus, and differential equations. But perhaps the first five lessons need some comment. Lesson 1 is a low-key introduction to computing and the idea of a computational algorithm. Lesson 2 on tabulation of functions and the solution of equations gives students a more concrete and numerical familiarity with the function concept than most calculus students begin with. Lesson 3 on bisection and interpolation introduces the idea of iteration that plays a central role in automated computing. Lesson 4 introduces the student to function-graphing programs that are available on their diskette, and which can be used to solve equations visually by "zoom" techniques. Lesson 5 on iteration and sequences introduces the concept of convergence in the natural case of a sequence of approximations to the solution of an equation.

Thus limits of sequences appear earlier and more tangibly than limits of functions in a computer approach to calculus. Limits of sequences are treated more generally in Lesson 6. This leads naturally to limits of functions and to derivatives in Lessons 7 and 8, and thereafter the order of topics is similar to a standard calculus course.

Each block of five lessons is followed by an "interlude" of optional material that may be used for supplementary reading and as a source of inspiration for students who are especially computer-oriented. For instance, the first interlude shows how the iteration $x_{n+1} = g(x_n)$ with the inocuous-looking function $g(x) = x^2 + c$ leads to the fantastic Mandelbrot set, and the second interlude shows how an investigation of the mundane-appearing "cork ball equation" (Lesson 2) leads to fractals and the idea of chaos.

Requirements

In regard to hardware and software, students are assumed to have access to MS-DOS (that is, IBM-compatible) personal computers, together with the Microsoft BASIC that is generally available (either formally or informally) in computer labs on most campuses. The program listings and illustrative numerical results included in this manual have been produced with both IBM and compatible non-IBM personal computers, and were photocopied directly so as to avoid the possibility of printing errors.

I attempt throughout to reinforce classroom learning of calculus by including reasonably complete discussions and alternative explanations of the calculus concepts that are used and illustrated in the manual. This manual and its predecessor *Calculus and the Personal Computer* (Prentice-Hall, 1986) provide a record of my attempts to bring the power and excitement of personal computing to bear on the development of a livelier calculus course. This is a continuing effort, and I invite student and faculty comments and suggestions.

C. H. E., Jr.

Calculators: from Square Roots to Chaos

Our teaching of mathematics should reflect the current revolution in the role and practice of mathematics in the world around us. This revolution is driven largely by computational technology. The computational power now offered by calculators and computers encourages a concrete problem-solving approach that emphasizes the formulation of mathematical models and the solution of equations. It permits the inclusion of realistic applications that otherwise would be impractical because of the amount of numerical calculation involved. Perhaps most vital is the way computational experiences can transform students from passive spectators to active participants in the mathematical experience. In this note we illustrate this theme with some calculator examples that can be employed to exhibit the vitality of contemporary mathematics as a living subject.

Consider first the problem of approximating the square root of a number A, for which we all learned in school a dreary and long-forgotten paper and pencil method. Of course our calculator has a square root key, but how does the calculator do it? More to the point, how can *we* solve the equation

$$f(x) = x^2 - A = 0 \tag{1}$$

to find the *numerical value* of its solution $x = \sqrt{A}$? We would like our students to recognize instantly that the question simply is: Where does the curve

$y = f(x)$ cross the x-axis? But with one of the newer graphics calculators (such as the Casio fx-7000G) we can quickly plot this curve and note (for instance) which consecutive integer interval contains its intersection with the (positive) x-axis. Moreover, we can use "zoom" techniques to look at successive magnifications of this intersection and thereby actually "eyeball" the solution, with each magnification by a factor of 10 producing an additional decimal place of accuracy.

Alternatively, we might employ the ancient Babylonian averaging iteration

$$x_{n+1} = \frac{1}{2}\left(x_n + \frac{A}{x_n}\right). \tag{2}$$

to start with an initial guess x_0 and generate a sequence $x_1, x_2, x_3, \ldots$ of closer and closer approximations to the number $\sqrt{A}$. This introduces the general idea of solving numerically an equation of the form $f(x) = 0$ by writing it in the special form $x = g(x)$ that suggests an appropriate iteration. Equations (1) and (2) illustrate this idea with

$$f(x) = x^2 - A = 0 \quad \text{and} \quad x = g(x) = \frac{1}{2}\left(x + \frac{A}{x}\right) \tag{3}$$

Even so simple an interation as the one in (1) can be tedious using paper and pencil alone, but a calculator with a single programmable function key will make short work of it. For instance, suppose that in order to find $\sqrt{2}$, we program this function key to take the entered number x, then calculate and display the value of the function $g(x) = (x + 2/x)/2$. If we first enter the number 2 itself as an initial guess, then press our g key five times, we see the numbers 1.500000, 1.416667, 1.414216, 1.414214, 1.414214 appear successively in the display, thus telling us that $\sqrt{2} \approx 1.414214$ accurate to 6 decimal places.

Some quite inexpensive scientific calculators now sport function keys, but with a graphics calculator like the Casio with a "replay" feature, we needn't even to program a function key. If we simply enter the lines

$$2 \rightarrow \mathrm{X} \quad \text{and} \quad (\mathrm{X} + 2/\mathrm{X})/\mathrm{X} \rightarrow \mathrm{X}$$

and then press the Execute key five times in succession, we get the display

```
2 → X
        2.000000
(X + 2/X)/2 → X
        1.500000
        1.416667
        1.414216
        1.414214
        1.414214
```

Starting with the number 2 in the x-memory (as a crude initial guess), the current value of x is replaced by the Babylonian average $(x + 2/x)/2$ with each press of the Execute key, thereby carrying out our iteration in especially painless fashion.

Merely finding square roots as the Babylonians did (though without graphics calculators) 2000 years ago is scarcely revolutionary. But the iterative approach of Equation (2) leads to tools with which students can attack successfully a wide variety of interesting and important problems, and can even lead to exciting developments near the frontiers of contemporary mathematics.

Consider, for instance, a cork ball of radius 1 ft and specific gravity $\frac{1}{4}$ floating in water. We ask for the depth x to which the ball sinks. Using the Pythagorean theorem and a volume formula of Archimedes, it turns out (see Example 2 in Section 3.9) that x satisfies the *cubic* equation

$$f(x) = x^3 - 3x^2 + 1 = 0. \tag{4}$$

The quadratic formula that students memorized in high school is no help, but we want them to learn that a modern calculator enables them to solve (virtually) *any* equation. For instance, if their calculator has a programmable function key, then they can quickly calculate sufficiently many values of $f(x)$ to spot sign changes in the intervals $(-1, 0)$, $(0, 1)$, and $(2, 3)$. Of course the physical situation which (4) models shows that the solution in $(0, 1)$ is the one we seek.

Once we have the desired solution bracketed, the Solve key on a calculator like the Hewlett-Packard 28S will quickly give us an accurate 10-digit approximation to the actual root. Alternatively, after several magnifications with a Casio (or the similar Sharp EL-5200) we see (literally) that the graph crosses the x-axis very near the point $x \approx 0.6527$, so the ball sinks in the water to a depth of about

$$(0.6527 \text{ ft})(12 \text{ in/ft}) \approx 8 \text{ inches}.$$

But this particular result is only the beginning of the story. Newton generalized Babylonian iteration to apply to essentially any equation $f(x) = 0$. Given $f(x)$, *Newton's method* (Section 3.9) tells how to make a propitious choice of a function $g(x)$ so that, starting with an appropriate initial guess x_0, the iteration

$$x_{n+1} = g(x_n) \tag{5}$$

leads rapidly to a solution of $f(x) = 0$. For the cubic equation in (4) the iteration function of Newton's method is

$$g(x) = x - \frac{f(x)}{f'(x)} = x - \frac{x^3 - 3x^2 + 1}{3x^2 - 6x}. \tag{6}$$

Once we know the proper interation function $g(x)$, we can start with an initial guess x_0 and generate the sequence $x_1, x_2, x_3, \ldots$ of (hopefully) closer and closer approximations to a solution by plowing each new x_n back into $g(x)$ to calculate the next approximation x_{n+1}. For instance, with $g(x)$ defined as in (6), the different initial guesses -1, $+1$, and $+3$ lead to the three different (approximate) solutions -0.5321, 0.6527, and 2.8794 of the cork ball equation in (4).

Once we know Newton's method and therefore can find an effective iteration function $g(x)$ to use, we can just as easily solve (virtually) *any* equation $f(x) = 0$ by iterative evaluation of $g(x)$, starting with a plausible initial guess. Armed with this tool, a teacher can present *real* problems, and need never again stoop to inquiring about the width of the foot-path around the flower garden.

A combination of the numerical and graphical methods that have been mentioned can be employed to place a healthy emphasis on problem-solving by interactive investigation rather than the use of inflexible rote methods. These techniques are ideal for inculcating the spirit of inquiry that is central to real mathematics, but too often is missing from the mathematics classroom. For instance, given an iteration function $g(x)$, we can ask what happens when different initial guesses x_0 are used to "seed" the iteration. Students can be led to discover for themselves that 3 different initial guesses that are arbitarily close together can lead to the three different real solutions of the cork ball equation in (4). For instance,

- the initial guess $x_0 = 1.648$ yields the root 0.6527;
- the initial guess $x_0 = 1.649$ yields the root 2.8794;
- the initial guess $x_0 = 1.650$ yields the root -0.5321.

In particular, these three different initial guesses yield the following Casio fx-7000G displays:

1.648 → X	1.649 → X	1.650 → X
1.6480	1.6490	1.6500
X−(X·X·X−3·X·X+1	X−(X·X·X−3·X·X+1	X−(X·X·X−3·X·X+1
)/(3·X·X−6·X) → X	)/(3·X·X−6·X) → 1	)/(3·X·X−6·X) → 1
0.1127	0.1092	0.1058
1.6227	1.6675	1.7156
0.1927	0.0414	-0.1839
1.0499	4.1344	-0.9245
0.6658	3.3642	-0.6343
0.6528	2.9922	-0.5421
0.6527	2.8878	-0.5322
0.6527	2.8794	-0.5321
0.6527	2.8794	-0.5321

The initial guesses are close together, but the solutions they yield are not! Thus the solution obtained by iteration appears to depend unpredictably or "chaotically" upon the initial guess. This, of course, is the phenomenon of *chaos* that is so prominent in contemporary science and mathematics. If we keep an appropriate pictorial record showing in different colors the particular solutions obtained for a wide variety of different initial guesses, we ultimately see a *fractal* on the screen that reveals the order underlying the chaos. For further details see Interlude B of C. H. Edwards, Jr., *Calculus: PC Companion*, Englewood Cliffs, N.J.: Prentice Hall, 1990.

One can forget about solving equations, and simply ask what happens with the iterates produced by the iteration (5) with a given function $g(x)$. Each new function $g(x)$ provides the basis for a possible new student "research" project. For instance, consider the simple function

$$g(x) = kx(1 - x) \tag{7}$$

(with k a given constant). Ask a student with a Casio calculator to enter the commands

```
0.5 → X
2.5·X·(1-X) → X
```

corresponding to $x_0 = \frac{1}{2}$ and $k = 2.5$, then press the Execute key ten or twenty times to see that $x_n \to 0.6$. Then suggest trying it again, first with $k = 3.25$ and next with $k = 3.5$. (In the former case the nth iterate x_n will oscillate between *two* distinct values, and in the latter it will cycle among *four* distinct values!) For the rest of the story—period doubling and all that—see the Finale to *Calculus Companion: The PC*.

Finally, suggest that students explore iterates of the lowly looking function

$$g(x) = x^2 + c \tag{8}$$

with different values of the constant c (starting with initial value $x_0 = 0$ in each case). The question is how the sequence $\{x_n\}$ of iterates obtained depends upon the chosen value c. Numerical experimentation with

```
0 → X
X·X+C → X
```

will show that the sequence of iterates appears for some values of c (such as $c = 0.1$ and $c = -0.5$) to "settle down" and approach a definite limiting value, while for others (such as $c = 1$ and $c = -3$) it appears to "diverge to infinity"—that is, the value x_n seems to get larger and larger without any limit. To discover the pattern as to which values of c are which, it is necessary to experiment (if you have a computer or an HP-28S) with complex as well as real values of x and c. The stunning color pictures of *Mandelbrot sets* seen in glossy magazines and books (such as H. O. Peitgen and P. H. Richter, *The Beauty of Fractals*, New York: Springer-Verlag, 1986) are produced by coloring different c-points with different colors, depending on the speed with which the resulting sequence of iterates diverges to infinity. For a detailed description and a simple BASIC program that produces a passable (black-and-white) Mandelbrot set, see Interlude A of *Calculus Companion: The PC*. The Mandelbrot set can even be generated and printed using the HP-28S; see pages 337–339 of W. C. Wickes, *HP-28 Insights*, Corvallis, Oregon: Larken Publications, 1988.

C. H. E., Jr.

Computers and Calculus

Using computers as a teaching aid for calculus has been evolving for the past ten years. The advent of the microcomputer and, more recently, the data tablet for overhead projection have made in-class use of computers very attractive. Most software has dealt with the graphical aspects of calculus, but modeling, simulation and computer algebra are areas of rapidly increasing activity. Computer graphics is typical of computer classroom use, and this discussion will concentrate on how good, fast graphics capability can be used effectively in a calculus course, and how the course is changed by this tool. The graphics software package $EPIC_{III}$[1] will be used as a concrete example of such capability.

Software features assumed for this discussion include the ability to input any function, including multiple section functions, graph the function, superimpose graphs of several different functions, draw the graph of the first and second derivatives, draw tangent lines and secant lines, illustrate regions of integration, and draw graphs of polar and parametric equations. A "zoom" feature is extremely useful and the user must have control of the "graphics window"—that is, the intervals along the x and y axes in which the graph is drawn. The software must be easy to use.

Classroom use of computers generally falls into one of three categories: (1) planned demonstrations, (2) spontaneous demonstrations, and (3) student activities. (Each category will be fully discussed in following paragraphs.) The term **Planned Demonstrations** is used to describe instructor-given presentations prepared before class. In class, the computer graphics output is displayed on a large screen via overhead projection while the instructor discusses important features. **Spontaneous Demonstrations** refer to the use of a computer as an aid to answer a student question as soon as it is asked. This assumes a computer is always available during class. **Student Activities** describes students performing investigative work with directed computer use. This requires student access to a computer, and consequently may be "lab" activity. These classroom activities can be combined with homework assignments designed to be worked with a computer.

The examples discussed below are generally computer implementations of teaching methods utilized by most mathematics instructors. That is, the computer software has been designed to mimic instructor activity to a small degree. The computer provides speed, accuracy and impartiality. It does not teach, but it may answer student questions, if the student is able to phrase the question properly.

An important question to answer before describing computer-related classroom activities is why use the computer at all. If it can do only what a well-prepared instructor can do, why make the effort to incorporate computers? From the author's viewpoint, there are at least four separate aspects to the answer. The first part of the answer lies with the key words of the previous paragraph: speed, accuracy and impartiality. Computers can draw graphs very quickly, seldom make an error, and eliminate the subtle step of requiring the student to believe the instructor has not made an error! A second component to the answer is the animation ability of computers. For example, many calculus questions involve limits, which intrinsically contain changing quantities. Often an illuminating presentation of these changes can be provided by some sort of animation. The third ingredient to the answer to the question of why use computers is that the computer is instrumental in making students active rather than passive learners. Its rapid response time encourages students to test ideas and methods, rather than wait for the