

MacLaboratory for psychology: General experimental psychology with Apple's Macintosh

DOUGLAS L. CHUTE

Drexel University, Philadelphia, Pennsylvania

Every student at Drexel University is required, on admission, to purchase a Macintosh computer. Consequently, there is an understandable demand to effectively utilize this resource in the undergraduate curriculum. We have developed what amounts to a series of "take-home" programs that convert the Macintosh into a number of "pieces" of experimental psychology equipment. Providing each student with a personal psychology "MacLaboratory" has apparent pedagogical and practical benefits, from creative hands-on experience to ease of independent research. This paper summarizes details of the program to date, its development process, supporting materials, and our experience when every student has a personal computer.

This paper represents a case study and discussion of our integration of the Macintosh computer into the psychology curriculum. At Drexel University, all incoming students are required to purchase a Macintosh computer, which comes with the usual MacWrite/MacPaint software as well as a Multiplan spreadsheet and two programming languages.

In part because of the unique nature of psychology as a discipline, I felt that we could make a novel contribution to the use of the microcomputer at Drexel. Specifically, software could be provided that converted the Macintosh into real and functional research equipment permitting the student to have, in effect, a take-home MacLaboratory for psychology. Our emphasis, then, is not on simulations, electronic page turning, or programmed learning, but rather on direct hands-on interactive and creative generation of research data. To date the effort has been geared to implementation in the introductory psychology course. At Drexel, this course enrolls 1,300 students a year and is structured as two hours of lecture and one hour of small (15 students) tutorial/laboratory sections. The actual Macintosh applications are designed to provide general research equipment and are not specifically geared to the introductory student. The intention is that the various modules produced should be sufficiently versatile that they may be used throughout the undergraduate and graduate curricula.

As the case study for this paper, I will use the introductory psychology course to highlight some of the design and development considerations illustrated with Macintosh software and with the introductory laboratory manual

(Chute & Daniel, 1985). Although institutional and student-demand factors contributed to the movement toward computerization, the major impetus was in fact achievement of a pedagogical goal. The content of the introductory course, contained in the laboratory manual, was devised and pioneered by Robert S. Daniel, a distinguished teacher of psychology. His work has shown that in postgraduation follow-up studies, students have consistently rated this course as "one of the most valuable courses" as well as "one of the most enjoyable." Not incidentally, the graduate-student instructors of the laboratory sections of the course gain significant insights, supporting the adage "You never learn something until you teach it."

When I first arrived at Drexel, the department was small, and equipment and laboratory space were poor. The Macintosh programs permitted the leapfrogging of intermediate stages in laboratory and curriculum development. No doubt similar benefits could occur at other small and medium-sized institutions. A secondary benefit will be the production of research equipment and computer experience not only for undergraduate projects, but also for work in the graduate neuropsychology programs.

At this preliminary stage, we are unable to assess how effectively the Macintosh can deliver the major features of Daniel's prototype. Clearly, some factors are substantially altered by computerization, for example, changes to normative data bases. Classroom interactions may also be quite different due to purported social factors, such as computer phobia or the isolation and solitude of working on the machine. Over the last three terms, however, there has been little evidence from the classroom or informal feedback from students suggesting that these are, in fact, problems.

For the case study, I will work through the various software projects in the laboratory manual, which will permit me to comment on some of the observations we have

The software development is supported in part by a grant from the Pew Memorial Trust. Software design and coding have been facilitated by John Harvey, David A. Barroso, Derek White, and Kerwin W. Thomas. Address correspondence to D. L. Chute, Director of Neuropsychology, Drexel University, Philadelphia, PA 19104.

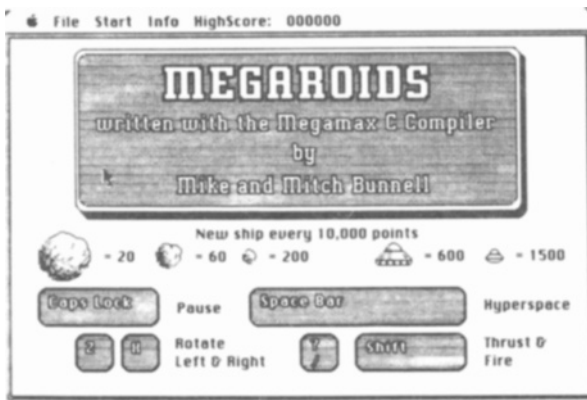


Figure 1. Public domain software used to conduct a simple experiment concerning the effects of handedness.

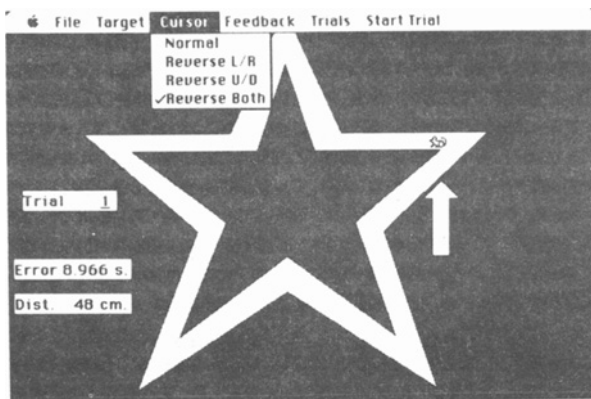


Figure 2. Data from the mirror tracing task are used to illustrate various aspects of statistical and experimental methods.

made that are primarily relevant to the issues of Macintosh computerization as opposed to psychological content. The first project, shown in Figure 1, uses an arcade-type game, Megaroids, which was borrowed from public-domain software.

The use of public-domain work was quite intentional. First, the project illustrates that principles of solid research can be applied to any topic or area. In this case, students investigate the effect (usually, counterintuitively, none) of playing the game with the preferred or nonpreferred hand. Second, it illustrates the point that the Macintosh should be viewed as a versatile tool, preparing the students to use the variety of public and proprietary software at their disposal in their research projects. Third, the fun aspects of the project serve as an ice-breaker in the tutorial class and establish social interaction and support. In introductory psychology laboratories, the take-home use of the software is slowly built up. By starting with laboratory experience, students acquire important content information as well as familiarity with the standardized operating procedures and unique Macintosh characteristics.

We perceive several advantages of the Macintosh over other microcomputers. For example, the Finder provides considerable uniformity and ease of operation. In addition,

our programs have been designed to be documentation-free, as students are notably lax about extra reading and we prefer that they concentrate on the lecture, text, and laboratory manual. I have observed that my colleagues with "easier-to-program" machines are still faced with the problem of whether they are psychologists first or computer programmers. The Macintosh development system is so complex that it obviates the need to be both. Accordingly, a surprising strength of the Macintosh is that it requires a sophisticated professional development team of programmers like Drexel's Software Development Group (Perkey, 1986). The psychologist then functions as architect, as opposed to builder, and can keep academic and pedagogical concerns in the forefront (Hewett, 1986).

The next three projects in the laboratory manual use the perceptual and motor skills application developed in-house. One of the software tools that can be created is a traditional mirror tracing task (see Figure 2). In the introductory psychology course, students use the program to study motor learning, progressive error, and counter-balanced designs, and to determine a standardized score for normative interpretation as a fundamental concept in tests and measurement. One problem that arises is whether normative data from the literature are in fact appropriate for the computerized version. One benefit is that apparatus generated by the Macintosh is often more versatile than the original. In the case of Macintosh mirror tracing, for example, it is possible to reverse left and right, as in normal mirror tracing, but it is also possible to reverse up and down, or both together. By selecting different parameters from the motor skills application (Figure 3), the user can conduct a pursuit rotor experiment. In introductory psychology, it is used to investigate "massed versus distributed" learning.

It is hoped that wider use of the programs will result from the production of general software tools not linked to a specific course or level in the curriculum. We hope that the emergent property of increased versatility will pro-

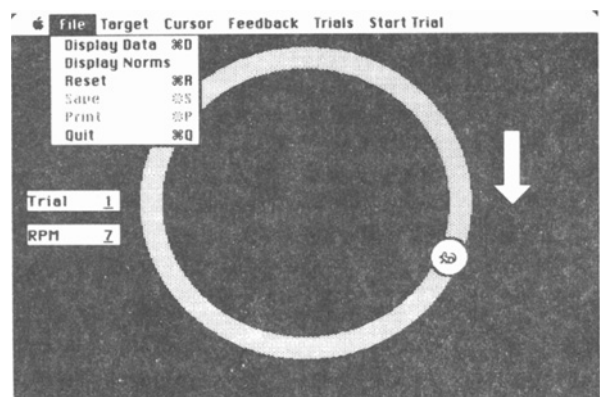


Figure 3. The pursuit rotor is a generally useful tool that is used by introductory students to investigate the effect of "massed versus distributed" practice.

vide unique new apparatus which will permit unique new research projects, so that students can have the pleasure of doing something quite original, as opposed to simply replicating well-established past work. Beyond the introductory course, the take-home aspect of the software, coupled with the fact that every student has a computer and supporting programs, permits hands-on experience in courses that would otherwise lack such a curriculum component. The addition of practical experience to these courses is facilitated by the lack of necessity for alteration in credit hours, physical space, equipment, scheduling, or staffing. Producing general software tools means that psychologists who teach small courses can justify the expense of creating a program in their area of expertise by the fact that it has general departmental utility.

The fifth project in the introductory psychology course is somewhat anomalous in that it doesn't produce a tool. Rather, it provides a computer representation of a brain (Figure 4) in which students can investigate the functional role of different cortical and subcortical anatomical structures in behavior. From a computerization point of view, this software illustrates the incredible time savings that can be gained by having coordinated university-wide development by a software development group. This particular application was written in a day and was up and running in a week. From the academic point of view, this was possible because, as is the case with many successful applications, the psychologist author already had lengthy teaching experience. From the programming point of view, the centralized resources of Drexel's Software Development Group enabled it to take a program already successfully used in the history department and, more or less, simply swap maps of Europe for maps of the brain.

Project 6 in the MacLaboratory manual uses an application entitled "Specialization of Hemispheric Function in the Central Nervous System" (Figure 5). Basically, students investigate a tachistoscopic-like presentation of either verbal or symbolic material to either the left or right cerebral hemispheres. This program is versatile because of the ease of file editing and the high-resolution graphics of the Macin-

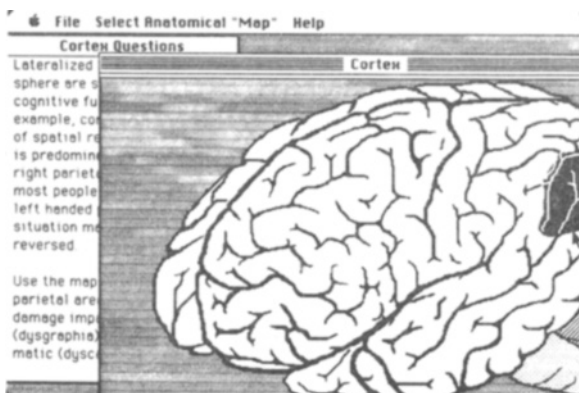


Figure 4. By locating various cortical and subcortical structures, students learn some elements of neuropsychology.

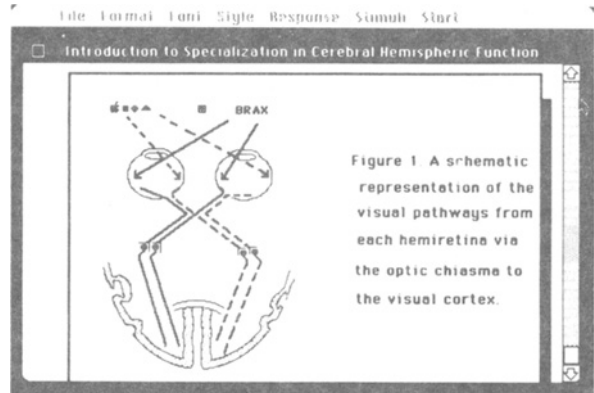


Figure 5. Using the "blind spots" to position the head, students are able to conduct their own experiments on hemispheric laterality.

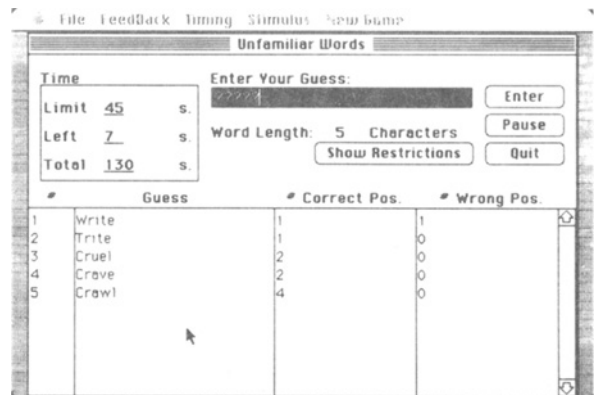


Figure 6. The problem-solving task. As with most MacLaboratory software, the user has editorial control over the creation of stimuli so that a number of unique experiments are possible.

tosh. Accordingly, a number of experiments in laterality are possible for appropriate places in the curriculum.

Project 7 refers to group effects on problem solving. One of the difficulties in any model of computerization is that many areas of psychology—for example, clinical, social, and physiological—are awkward to effectively represent. This, no doubt, is due partly to the content and partly to the differential interest in computerization by various subdisciplines in psychology. For the introductory course, MacLaboratory uses a "word discovery" problem-solving task (Figure 6) to test two colloquial hypotheses concerning group effects, namely, "two heads are better than one" versus "too many cooks spoil the broth." The usual result is an interaction with the former superior for hard tasks (unfamiliar words) and the latter superior for easy tasks (familiar words).

Project 8 constitutes a simulation of psychophysiological polygraph recording; however, it is interactive in that students condition a Galvanic Silicone Response (GSR) and investigate the polygraph's limitations as a device for the detection of deception.

Project 9 involves correlation of reaction time with a motor skill. The application disk, however, is quite ver-

satellite, providing the possibility for psychophysical measurements and standard reaction time paradigms like the Sternberg task. As illustrated in Figure 7, the stimuli and their location on the screen are readily edited and can be strung together in various test files.

For fine time measurement, the return of the raster scan initiates the clock, thus eliminating the variability that otherwise could occur due to the screen refresh rate.

Project 10, operant conditioning of the rat, is not computerized. The point, I suppose, is that not everything you might wish to accomplish pedagogically needs to be on a computer. In fact, students are much more sophisticated now than they were even a few short years ago. No longer is simple access to the machine or flashiness of the software sufficient to impress them. Preliminary feedback in introductory psychology suggests the overall program design provides the academic "meatiness" that now seems to be required. The hands-on creative and interactive effect that is real, as opposed to simulated—that makes apparatus out of the machine, as opposed to electronic page turning—is most welcome. We try to avoid needless computer busywork simply for the sake of using the computer.

Although the current MacLaboratory for psychology includes only these projects, a revised version is planned for the fall. Included will be new modules that have been developed and integrated into the program at the suggestion of other faculty members. For example, Figure 8 shows the "desk top" of an auditory perception module. Although details have yet to be finalized, I hope that Drexel will be in a position to provide contracted software development for additional MacLaboratory modules to expand the breadth and depth of curricular materials available to psychology as a discipline.

The final project in the introductory psychology package is an independent individual research project designed and conducted by the students. At the introductory level, this would normally be considered an overly ambitious

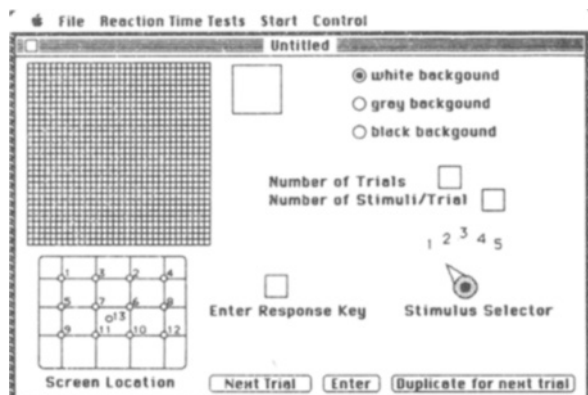


Figure 7. Editor screen to create different types of stimuli for the reaction time tasks. Other menus permit the user to select various stimulus and trial parameters.



Figure 8. A new auditory perception module facilitated in development by faculty interaction and the collective experience of Drexel's Software Development Group.

exercise, especially for the 350 students enrolled each quarter. Students report higher satisfaction and greater learning from this exercise than from any of the others. Of course, they are not required to use the computer, and a number, in fact, don't. Others, however, employ the versatility of the programs with which they have become familiar to create new and original research protocols. There are attendant risks, and, of course, ethical concerns. The small class size of 15 permits quite effective direct control, and a great deal of time is spent on ethical issues and the limitations we place on the project.

There are some tremendous advantages to every student's having a computer. Aside from those already suggested, there are other pleasant effects. For example, the availability of word processing with MacWrite produces a respectable finished product; in addition, the students actually engage in self-editing, with a consequent improvement in academic quality. Graphics are similarly improved with MacPaint or other available applications.

I have presented these observations as a case study in part because insufficient development has occurred as yet for formal evaluation. However, a sophisticated evaluation process is part of Drexel's overall approach to microcomputerization (McCord, 1986). For the MacLaboratory program, we are just beginning a Macintosh-based automated data-collection system to provide immediate evaluative and prescriptive feedback to graduate instructors and lecturers in the course. MacLaboratory for psychology has evolved because of the unique, and therefore somewhat idiosyncratic, situation as a result of being a part of Drexel University's interesting experiment in universal microcomputerization. For example, what we've accomplished works not just because every student has a computer, but because they all have the same computer. Drexel's institutional commitment to the Software Development Group from supportive and facilitative administrators has been essential.

I would also especially acknowledge the contribution of the Pew Memorial Trust.

But the real evaluative criterion is not whether a computer can be used in the introductory psychology course, but whether this form of the course can achieve the highest standards, as has its predecessor. I hope in the future to show that this version of introductory psychology can also make most students effective consumers of research, and perhaps make a few effective generators of research, by allowing them to learn from doing research.

REFERENCES

- CHUTE, D. L., & DANIEL, R. S. (1985). *MacLaboratory for psychology*. Dubuque, IA: Kendall Hunt.
- HEWETT, T. T. (1986). When every student has a computer: A new perspective on courseware and its development. *Behavior Research Methods, Instruments, & Computers*, **18**, 188-195.
- MCCORD, J. (1986). Computing: Is it a better mousetrap? *Behavior Research Methods, Instruments, & Computers*, **18**, 210-213.
- PERKEY, M. N. (1986). The effect of a machine-rich environment on courseware development: The process and the product. *Behavior Research Methods, Instruments, & Computers*, **18**, 196-204.