Distance Learning: Can We Use It to Teach M Programming?

by Richard F. Walters and Nancy E. Reed

Abstract

This report describes two sections of an introductory computing class in which M was used as the programming language. One course was offered using conventionally scheduled lectures and laboratories and the second in an independent (Autotutorial) mode. Comparisons of the students' selection of each type of class and their performance in the course are presented. Results show that the independent study students performed as well as those in the lecture course. The report recommends that MTA consider offering Internet-based courses using a model similar to the one described in this study.

Introduction

One of the problems faced by the M community is availability of M programmers. While scheduled courses offered by several competent vendors help to meet this need, a great many people would benefit if other alternatives were available. Distance Learning offers such an option. However, there are few studies that have analyzed whether Distance Learning is as effective as scheduled instruction. This study describes one effort to ascertain whether students can learn as easily in a Distance Learning mode as by taking a conventional course.

Independent study courses offer a potentially attractive alternative to scheduled lecture courses for many students. Distance Learning is made possible by such courses in ways not matched by remotely televised instruction. Furthermore, independent study courses can reach audiences that would not otherwise be able to take such courses, owing to scheduling, location, financial or other factors.

In an earlier paper (Walters, 1994), an introductory course in computing was described which included an introduction to the M programming language. This class has been taken by several hundred students each year (some sections use a different programming language, but over 100 students a year learn M in this course) and has become a highly popular class. During the past two years, the course was adapted for presentation as an Independent Study class. In late spring, 1995, ECS 15 AT (Autotutorial) was approved by the campus. The first offering of the course was in Fall Quarter, 1995. At the same time, a section of the regular ECS 15 course was also offered. Both were taught by the same instructor, the senior author of this report.

This study presents the results of a comparison of the two versions of ECS 15 offered in Fall Quarter, 1995. It describes the course, its resources, the delivery of both lecture and independent study sections, and presents initial conclusions regarding the outcome of each section. The paper concludes with recommendations to the M community for possibly offering comparable Distance Learning courses via the Internet.

ECS 15: An Introductory Course in Computers

ECS 15, Introduction to Computers, is a general education course that has been offered to non-computer science majors on the University of California, Davis campus since 1991 (Walters, 1994). It includes, among other things, a series of laboratories on programming in M. The course has been taught using conventional lectures and scheduled laboratories. For the past two years we have been preparing to offer the course in independent study mode by videotaping lectures, preparing a laboratory manual and other materials for use in non-scheduled mode, and designing a new mode of student-instructor interaction which we have named the "Remote TA" project.

The course includes 9 laboratory exercises (four of them in M programming), a midterm, a final, and a term paper written on the use of computers in some

field of interest to the student. Further details of the course may be found in the earlier reference (Walters, 1994).

A Comparison of Scheduled vs. Independent Study Offerings of ECS 15

In the Fall Quarter, 1995, we were allowed to offer two sections of ECS 15. The first section was open to 100 students and used the same approach that had been in use for several years: lectures, scheduled laboratories, and conventional deadlines for all assignments. The second section was open to a maximum of 25 students who were required to take the course independently, using available resources including access to the instructor and TA during office hours and the learning materials described earlier. In this report, we will refer to this group of students as "AT students," using the "Autotutorial" designation assigned the course.

Both groups had the same deadlines. Since the same instructor and teaching assistants were available for all students, and since the same materials were used by both groups, possibilities of different exposure to instructional resources were minimized.

Approximately 110 students enrolled in the lecture mode class and 20 students in the AT section. Their reasons for taking the course in AT mode included an urgent need to fulfill graduation requirements that quarter, scheduling conflicts, living some distance from campus, and a desire to learn independently.

AT enrollment was restricted to students at the upper division (junior-senior) level. About 40 of the students in the lecture mode were also at this level and served as the control group for the study.

Recognizing that this paired offering represented an excellent opportunity to study the effects of Distance Learning, we requested and received support from the Office of the President of the University of California and from the Vice Provost for Undergraduate Affairs on the Davis campus to support a careful analysis of the relative merits of the two courses. The funding enabled us to hire the co-author of this report and to augment significantly the numbers of teaching assistants and readers assigned to the class.

We decided that one of the most important elements to incorporate into our study was a pre-test to give us baseline information on the students taking both versions of the course. We designed the pre-test to include several types of questions: Term identification was an important component. In addition, we included questions on programming concepts. There were also some general questions about computer components and the relative costs of hardware and software. To augment our understanding of student backgrounds, we also asked them to provide information about their general background, including educational level of parents, high school and college GPA (grade point average), and the degree to which they make use of computers for word processing or other applications (including email). GPA was checked by obtaining official information from the registrar on students enrolled in the class.

Learning Resources

Learning resources included the instructional staff, who were available in laboratories and during office hours; handouts including the laboratory manual, lecture notes, term paper guidelines, and a list of references used by former students; Netscape-based auxiliary material (including copies of most of the items already described); and feedback on the midterm.

One of the most important resources we expected to make available to students in these classes was the software package called Remote Technical Assistance, or RTA. RTA is a concept under development at the Davis campus that provides students with three resources. First, is an advanced form of messaging, significantly improved over email, which includes the ability to attach files (programs, term paper drafts, spreadsheets, etc.) and even screen snapshots, to help messages. The second mode permits live interaction with a member of the instructional staff, including shared screen annotation of images, and multimedia file transfer to enhance the interactive dialog. The third component is an "expert system" based on resources prepared for the course and augmented by responses to frequently asked questions as they occur in other modes of RTA use. Although we had expected to use this form of student-instructor interaction, the package was not sufficiently robust for use during the quarter, and only the expert system was used by students in these classes.

Scheduling Constraints

Strict rules were imposed and adhered to with respect to deadlines for submission of completed laboratory

exercises and term paper-related assignments. Students taking the AT course were allowed to complete all assignments early (including taking examinations when they felt prepared), but they were not permitted to defer any submission beyond the deadline set for the lecture mode section.

Student Performance Measures

Grading in both sections was done by the same individuals, so that there would be no bias in favor of either group. The control group consisted of juniors and seniors enrolled in the lecture mode course (one sophomore missing only a few units for junior status was included). Grading was done using absolute measures; no grades were adjusted based on student performance.

As a further incentive to in-depth learning, students were given opportunities to earn extra credit by completing advanced portions of the laboratory exercises and by turning in newspaper clippings related to computer uses in today's society. A maximum of 7 points of extra credit could be earned, making it theoretically possible to accumulate up to 107 course points. Final grading was on the basis of 100 points, with letter grades assigned roughly using 88, 78, and 70 as dividing lines between grades of A-, B-, C- and the related lesser grades respectively. Weighting of points assigned to different portions of the class were: laboratories, 40%; term paper, 25%; midterm, 15%; and final, 20%.

Preliminary Analysis of Student Performance

This report was written immediately after final grades were turned in and prior to an in-depth analysis of the relative performance of the two groups of students. The average final grade in the AT class was 88.88; upper division students in the lecture course had an average of 87.77; for all students in the lecture class the average was 84.16. These results suggest that students in the AT class did as well as or perhaps slightly better overall than their counterparts (upper division students) in the lecture mode class. We emphasize that the students took precisely the same examinations, had the same minimal progress deadlines, and in every respect were evaluated on an equal basis. The remaining analysis, therefore, presents our preliminary attempt to understand the relative nature of the two student groups in an effort to determine whether the equal-or-better performance of the AT students is due in part to a different background or to other selective measures that led in part to their selection of this form of instruction. At the outset of this study, we posed several questions that we hoped to answer by comparing the two modes of instruction. Some of the more important questions were the following:

- Can students learn effectively in an independent study mode?

- More specifically, can programming (e.g., in M) be taught in this manner? Are there subject areas that are better or less-well suited to the independent study mode?

- What cost factors can be assigned to the preparation and delivery of independent study classes? Is this form of instruction cost-effective?

- What reactions, positive and negative, did the students have taking a course in this manner? How did it meet or fail to meet their expectations?

- What might be done to improve the instructional materials or delivery of the course?

We are still analyzing the results of the study and cannot provide definitive answers to many of these questions. However, we believe that we have some partial answers, and we present them together with some recommendations for future approaches to helping people learn M programming.

In the next sections we present our provisional answers to some of these questions, modified by our own understanding of the problems encountered as the course evolved.

Can Students Learn in Independent Study Mode?

This question is perhaps self-evident: a great many people do in fact learn on their own. Indeed, continuing education today is almost by definition forced into the self-study mode in many cases. A more appropriate question might be: is the learning achieved by independent study students different in important ways from that of students in more conventional courses?

A rigorous analysis of this question requires an understanding of entry levels of knowledge in the subject area. We have yet to complete our study of this com-

ponent. However, preliminary analysis suggests that the students taking the AT course were no more experienced in computer usage than those in the lecture course. Hence, the outcomes appear at first to be comparable. If anything, the fact that AT students in the lower levels performed at a level higher than their average GPA suggests that they were motivated in one way or another to put forth more effort in this course.

It is safe to conclude that with proper motivation and adequate resources, an AT student can indeed learn at the same rate and level as students in conventional courses.

Can M (or other Programming Languages) be Taught in AT Mode?

Students in this course take four laboratories in M programming. The first two deal with basic concepts, a cook-book approach to learning the fundamental commands, string operators, and use of $ORDER to sort local and global arrays. The third laboratory introduces programming mode using a conventional editor and building an M routine (address/phone database builder). The final laboratory requires students to create a program for output of the address database completed in the previous laboratory exercise, with appropriate formatting, pagination, and related details. The final examination gives students a new program to analyze and requires responses to questions about the way the program accomplishes its stated purpose. There were also two questions closely matched to the pre-test, evaluating student understanding of concepts such as execution flow control, and the correct identification of operators, variables, commands, and constants in a command line.

The initial results of the final exam suggest that students did perform at adequate levels in analyzing a program written in M and that on average, basic concepts associated with programming were understood by the students in both the AT and conventional versions of the class. There was of course variation, reinforcing the fact that material presented was not equally absorbed by all recipients. It is also true that a great deal more office hours and one-to-one assistance was required to help students through this last part of the course objectives. Nevertheless, all of the students did perform as well on this part of the examination as on the other portions of the exam.

What Costs Are Associated with Development of an AT Course?

The resources generated for ECS 15 include a series of lecture notes, a laboratory manual, video tapes, and electronically stored resources. In addition, the instructor has compiled both a hard copy and an electronic bibliography, indexed by keywords, of references used by previous students taking the course.

All of these resources are used by the students in the lecture mode class. Whereas only selected videotapes are shown each quarter, they are also used in lectures as a better means of bringing exhibits to the classroom and of using animation and other techniques to make points that are not as easily done with chalk or overhead transparencies. Since the instructor (Richard Walters) makes a practice of preparing lecture notes and laboratory exercises for other courses not presently taught in independent study mode, it is difficult to assign a cost specific to AT instruction in this case (and most probably in the general setting).

The human resources needed to carry out an effective AT course cannot be accurately estimated at this time. When we started the class, we expected RTA to be operational and to require monitoring by TAs and/or instructors for significant periods of time. Since it was not available, however, these resources were used largely in office hour help on laboratories and term paper guidance. The instructor spent a significant amount of time answering email, but this was generated as much by the lecture mode students as by the AT students.

It is not clear, therefore, what costs can be specifically ascribed to the AT class. In an attempt to obtain more precise information, we plan to offer an AT class this spring with only the instructor and one reader responding to email and office hours.

What was the Student Response to the AT Course?

The overall reaction to the course content and experience was positive. However, students were mixed in their response to the question: Would you take another AT course? Eleven stated that they would take another such course; four stated that they would not; the remainder were unsure, most of them saying it depended on the specific course.

Problems cited in the course experience included the need for assistance during laboratories, the possibility of an additional text, and difficulty in coming into the campus videotape playback center to view the videotapes. (Alternatives in tape check-out or night broadcasting of the tapes might be helpful to solve this problem.)

Several felt that the additional attention of instructor/TAs made the course viable. One felt that RTA, had it been up, would have been a great help, but several liked the Expert Help System component.

What Instructional Resources Need Improvement?

The overall reaction to course material was positive. However, the final laboratory was felt to be too great a leap beyond the cookbook approach of the earlier exercises. We plan to modify this assignment to give more guidance in the next offering of this class. We also intend to augment the electronic resources as time permits. One long term goal is to adapt the video lectures to a series of shorter sequences that can also be stored on WWW resources.

Undoubtedly the greatest disappointment to the instructor, TAs, and students, was the fact that RTA did not get to operational status in time for it to be used in this class. The clear need for precisely this form of support was made in numerous cases, and it is clear that the interactive dialog mode would have received significant use had it been available in a form convenient to students.

One conclusion seems inescapable: resources for courses of this type will require constant updating and revision. This is not different from the needs of conventional courses, but it may require more planning.

Conclusions and Recommendations

These results strongly support the idea that independent study courses can succeed given the right motivation, instructional materials, and support. Since the course included presentation of programming in M, we can also conclude that programming can be learned in this mode.

Although students taking the AT course were in fact on campus during some portion of each week, the results could be extrapolated to Distance Learning, where student-instructor interaction would have to be via electronic means. Here, we believe that the RTA concept will be a major benefit, even if we could not fully utilize RTA in this study.

If we assume that courses in M might be offered using the Distance Learning approach, then use of the Internet would be an ideal mechanism for presentation of such courses. We strongly urge MTA to consider seriously negotiating with course providers to establish Internet-based courses in M and related technologies. Such an approach would greatly extend the visibility and availability of M and provide a source of income for MTA. There are no impediments with respect to software: single user student versions of M are available for Windows and UNIX platforms. The RTA software is public domain and encompasses Windows, UNIX and Macintosh platforms. Netscape and other viewing tools are also readily available on all platforms, as are packages that provide the Serial Link Internet Protocol (SLIP) necessary for RTA use. This study also highlights the need for additional investigations of the same type, to learn more about alternative forms of instruction, including Distance Learning. Much was learned from this study, but many questions remain unanswered. We hope that this study will provide a stimulus for future work of a similar nature. **M**

Acknowledgments

This study was supported by funding from the Office of the President, University of California, from the Vice Provost for Undergraduate Affairs, and the Teaching Resources Center of the University of California-Davis. The authors express their sincere appreciation for this support. We also express our thanks to Dr. Neil Willits, Director of the Statistical Laboratory on the Davis campus for advice and guidance in the design of this study. We are especially grateful to Andrew P. Austin, who served as the senior reader for the courses, giving generously of his time to assist in course evaluation beyond his regular duties.

References

Walters, R.F. "An Introductory Course on M Worth Exporting." *M Computing* 2, no. 1 (1994): 13-19.

Richard Walters, Ph.D., is a professor at the University of California-Davis. Walters@cs.ucdavis.edu

Nancy Reed, Ph.D., received her M.S. and Ph.D. in Computer Science from the University of Minnesota. She is currently teaching at the University of California-Davis. reedn@cs.ucdavis.edu