

Computer Display Tools for Lasers and Optics

A. E. Siegman

Edward L. Ginzton Laboratory, Stanford University
Stanford, California 94305-4085

1. INTRODUCTION: PERSONAL COMPUTERS IN EDUCATION

As I began preparing the talk on which this article is based its subject matter began to evolve, moving from the original topic of computer display tools for optics education toward the more general topic of personal computers in higher education, especially in science and engineering; and so this article will be directed more broadly to that topic. The starting point for this discussion is that modern personal computers or desktop computers have unparalleled capacities for:

- Fast, tireless and error-free calculation of complicated numerical results, together with instant graphic display of those results;
- Extensive symbolic manipulation of mathematical symbols, including algebra, calculus, matrices, logic operations, number theory, and other forms of higher mathematics and theoretical physics;
- Dynamic displays of physical phenomena for the visualization and understanding of those phenomena, as well as capabilities for multimedia displays including graphics, color video, and sound;
- And finally, “knowledge navigation,” a term Apple Computer uses to include such capabilities as extensive database storage, searching and indexing, and hypertext capabilities.

All of these capabilities then lead to unparalleled educational opportunities for both students and instructors.

To review some of the potential application of these personal computer capabilities to educational purposes I will adapt and extend a list of such applications that was originally developed in a report entitled *Computing in Higher Education: The Athena Experience*, prepared by the Massachusetts Institute of Technology in 1985. According to my extended version of their list, a personal computer can function in an educational environment:

1. As a *teacher*, providing computer-aided instruction and interactive teaching, tutoring and testing.
2. As a *dynamic textbook*, with hypertext capabilities and dynamic illustrations.
3. As a complete *learning environment*, incorporating the entire range of multimedia capabilities including text, illustrations, video and sound.
4. As a *library* or “*knowledge keeper*,” maintaining vast databases of text, images and sound, together with facilities for knowledge storage and searching.
5. As an *academic administrator*, tirelessly taking care of record keeping, class scheduling and bookkeeping.
6. As a *laboratory instrument* for recording, processing and displaying data from laboratory experiments.
7. As a “*virtual laboratory*” in which simulated laboratory experiments can be displayed and carried out on-screen, without requiring physical laboratory equipment (“chemistry without test tubes”).
8. As a *communications medium* including networks, electronic mail, and the wonderful worlds of electronic bulletin boards and newsgroups.

9. As a *mediator* or *facilitator* for discussions, negotiations and what is coming to be called “group work.”
10. And there is of course the omnipresent use of the computer for *recreation* and *entertainment*, extending from the present computer games to the “Nintendo classroom.”

Continuing this list with some topics that seem of most interest to me, we might also employ a personal computer in education:

11. As a *simulator of complex systems*, for calculating and displaying the dynamical behavior of complicated real physical systems—and the behavior of imaginary or nonphysical systems or models as well.
12. As already mentioned, as a *visualization tool* for dynamic on-screen displays of scientific and engineering phenomena or scientific and engineering data, in the form of dynamic textbook illustrations or dynamic “computer blackboard” displays.
13. And finally, the computer as a *super-calculator* or a “*personal math wizard*” for carrying out numerical calculations, plotting and graphing, and doing symbolic algebra including calculus, matrices, and advanced math and theoretical physics.

The specifications that the MIT Athena program defined for an educational personal computer to satisfy these applications five years ago were something like:

- o 1 Megapixel of screen display
- o 1 Megabyte of memory
- o 1 Megaflop of computing power

and all of this for approximately \$3,000 total cost. This was often referred to as the “3-M machine” for obvious reasons. Today we might expect as an only slightly optimistic extension of these capabilities to have:

- o 10 Megabytes of memory
- o 10 Megaflops of computing power
- o 100 Megabytes of hard disk storage
- o 100 (1000?) dots per inch of printing capability
- o 1 Gigabyte of optical disk or CD-ROM storage

for perhaps as low as \$1,000 total cost (at least eventually). And, we would like or hope to see a machine with this capability on essentially every student and faculty member’s desk in the near future.

2. SOME PERSONAL PREJUDICES

At this point I might express some of my personal prejudices about a few of the capabilities and applications that have just been mentioned:

1. I am first of all a fervent advocate of “one student, one computer”—in other words an individually owned, free-standing (although almost surely networked) computer on every student and faculty member’s desk, rather than the diskless networked work stations with central file servers and centralized software storage that the Athena program originally envisioned. One should not have to travel to a central computer cluster in order to access a computer; it should be on one’s desk.

2. My second belief is that “books will survive,” in part because, despite dynamic graphics and multimedia, in my observation people simply do not like to read on screen (though they may search, or calculate, or edit on screen). Therefore the personal computer will play an auxiliary though still very exciting role in education, but it will not replace conventional textbooks.

“... It's going to be a long, long time before a computer screen replaces the newspaper or books.”

*Bill Gates (Founder of Microsoft)
EDUCOM Conference, October 1987*

3. Finally I believe the major roles for personal computers in higher education (outside of research applications, and purely administrative applications) will consist of:

- (a) Supplemental tutoring and testing, motivated in large part by reasons of economy and efficiency in handling large numbers of students, but also allowing for self-timed and self-paced education.
 - (b) Virtual laboratories and laboratory simulation in large introductory classes, again motivated by considerations of economy, efficiency, and also safety.
 - (c) The very widespread use of dynamic displays and visualization, to accomplish dynamic animations which cannot be done in any other way
 - (d) And finally, the very widespread, indeed universal use of the personal computer by students and faculty as a personal supercomputer and personal math wizard.
-

“ The real promise of computing in higher education: give people via simulation what you can't give them in reality.”

*Steve Jobs (founder of Apple Computer
and developer of the Macintosh)
EDUCOM Conference, November 1986*

3. VISUALIZATION AND DYNAMIC DISPLAYS OF PHYSICAL PHENOMENA

Let me now say a bit more about the topic of dynamic displays of physical phenomena—in brief “the ability to see things that are invisible,” based either on real-time calculation and simulation, or on pre-prepared “canned” animations which will increasingly become available on videodisk or cd-roms. I am sure that all of us in higher education will be making increasing use of animated illustrations, plots, and computer displays in the future, both to go along with textbooks for private study, and in the classroom for use with lectures (although there can be some surprises in this use of computer displays, as I will mention below).

One of our major needs in this area, however, is for improved and standardized classroom electronic display capabilities, so that we can really have omnipresent “computer blackboards” in the future. I hope that in this regard, HDTV may provide part of the answer. That is, I hope that the routine classroom of the future will still have generous chalkboards, as at present, but also a set of conveniently located, built-in television monitors which can be used for the convenient display of any sort of electronically generated information, including computer displays, VCRs, television systems, or any other electronic source. In particular, I view such monitors, along with a small

built-in color TV camera, as entirely replacing the projection screen and the overhead projector in the classroom of the future.

The only way to demonstrate such computer animation programs effectively is obviously with a computer; but for a printed manuscript a few illustrations taken from the computer screen will have to suffice. Shown in Figure 1 are snapshots from just a few such programs, taken more or less at random from my personal collection of animated display programs. In the upper left is an animated automobile engine demo, taken from a Buick floppy-disk sales pitch. Click the button and the engine begins running, with the cylinders going up and down, the valves opening and closing, the gas mixture being injected and exploding, and exhaust gases being pushed out. Press the space bar and the engine speeds up, accompanied by a realistic computer-generated engine roar which speeds up as the engine does.

Adjacent to this is a molecular drawing program, not a very good one actually, but with it you can still build, view, rotate, rescale and even animate complex molecules on the screen. Much better versions are now available.

One row down is a favorite for electrical engineers: a small, inexpensive, but powerful digital circuit analysis program. Using the mouse one can select any of the standard logic circuit elements shown in the palette on the left, drag it out onto the screen, and then wire up a digital circuit, complete with (rather old-fashioned) toggle switches and LED displays. (Note that the mouse pointer has the form of a soldering iron at this point; if you want to delete an element it turns into a pair of wire cutters.) Other windows permit you to draw input waveforms and observe the output waveforms at different points in the circuit; and save complex collections of elements away as "black boxes" to be built into more complex circuits. Shown next to this is a simple group velocity demonstration in which a pulse moves across the screen with user-adjustable phase and group velocities.

At the bottom left is one screen from a very useful transmission line simulator program written by Professor Charles Roth, Jr., of the University of Texas, which I have found exceptionally effective in teaching undergraduate electromagnetic theory courses. Finally, at the bottom right is one frame from an animation program by Professor Blas Cabrera of the Stanford Physics Department, showing the outgoing electric fields from a linearly oscillating electron. I find watching this program in operation extremely instructive.

4. PITFALLS AND SURPRISES IN COMPUTER-AIDED EDUCATION

One needs to note, however, that there can also be surprises in computer-aided education. There was a recent report, for example, of an unintended experiment on writing quality in a college English department in the United States. When the students in a large first-year writing course were first provided the opportunity to use personal computers and word-processing software instead of the conventional typewriters, instructors in the course began to note what seemed to be significant differences in writing quality of students using different computers. In fact, the clear-cut results of this one experiment seemed to be that students using Apple Macintosh computers wrote significantly poorer quality (though typographically much prettier!) compositions than very similar students using IBM-PC computers. This one experience (admittedly unplanned and as yet unrepeated) is not to be taken totally seriously, but it does illustrate some of the questions to be raised.

More serious is the experience reported in an interesting article, "The Computer as an Electronic Blackboard: Remodeling the Organic Chemistry Lecture" by Joseph Casanova and Sally L. Casanova, reported in the *EDUCOM Review* for Spring 1991. This deliberately planned and carefully performed experiment involved two parallel sections of a large undergraduate organic chemistry course with essentially identical groups of students given in the same academic semester. Both sections were taught by instructors with substantial previous experience in the course who collaborated carefully in their plans. One group was given conventional lectures; the other group was given lectures enriched with many computer graphics demonstrations of the type described above to illustrate molecular bonding and similar topics. Both sections were then given the same final examination.

Some of the observations made by the authors during this experiment were that "students in the computer graphics section took few notes but watched intently. . . Class participation was unusually high, of better quality and more stimulating. . . The students asserted they had good understanding of the subject. . ." But nonetheless, "On

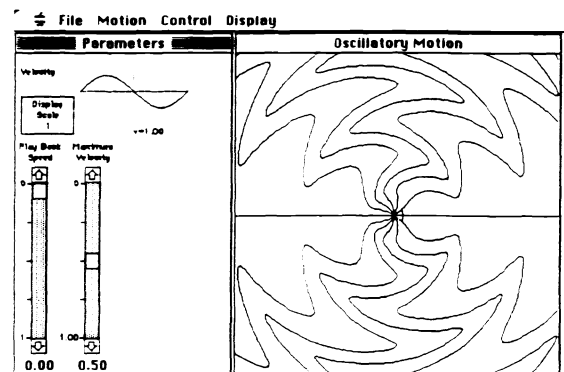
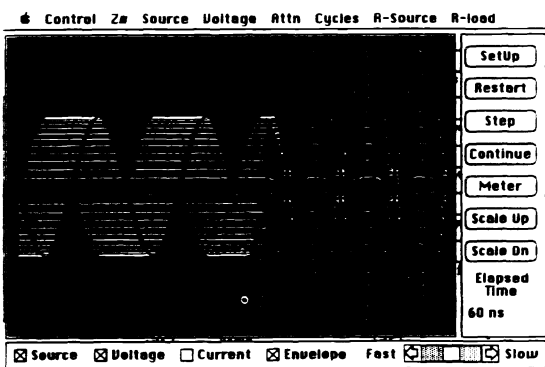
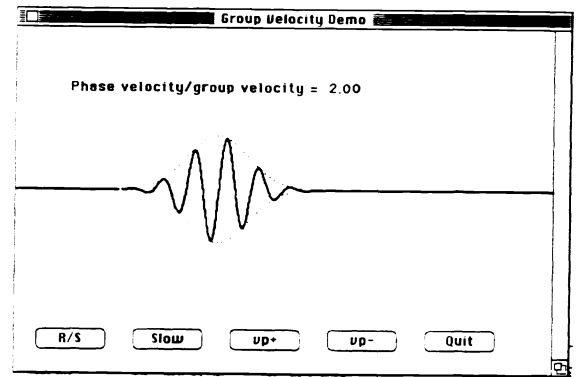
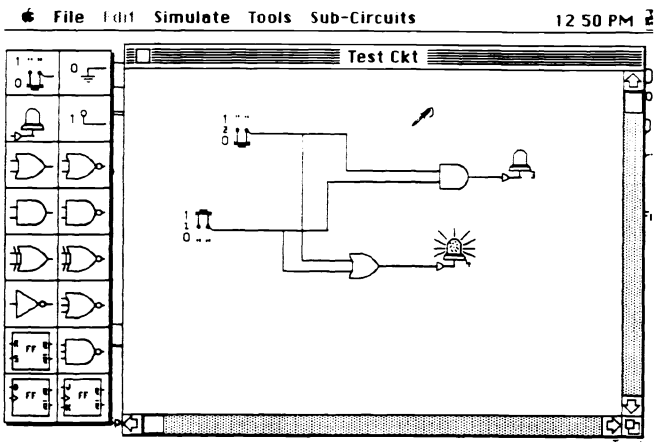
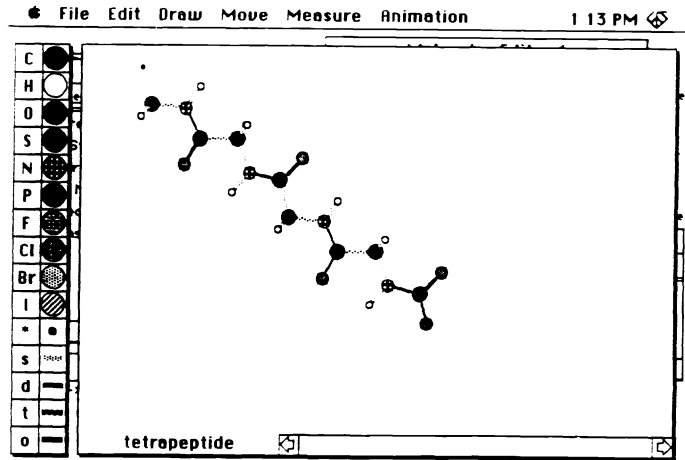
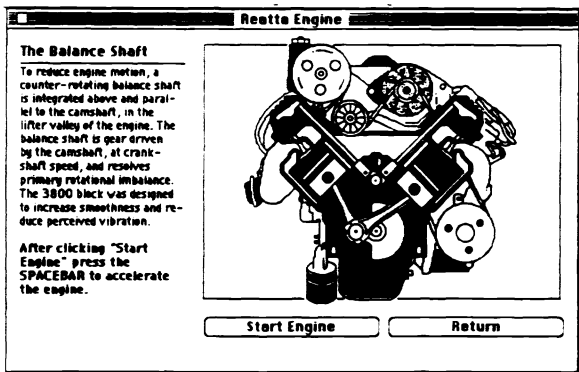


Figure 1. Examples of various computer display programs for use in engineering and science education, as described in the text.

the final examination the experimental (i.e., computer-aided) section performed worse in all categories of test items, even on topics judged to be visually intensive."

Other speakers in this meeting have made somewhat similar points: the best way to use computer graphics animations to illustrate a conceptual principle may be to present the physical concepts first in the "old-fashioned" manner, that is *without* computer assistance, forcing the student to do the intellectual work of visualizing the concept internally without graphics aids. Once the students have grasped the core concepts, however, computer visualization and animation can permit much more meaningful exploration of many different cases and situations.

5. SOURCES OF EDUCATIONAL SOFTWARE

If personal computer applications in education are indeed to become widely used, appropriate personal computer software for this purpose will have to be widely available. I can mention here some of the sources for the distribution of educational software that I am presently aware of. These include:

1. On-line computer archives, such as Info-mac for the Macintosh, or many other Internet or Bitnet-accessible network archives, along with commercial electronic information services such as CompuServe, Genie, and others.
2. Educational clearing houses, including commercial sources such as Intellimation and Educorp, along with a growing number of professional societies such as the American Chemical Society, and various nonprofit clearing houses.
3. Computer user groups, of which examples are the Boston Computer Society (BCS) and the Berkeley Macintosh Users Group (BMUG).
4. Commercial publishers, including educational software publishers, and the growing number of conventional textbook publishers who are putting out textbooks and workbooks with floppy disks included.
5. Finally there are a growing number of educational computer-oriented magazines and textbooks, such as the journal *Computers in Physics* published by the American Physical Society and the textbook *Computational Physics* by Steven Koonin.

We must keep in mind that there are thorny policy and economic issues still to be settled in the preparation and distribution of educational software, such as how do software authors get paid (or do they?) How do institutions and universities budget for the acquisition of educational software, a new item suddenly appearing in their budgets? And how do educational users and institutions deal with issues of privacy, security, and possibly even liability?

6. STUDENT-AUTHORED EDUCATIONAL SOFTWARE

One of the potential sources of educational software and of dynamic computer displays in which I am particularly interested is the category of *student-written educational software*. That is, I have now taught several times a project course for undergraduate and graduate students in which the primary task for each individual student or group of students is to write a useful demonstration program for the Macintosh computer, demonstrating some physical phenomenon or topic that is of particular interest to them or that they have encountered in one of their undergraduate or graduate courses. This is not a programming course; it is a project course, in which the objective is to produce some small piece of software, which will, hopefully, be useful as an educational demonstration for others.

To accomplish this goal, the students involved have to learn a modest amount of programming (Microsoft QuickBASIC is my recommended choice for novice programmers wanting to produce effective programs on the Macintosh with minimum startup effort), together with something about user interface considerations, in most cases a certain amount of numerical analysis, generally a certain amount of graphics, and finally, last but not least, a good deal about the physics and the behavior of the specific system being demonstrated. All of these are valuable skills for use later in their careers. Indeed, it is a truism that there is no better way to develop a deep understanding of

the physics of some particular topic than attempting to write a computer program that calculates and demonstrates that particular topic—not to mention the communications skills that need to be employed for doing this. Writing a small software project can also provide experience in carrying out a kind of design project, and it can provide opportunity for experiencing group work or a team effort, which is particularly important for engineering students.

Shown in Figure 2 are three examples of such projects, representing neither the best nor the worst examples from these student projects. All of these programs, incidentally, were written using Microsoft QuickBASIC. Shown in the top line is a simple implementation of the resonator stability diagram for optical resonators; you can click the mouse over any point in the stability diagram and the program will design and display the appropriate stable or unstable resonator (for unstable resonators it shows the geometrical mode outline). The center pair shows another on-screen circuit analysis program, this time for analog circuits. Again you can click on the circuit elements or signal generators in the palette at the left, drag them into position on the circuit board, and calculate both time and frequency responses for the resulting circuit. The students who did this project didn't have time to figure out a wiring algorithm for arbitrarily located elements, so a circuit pegboard approach was used instead. This was a one-quarter project for a team of three undergraduate students.

The bottom project represents one of my favorites: a dynamic user-controlled animation of integrated circuit fabrication. The program begins with a blank silicon substrate on the screen, along with icons for the IC fabrication steps shown at the right. Select "Oxide" and optionally adjust some parameters using the menus shown, and you will see an oxide layer dynamically grow on the surface of the silicon. Add nitride and photoresist layers, then select "Light" and use the mouse to define the portions to be illuminated, as illustrated in the left-hand illustration. Continue with etching, ion implantation, and metallization steps (you see all of these happen dynamically), and you can end up with something like the MOS device shown in the right-hand illustration.

7. PERSONAL COMPUTERS AS PERSONAL MATH WIZARDS

Finally, I would like to add a few observations on the topic of computers as personal super-calculators or personal math wizards. Professor A. Consortini raised in an earlier session of this meeting the interesting question, "How much mathematics should an optics student know?" The availability of personal computers as personal math wizards adds a new dimension to this question.

The point is that personal computer programs are now available which can not only do purely numerical calculations ("number crunching"), but which also know how to do numerical integration, differential equation solving, advanced special functions, and matrix operations with a minimum of user programming. Beyond this, these programs can also give immediate and accurate graphic displays of the results, immediately, on the screen, including immediate presentation-quality 2-D or 3-D plots.

But in addition, these programs can also do sophisticated symbolic manipulation, including algebra, calculus, matrix operations, transforms, set theory, number theory, and many more sophisticated mathematical operations. I won't attempt to present either a complete list or any comparative evaluation of these programs, but some of the programs that I know of just for the Macintosh include Eureka, Maxima, Maple, MathCAD, Mathematica, Matlab, Theorist, and TK-Solver. (The frequent mention of Mathematica in the following paragraphs will merely indicate that it is the particular program I happen to use the most at present.)

We can all think, for example, of how our (real) desktops used to look when we set out to carry through some sort of mathematical analysis in "the old way." On my desktop you would likely have found a pad or two of paper with soft pencils and erasers, a collection of earlier notes on the subject under discussion, one or more mathematical physics textbooks such as Morse and Feshbach or Courant and Hilbert, a handbook of functions such as Abramowitz and Stegun, perhaps a table of integrals, a table of transforms, and so forth.

When I undertake a mathematical analysis in the new style today, however, nearly all of the above are apt to be gone. Instead I will have just my Macintosh computer with all its files of previous work, a copy of Mathematica or some similar piece of analytical software, the manual for that program, a pad on which to make a few notes, and a laser printer available in the next room.

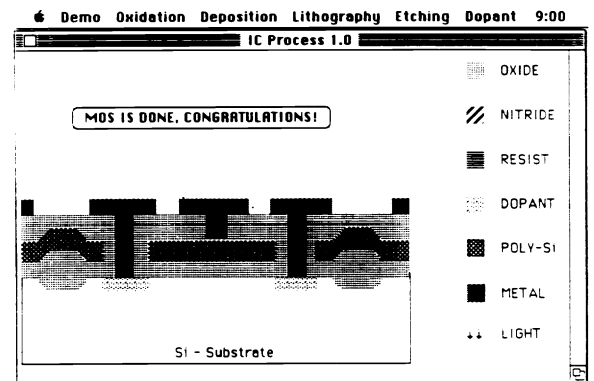
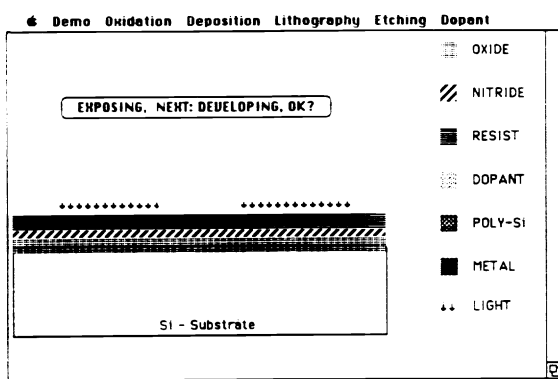
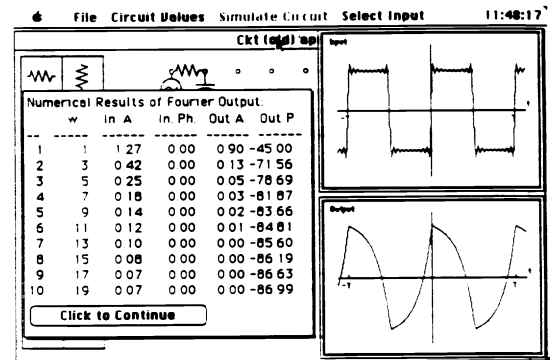
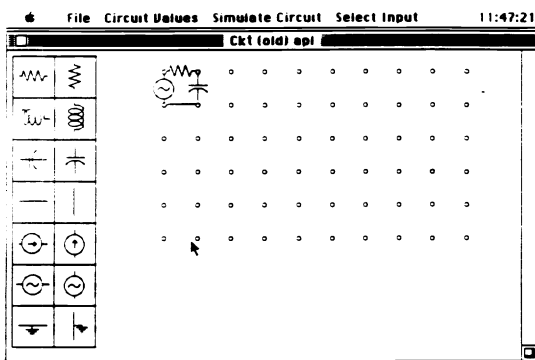
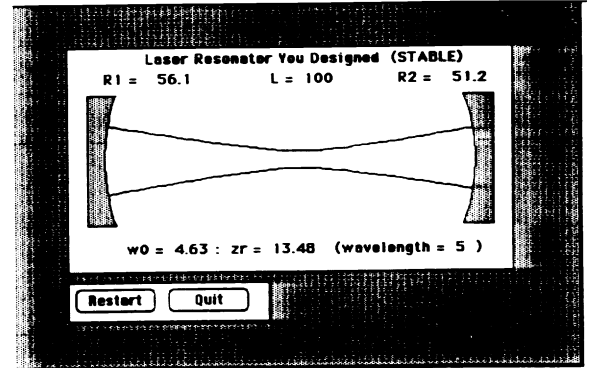
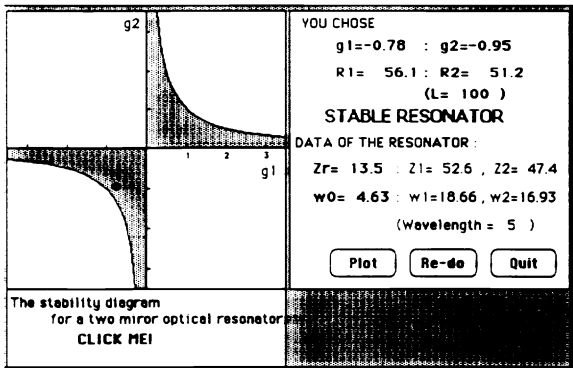


Figure 2. Examples of student-written educational demonstration programs. All of the programs shown here were written using Microsoft QuickBASIC on the Macintosh.

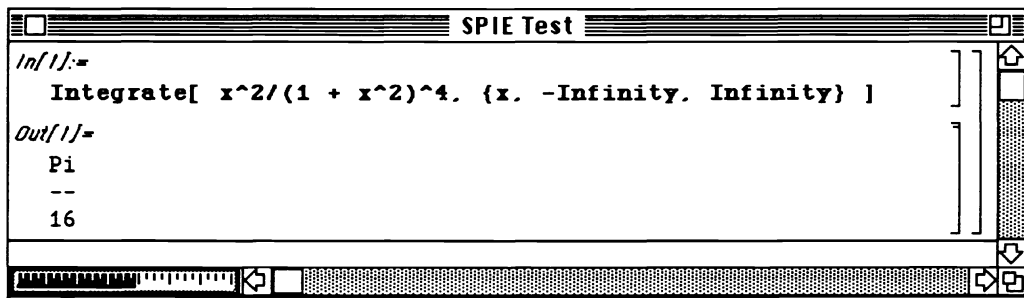
As just one elementary example of a mathematical analysis done in both the old way and the new way, I was interested recently in calculating the second moment of the far-field beam pattern from a diode laser junction. This field distribution has approximately the form of a lorentzian-squared function, namely

$$u(x) = \frac{1}{(1+x^2)^2} \quad (1)$$

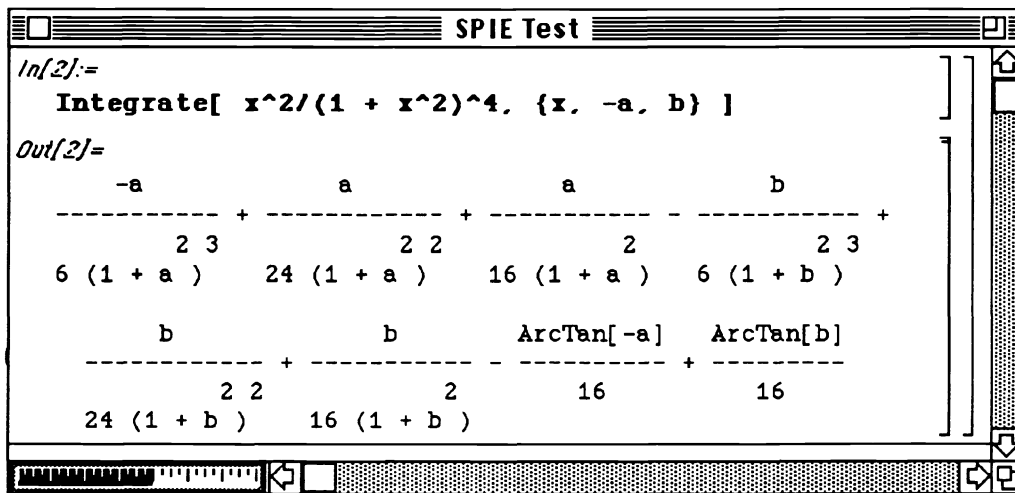
and I needed to calculate the second moment of this squared-lorentzian amplitude function, or

$$\overline{x^2} = \int_{-\infty}^{\infty} \frac{x^2}{(1+x^2)^4} dx . \quad (2)$$

Now in the classical mathematical approach to this bit of analysis, I would view this as a contour integral which can be closed with a semicircle at infinity, and I would then consider the second-order poles of the denominator at $x = \pm j$ and attempt to evaluate the integral as $2\pi i$ times the sum over the residues at these poles. I'm still able to carry out these kinds of calculations, even several decades after my last course in complex variable theory, but when I began to do the actual algebra in this particular case it rapidly became considerably more tedious than I was happy with. I was therefore led to start up Mathematica on my desktop computer instead, and after doing so I found I could evaluate this second moment by simply typing into the program the single line



from which I immediately got back the on-screen result shown. Just to see how smart Mathematica really was, in fact, I typed in the same integral with indefinite limits, and again immediately got back the result:



This is only one example. But, as regular users of such programs will know, programs such as Mathematica or several of the others listed can not only do integrations like this, but can perform essentially any mathematical manipulation that involves any of the familiar and less familiar special functions, including Hermite, Laguerre, Gegenbauer and Jacobi polynomials, Bessel, Legendre and generalized hypergeometric functions, and so on. In essence any mathematical knowledge that exists in Abramovitz and Stegun, and probably several other mathematics textbooks as well, is also present and available for immediate access in Mathematica.

Given these programs and their capabilities, we are faced with the difficult question of just how much traditional mathematical knowledge a student should possess, as well as how the student should acquire this knowledge, in order on the one hand to have something like the traditional depth of understanding of mathematical analysis that was desired in "the old days," combined with the ability to take advantage of these marvellous and powerful new tools, without being conned by them. Learning how to best make use of this computing power, and in particular how to develop and maintain fundamental knowledge and understanding in students while still taking advantage of these new tools, will be a challenge to all of us in education.

8. ACKNOWLEDGMENTS

I appreciate very much the continued support of the Air Force Office of Scientific Research for my research programs in lasers and optics, and the support of Stanford University and Apple Computer, and the students in my classes, for my educational efforts, especially in computer display techniques. I also appreciate the partial support of the SPIE for my attendance at the meeting where this paper was presented.