

# Chapter 10

## Developing Customized Instructional Software

*Michael Parker*

Department of Zoology and Physiology  
University of Wyoming  
Laramie, Wyoming 82071-3166  
(307) 766-4207

Michael Parker received his B.S. from the University of Michigan and his M.S. and Ph.D. from the University of Wisconsin (Madison). Teaching responsibilities at the University of Wyoming include courses at the freshman through graduate level. He first used mainframe machines for computer exercises in introductory classes during the mid 1970s. Since the early 1980s he has used commercially-available software to develop exercises for classes, and most recently he developed the authoring system described here. Research interests are in limnology, and he and his students have worked on a variety of projects located from Wyoming to Greenland.

**Reprinted from:** Parker, M. 1992. Developing customized instructional software. Pages 159-166, *in* Tested studies for laboratory teaching, Volume 13 (C. A. Goldman, Editor). Proceedings of the 13th Workshop/Conference of the Association for Biology Laboratory Education (ABLE), 191 pages.

- Copyright policy: <http://www.zoo.utoronto.ca/able/volumes/copyright.htm>

Although the laboratory exercises in ABLE proceedings volumes have been tested and due consideration has been given to safety, individuals performing these exercises must assume all responsibility for risk. The Association for Biology Laboratory Education (ABLE) disclaims any liability with regards to safety in connection with the use of the exercises in its proceedings volumes.

© 1992 Michael Parker

159

### Contents

Introduction.....160  
 Why Isn't Instructional Computing Widespread?.....160  
<sup>mp</sup>AUTHOR – One Solution .....160  
 Students as Authors .....161  
<sup>mp</sup>AUTHOR – Examples of Use .....161  
 University of Wyoming .....161  
 Other Examples.....162  
 Some Strengths and Weaknesses of <sup>mp</sup>AUTHOR.....164  
 Notes for the Instructor .....164  
 Materials .....165  
 Acknowledgements.....165  
 Literature Cited.....165

### Introduction

#### Why Isn't Instructional Computing Widespread?

Aside from the initial cost of hardware, instructional computing is not widespread for at least two reasons: the cost to faculty in time and the cost to institutions and students for purchase of software. For example, few teachers have adequate time for instructional development, and this problem is exacerbated in research-teaching departments with an inequitable reward system for teaching versus research (Moore, 1990; Trollip, 1988). Time is often required to learn programming or complicated software and time certainly is needed to develop instructional products (100–1000 hours to develop an application for 1 in-class hour commonly is cited; Glick, 1990). Even if instructional software were available, many faculty still would spend time restructuring courses because the software would not fit well with the organization of their courses. Therefore, minimizing time spent to produce a product usable by students is a requisite for widespread instructional computing.

Assuming an institution has microcomputers available for instruction, funds still are required to obtain legal software. For example, faculty need a legal copy of a spreadsheet to develop an instructional exercise, and students need a legal copy to use it. While networked machines provide one avenue to ensure legal use of software by many people on many machines, such software is often expensive, and networks are not available in many places. Student versions can be purchased for some software at a reduced cost, but most students do not consider this reduced cost a trivial one. Thus minimizing the cost of legal software to the institution and to students is important to foster widespread instructional computing.

#### <sup>mp</sup>AUTHOR – One Solution

A search for software which would allow me to create, or author, exercises for students in minimal time, which was inexpensive to purchase, which would cost little to distribute to many students (500–1000), and which could be used on virtually any DOS (IBM or compatible) microcomputer was unsuccessful. As a consequence I developed my own program, <sup>mp</sup>AUTHOR (MJP Software, 1621 Custer Ave., Laramie, WY 82070; demonstration disks are available).

This is an authoring system which allows anyone able to use a word processor which can produce ASCII files to create menu-driven exercises. When students use exercises they are presented with a screen of text (scrollable through 100 lines) which has a menu of one to eight

choices at the bottom; choosing an item from the menu causes a new screen of text and a new menu to be displayed, etc. The License/Use Agreement allows distributing free or selling unlimited numbers of exercises to students. Because all exercises are based on ASCII files created by authors (e.g., a faculty member), interested teachers easily can exchange exercises. Exchanged exercises can be used as is, or as templates to be modified for your courses.

### Students as Authors

Because most faculty can use word processors, most should require little time to learn to use <sup>mp</sup>AUTHOR. A much larger amount of time will be spent developing exercises. To minimize such time two strategies may be available to many teachers. First, students could earn credit within a class by creating an exercise (e.g., as a project or instead of writing a term paper). Second, some students could become student-developers; that is, students who develop and implement an exercise under the guidance of faculty and who receive, for example, 1 credit hour in the course Zoology 690M, Section 3 (Problems in Physiology). Thus, by spending time only to supervise and grade student activities, faculty gain instructional software customized for use in their courses. Students gain course credit and improved understanding by having to acquire and apply facts when creating exercises (McKeachie, 1990).

Students with whom I have discussed the idea were interested in the possibility of creating exercises. However, colleagues have been concerned about providing course credit for such activities. Clearly one credit in, for example, Problems in Ecology, would be inappropriate were the student-developer merely to act as a typist in creating an exercise. However, just as clearly, developing certain exercises could provide a student-developer with the opportunity to learn not only facts but to improve their understanding of processes, concepts, and principles. Some research even suggests that teaching or tutoring by students results in better learning than does being taught in the classroom (McKeachie, 1990). But, because students conceivably could learn little and might demonstrate no creativity while developing exercises, criteria must be established to determine when course credit is appropriate.

### <sup>mp</sup>AUTHOR – Examples of Use

#### University of Wyoming

Another faculty member and I share lecturing duties in a second-semester general biology course (with about 500 students, mostly non-majors). He covers botanical topics during the first half of the semester and I discuss zoological topics in the second half. Each of us examines students with multiple-choice hour exams.

For my first hour exam I used the <sup>mp</sup>AUTHOR system to create exercises which I then copied to floppy disks. About 15% of the students purchased disks on a voluntary basis. Because the authoring system requires only minimal memory (384K RAM to create exercises, 256K to use the exercises), a monochrome monitor, and no graphics board, students could use the disks on virtually any IBM or IBM-compatible machine. Studying was done in university computer rooms and at home.

Information on each disk was divided into two parts. First, students could choose to review factual information for which they were responsible and which was covered in the lectures and/or the text. However, this information was presented in a different fashion from that in either the lectures or text. Second, students could choose to view multiple-choice questions similar to those which would be asked on an hour exam. When using this part of the disk, a question was displayed on the screen just as it would appear on an exam. Unlike an exam, choosing any answer caused the display of the reasoning underlying why that was a correct or incorrect response (see also the next section of this chapter, Other Examples). Often, related information would also be made available for viewing in conjunction with the explanation. For many questions, if a student still was confused after viewing explanations for each choice, reference to the other part of the disk would provide more complete information. As well, reference to pages in the text and class notes from a particular day could help students find more detailed information when it was desired.

After taking my first exam students who used the disks for study answered a questionnaire, and those who were willing identified themselves. I compared the performance of the identified students on my exam to their performance on an exam given by my colleague. On average, students who identified themselves on the questionnaire significantly improved their score compared to their score on my colleague's exam by half a letter grade. While clearly not an analysis with well-designed controls, results do suggest that using the disks was helpful for many students. And, because the distribution of grades for students who used the disks was very similar to that of others in the class, the significant improvement in grades of disk-users did not occur because only better students used the disks. Reaction of students responding to a question about whether they perceived the disks to be useful was overwhelmingly favorable; almost all felt the disks were somewhat or very helpful in affecting their learning.

For several reasons the capability to use exercises outside the classroom was an important attribute when I developed the exercises. Most important, computers are not available in classrooms I use while they are readily available at a variety of non-classroom locations on campus. Second, outside the classroom students can work at their own pace which can be important because it may allow time for self-discovery (Duhrkopf, 1989). Third, I never have felt very successful at enticing poorly-performing students to use me as a source of help to improve their performance. This may occur because when these students come to me for help they, at least implicitly, must admit inadequacy or failure. However, using disks does not require such admission, and thus exercises on disks may be more likely to be used by students who need help most (e.g., Lavin, 1980).

## Other Examples

Successful approaches have been developed to teach cognitive skills other than memorization, and <sup>mp</sup>AUTHOR can be adapted for use with many of these approaches. For example, several authors have described practical approaches for improving thinking skills in introductory courses, in part by use of fill-in and multiple-choice questions in an unconventional manner (Crow, 1989; Maloney, 1982; Moll and Allen, 1982; Statkiewicz and Allen, 1983; Tyser and Cerbin, 1991). Maloney (1982) suggests that this approach is not as productive as having students answer essay questions. However, if the students are freshman then his suggestion may be invalid; it may be best to start with fill-in and multiple-choice questions to provide training not possible with essay questions alone. To understand why, read the following poem which was written about a century ago by Gerard Manley Hopkins.

*Spring and Fall:*  
*to a young child*  
 Margaret, are you grieving  
 Over Goldengrove unleaving?  
 Leaves, like the things of man, you  
 With your fresh thoughts care for, can you?  
 Ah! as the heart grows older  
 It will come to such sights colder  
 By and by, nor spare a sigh  
 Though worlds of wanwood leafmeal lie;  
 And yet you will weep and know why.  
 Now no matter, child, the name:  
 Sorrow's springs are the same.  
 Nor mouth had, no nor mind, expressed  
 What heart heard of, ghost guessed:  
 It is the blight man was born for,  
 It is Margaret you mourn for.

Now that you have read the poem, write an essay about the following question: What is the significance of the title of the poem?

If you have been trained like many scientists, then the skills you need to write this essay may be rusty at best. But what if instead of writing the essay, you were asked the following? Which of the following best explains the significance of the title, *Spring and Fall*? Explain the reason for your choice and your reason for rejecting the other choices. (a) It is fall, and Margaret wishes it were spring; (b) Because Margaret is a child, she doesn't like the signs of death fall brings; (c) Margaret is young, and the season of fall depresses her; (d) Margaret is in the spring of her life, and the season is fall, for the first time she recognizes her own mortality; and (e) Spring and fall are seasons, and people experience seasons in life, too. In this case the choices (a) through (e) limit the scope of possibilities to consider. You still must struggle with why each answer is correct or incorrect, but now your efforts should be sufficiently focused so that figuring out the answer seems feasible. The point is that such questions, carefully worded and with carefully chosen multiple-choice answers, (1) can provide needed focus and guidance to a bewildered freshman in introductory science courses, (2) require thinking and reasoning to answer, and (3) can be graded quickly.

Similarly, an important component of the program which won the 1988 EDUCOM/NCRIPTAL Higher Education Software Award for Best Curriculum Innovation, Unprepared Students, was multiple-choice test questions with explanations of why each response is right or wrong (Johnston and Kozma, 1988; see also Alperson and O'Neil, 1990). This introduced students to problem-solving strategies of professionals, and the performance of underprepared students improved markedly.

As another example, case studies have been used in law, business, and medicine to present realistically complex problems, to show what processes can be used to deal with this complexity, and to teach appropriate problem solving skills. Linn and Clancy (1990) discussed combining case studies with computerized templates to teach computer programming. In simplified terms, their templates represent ways that expert programmers organize knowledge and can be used as the basis for devising a strategy for solving specific problems. Providing students access to templates when faced with interpreting a complex case study enables them to see how experts have dealt with similar situations, and to learn how to identify appropriate, general strategies for solving different types of problems (note that, as discussed above, this is what one hopes to accomplish with

unconventional fill-in and multiple-choice questions). <sup>mp</sup>AUTHOR can easily be adapted to teaching strategies using case studies and templates.

Finally, exercises can be developed with <sup>mp</sup>AUTHOR which could be incorporated into almost all of the instruction presented in Volume 12 of *Tested Studies for Laboratory Teaching* (Proceedings of the 12th Workshop/Conference of the Association for Biology Laboratory Education). For example, one could develop an alternative to the TRIBBLES program to use in teaching reasoning (Boone, 1991). Or, "exercises" could be created as promotional material and for use in making presentations at seminars and workshops (Karpala et al., 1991).

### Some Strengths and Weaknesses of <sup>mp</sup>AUTHOR

1. Minimal computer skills are required by author and student; even students can create exercises (e.g., instead of a term paper or as a project).
2. Only inexpensive hardware is required so students can use exercises on virtually any IBM PC or compatible (e.g., classroom, computer room, and home).
3. Inexpensive software; legally distribute unlimited numbers of exercises to students for only the cost of floppy disks.
4. Generic; use in any discipline, create customized exercises for classes.
5. Does not display graphics (some simple graphs can be made using high ASCII characters if your word processor can access them).
6. Student cannot enter new data or information.

### Notes for the Instructor

1. Your most precious and limited resource likely is time; don't underestimate the time needed to create exercises. Do consider ways that students could develop exercises under your supervision, but also remember that some such exercises will need modification before use in your courses.
2. Students will likely resent having to spend time on activities not directly related to their course grade. Therefore, be certain both you and the students understand how performing the exercises may affect their grade.
3. Consider exchanging exercises with colleagues (this can be done easily because all exercises are based on ASCII files created by authors; use a modem and phone or mail disks to exchange ASCII files). Even if you don't use a colleague's exercises as is they can provide templates to be modified for your courses, and this could save considerable time.
4. If you plan to provide hundreds of disks to students, then consider having disk duplicating services make the copies. However, be certain to allow sufficient time. Such services often are advertised in computer magazines.
5. If you plan to make copies of disks for students yourself, then purchasing disks in bulk may cut costs by 50% or more. Local prices may approach those available via mail order (see advertisements in computer magazines), and local vendors may be willing to replace any bad disks. If you use 5.25-inch disks then purchase them with sleeves (the paper half-envelopes which protect disks), but don't pay extra for labels (it's easier to print your own labels from continuous feed labels with a dot matrix printer, or from sheets of labels with a laser printer).
6. I have found that most students are able to use disks and exercises on their own with three sources of information: (a) a demonstration by me of how an exercise works (e.g., in lecture); (b) written, step-by-step instructions for getting started printed on a slip of paper provided with

each disk (a 8.5-inch × 3.7-inch slip of paper, or 1/3 of an 8.5-inch × 11-inch sheet); and (c) information provided on the first screen displayed by each exercise.

### Materials

#### Instructor

1. IBM PC or compatible computers with 384K RAM (students need only 256K RAM), DOS version 2.0 (or more recent version), and monochrome or color monitor.
2. <sup>mp</sup>AUTHOR System (MJP Software, 1621 Custer Ave., Laramie, WY 82070; reduced prices for multiple copies; demonstration disk available).
3. Word-processing program which will save files in an ASCII format (sometimes terms like DOS text file are used instead of ASCII file; see the manual for your word processor).
4. Floppy disks for your machine; for storing exercises while they are being developed, and for final copies of exercises.
5. Labels for floppy disks.
6. Paper on which to print instructions about how to use the exercises.

#### Student

1. IBM PC or compatible computer with 256K RAM (authors need 384K RAM), DOS version 2.0 (or more recent version), and monochrome or color monitor.
2. Floppy disk(s) with exercise(s) (for use on any machine; e.g., home), or copy of exercise(s) on hard disk (for use only on the machine with the hard disk; e.g., classroom).
3. Instructions for use.

### Acknowledgements

Eileen Oswald (Valencia Community College, Orlando, Florida), presented the multiple-choice question about the poem by Hopkins at a Chautauqua Conference on Critical Thinking in Science Teaching, spring 1991.

### Literature Cited

- Alperson, I. R., and D. H. O'Neil. 1990. The boxscore: Tutorials 2, simulations 0. *Academic Computing*, 4(5):18–19, 47–49.
- Boone, G. C. 1991. Scientific inquiry. Pages 129–135, *in* Tested studies for laboratory teaching. Volume 12 (C. A. Goldman, Editor). Proceedings of the 12th Workshop/Conference of the Association for Biology Laboratory Education (ABLE), 218 pages.
- Crow, L. W. (Editor) 1989. Enhancing critical thinking in the sciences. Society for College Science Teaching, Washington, D.C., 117 pages.
- Duhrkopf, R. 1989. Simulating the learning cycle. *American Biology Teacher*, 51(4):246–248.
- Glick, M.D. 1990. Integrating computing into higher education. *EDUCOM Bulletin*, 25(2):35–38.
- Johnston, J., and R. B. Kozma. 1988. Software awards: The 1988 EDUCOM/NCRIPAL competition. *Academic Computing*, 3(4):30–35.
- Karpala, A. L., D. D. Burke, and R. Williams. 1991. Strengthening biology teaching by working with the local school system. Pages 175–187, *in* Tested studies for laboratory teaching.

- Volume 12 (C. A. Goldman, Editor). Proceedings of the 12th Workshop/Conference of the Association for Biology Laboratory Education (ABLE), 218 pages.
- Lavin, B. F. 1980. Can computer-assisted instruction make a difference? *Teaching Sociology*, 7(2):163–179.
- Linn, M. C., and M. J. Clancy. 1990. Designing instruction to take advantage of recent advances in understanding cognition. *Academic Computing*, 4(7):20–23, 35–41.
- Maloney, D. P. 1982. Fill-in problems. *Journal of College Science Teaching*, 12(2):104–106.
- McKeachie, W. J. 1990. Research on college teaching: The historical perspective. *Journal of Educational Psychology*, 82(2):189–200.
- Moll, M. B., and R. D. Allen. 1982. Developing critical thinking in biology. *Journal of College Science Teaching*, 12(2):95–98.
- Moore, R. 1990. What's wrong with science education and how do we fix it? *American Biology Teacher*, 52(6):330–337.
- Statkiewicz, W. R., and R. D. Allen. 1983. Practical exercises to develop critical thinking skills. *Journal of College Science Teaching*, 12(4):233–236.
- Trollip, S.R. 1988. Wrestling with instructional computing. *Academic Computing*, 2(4):26–29, 40–41.
- Tyser, R. W. and W. J. Cerbin. 1991. Critical thinking exercises for introductory biology courses. *BioScience*, 41(1):41–46.