Implementing Educational Software and Evaluating Its Academic Effectiveness: Part II

Karen Jolicoeur and Dale E. Berger

In Part I, published last month, we presented a successful classroom implementation plan for integrating educational software into elementary school classrooms. Implementing educational software successfully into classrooms is an essential prerequisite to achieving the second goal of the present research—evaluating how much students learned by using specific software programs. In Part II, we will measure how effective eight software programs were at teaching fifth grade students new fraction concepts and spelling words. In addition, we will examine the validity of teacher and student software ratings based on the effectiveness of each program.

Evaluating Educational Software

Surprisingly, there have been very few research studies of the educational effectiveness of individual software programs for microcomputers. Research has shown that computer-assisted instruction (CAI) is an effective medium for improving academic skills in significantly less time than conventional classroom methods (Kulik, 1985; Kulik, Bangert, and Williams, 1983; Kulik, Kulik, and Cohen, 1980; Niemiec and Walberg, 1985; Niemiec and Walberg, 1987; Thomas, 1979). However, these earlier studies examined mainframe computer systems that presented CAI in monochrome/text-only formats—formats that are quite different from the colorful animated presentations that are displayed on classroom microcomputers.

In addition, little empirical research is currently available on the specific factors that make educational software effective (Jolicoeur and Berger, 1986). A goal of the present study was to increase the amount of controlled research available regarding the academic effectiveness of specific educational software. If we can measure the academic effectiveness of individual educational software programs, then we can begin to discover the specific factors that contribute significantly to good and poor quality software. Once successful software factors are identified, educational software developers can focus on building these factors into all educational software, thus improving the quality of the software. In the current study, four fraction and four spelling software programs were examined for the instructional value that they had with fifth grade students. These findings and their implications are summarized below.

Karen Jolicoeur is an Educational Computing Consultant for Lehrer Associates, 4208 Dundee Drive, Los Angeles, California 90027. **Dale E. Berger** is Professor of Psychology at The Claremont Graduate School, Claremont, California.
This research was supported by Apple Computer, Inc., and by Hesa, Inc., in conjunction with Project "Apple Core" that is currently in progress in the Archdiocese of Los Angeles.

Software Effectiveness

The effectiveness of the software was examined by addressing the following questions:

- Did the software programs contribute to learning?
- Did individual programs differ in effectiveness?
- Were software tutorials and games equally effective?
- Was new knowledge retained?
- Did gender predict learning?

Pretest (*Test 1*) scores differed significantly across the eight participating schools. This was not a surprising finding since the eight participating schools were located in different SES areas throughout Southern California. To compensate for pretest differences, *Test 1* scores were treated as covariates and adjusted cell means are reported for all subsequent analyses measuring software effectiveness. This means that *Test 2* and *Test 3* scores for each student were adjusted appropriately, treating all students as if they had each performed equally well on the fraction and spelling pretests. This precaution served to minimize effects due to pretest differences in the final analyses of software effectiveness.

One of the goals of the present study was to determine if the fractions and spelling software were effective teaching media. In other words, did the students increase their knowledge of fractions and spelling words during the four-week computer sessions? And if so, did they learn more than students who received normal classroom instruction? The students at all eight schools were receiving fraction instruction as part of their normal classroom curriculum during the present project. The fraction software therefore supplemented conventional classroom instruction. The spelling software, on the other hand, was not supplemented by conventional classroom instruction.

The *Test 2* scores of students in Group II (spelling first, then fractions) served as controls for the *Test 2* scores of students using the fractions software first (Group I), and vice versa (see Part I, Figure 2 to review the design of the implementation process). This arrangement permitted effects due specifically to the fractions software and spelling software to be separated from effects due to classroom instruction since both groups were receiving the same classroom instruction for fractions and spelling. If the fractions software was effective, students in Group I should have scored significantly higher on the fractions portion of *Test 2* than students in Group II. Likewise, if the spelling software effective was effective, students in Group II should have scored significantly higher on the spelling portion of *Test 2* than students in Group I.

Fraction Effects. The adjusted *Test 2* means in Table 1 show that the students who used the fractions software (Group I, mean = 39.7) learned significantly more about fractions than students who did not use the fractions software (Group II, mean= 35.0), $F=7.3$, $p < .01$. Even though all students were receiving fraction instruction in their classroom, the students who supplemented classroom instruction with the fractions software during the first two-week computer period learned significantly more about fractions than students who used the spelling software.

Spelling Effects. Table 1 shows that students using the spelling software (Group II, adjusted *Test 2* mean=82.1) learned to spell significantly more words presented on the spelling software than students who did not use the spelling software (Group I, adjusted *Test 2* mean=71.3), $F= 72.6$, $p < .01$. Thus, the spelling software also contributed significantly to student learning.

Individual Software Effects

Table 2 shows that some of the fraction software were clearly better than others, $F=16.3$, $p < .01$. *SVE Fractions* was by far the most effective fraction software (adjusted *Test 3* mean= 50.4), followed by *Eduware Fractions* (adjusted *Test 3* mean=38.7) and *Galaxy Fractions* (adjusted *Test 3* mean=38.3). The least effective fraction program was *Fraction Action* (adjusted *Test 3* mean=31.8).

One of the main differences between *SVE Fractions* and *Eduware Fractions* was the ease of use. In contrast to *SVE Fractions*, many students rated *Eduware Fractions* as confusing, frustrating, or difficult to use. This type of information is important for software developers and teachers to know when producing or selecting educational software. It emphasizes that even when the content of the materials presented in the software appears good, students will probably not learn as much if they find the software difficult to operate. They are likely to waste time trying to figure how to work the program, rather than focusing their attention on understanding the educational content presented.

Like the fraction software, significant differences were also observed among the different spelling programs, $F=7.3$, $p < .01$. The *Spell It! Tutorial* (adjusted *Test 3* mean=82.2) and the *Spell It! Game* (adjusted *Test* 3 mean=79.8) were the most effective programs, followed by *Spellicopter* (adjusted *Test 3* mean=74.7) and *A+ Spelling* (adjusted *Test 3* mean=73.2).

Since the spelling words were constant across the four spelling programs, software features must account for the observed differences. For example, features such as the type of help files and student feedback differed among the spelling software. It may be that these kinds of features play an important role in the effectiveness of educational software and is therefore important for future studies to relate specific software features with significant learning effects.

Tutorial and Game Effects

The academic effects of the educational tutorials and games were also compared in the present study. The tutorial format was expected to be more effective than the game format. This hypothesis stems from several basic cognitive principles. First, research supports the view that multiple forms of representing information increases the likelihood of retrieving that information (Rubenstein, 1975). Second, possessing conceptual knowledge of the underlying principles required to solve a particular type of problem increases the likelihood that the student will also be able to solve related types of problems (Greeno, 1980; Loftus and Suppes, 1972; Norman, 1980; Schoenfield, 1979; Thornton, 1978). Finally, learning basic conceptual principles permits students to acquire and process even more complex concepts in a hierarchical manner, thus expanding their knowledge bases and permitting them to solve even more difficult problems (Chase and Chi, 1980; Greeno, 1973). Since students using the tutorial programs will have exposure to all three of these principles, they should score higher on *Test 3* compared to the students using the games, who will have exposure to only one form of problem representation and little exposure to the conceptual basis of the drill and practice exercises provided in the games.

Table 2 presents the mean test scores by type of software. The fraction tutorials (adjusted mean=

Table 1

Mean Test Scores (% correct) by Subject and Software Order

Software Order	N	Observed Means Test 1	Test 2	Test 3	Adjusted Means* Test 2	Test 3
Fractions						
Group I	108	30.1	38.1	43.1	39.7	43.7
Group II	107	34.7	36.5	40.8	35.0	41.0
Spelling						
Group I	108	67.6	70.8	80.8	71.3	79.8
Group II	107	69.3	82.7	76.2	82.1	74.1

* Mean adusted for *Test 1* scores

Table 2

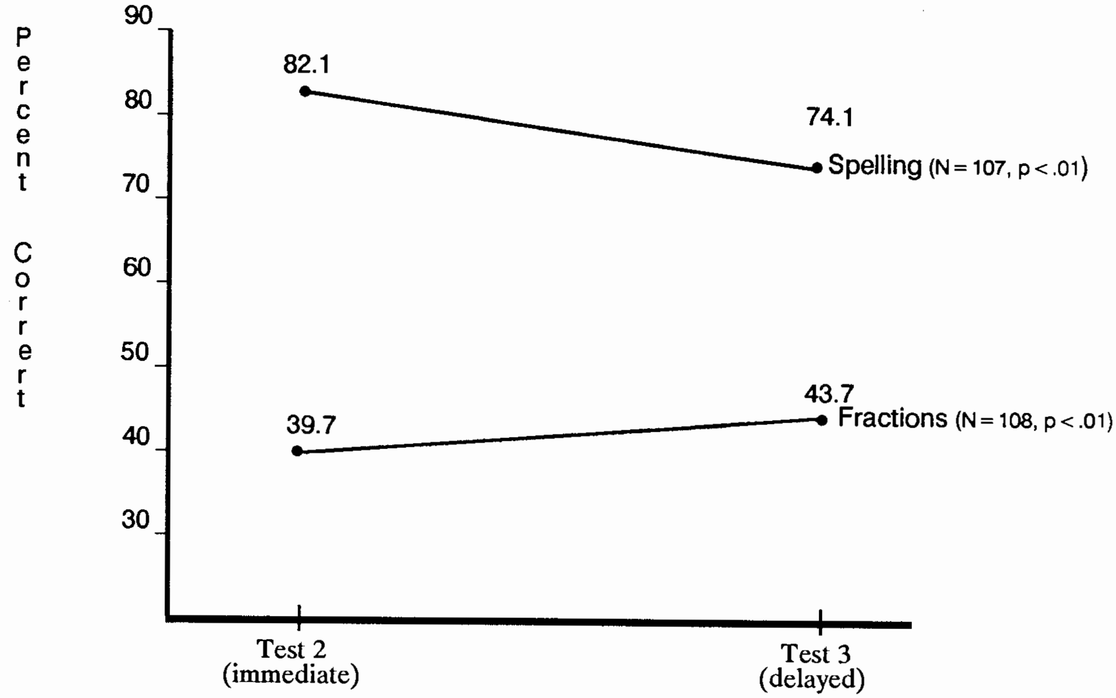
Mean Test Scores (% correct) by Program Title

Program Title	N	Observed Means Test 1	Test 3	Gain	Adjusted Mean* Test 3
Fractions					
Tutorials	136	34.5	46.8	12.3	45.1
SVE Fractions	74	37.5	53.8	16.3	50.4
Eduware Fractions	62	30.9	38.4	7.5	38.7
Games	79	28.8	33.7	4.9	35.1
Galaxy Fractions	41	30.9	37.9	7.0	38.3
Fraction Action	38	26.6	29.1	2.5	31.8
Spelling					
Tutorials	79	66.7	78.1	11.4	77.5
Spell It! Tutorial	47	73.3	86.2	12.9	82.2
A+ Spelling	32	57.0	66.3	9.3	73.2
Games	136	69.5	78.7	9.2	77.2
Spell It! Game	68	71.4	82.5	11.1	79.8
Spellicopter	68	67.6	74.9	7.3	74.7

* Mean adjusted for *Test 1* scores

Figure 1

Immediate and Delayed Test Means for Fractions and Spelling (% correct adjusted for Test 1 scores)



45.1) were significantly more effective than the fraction games (adjusted mean=35.1); F=11.8, $p < .01$. However, the means in Table 2 reveal that this significance was largely due to especially good performance from students who used the *SVE Fractions* program. Students who used the other tutorial, *Eduware Fractions* (adjusted mean=38.7), did not score significantly higher than students who used the fraction game, *Galaxy Fractions* (adjusted mean=38.3). The low scores on the *Eduware Fractions* tutorial are consistent with the poor likability ratings provided by many of the students who used this software. If *Eduware Fractions* was confusing and difficult to use, as reported, the tutorial aspects of the program may have lost their impact while the students were trying to figure out how to work the program.

Unlike the fraction software, there was no significant difference between the spelling tutorials (adjusted mean=77.5) and the spelling games (adjusted mean=77.2). In other words, students who used the spelling tutorials did not learn significantly more than students who used the spelling games. However, it is important to note that comparing tutorials to games in the present study is a weak measure because of the limited generality of comparing only two tutorials and two games for each subject area studied. More outcome evaluations comparing the effectiveness of software tutorials to games are needed to determine if tutorials teach academic objectives better than arcade-style games.

Knowledge Retention Effects

According to the Levels of Processing approach (Craik and Lockhart, 1972), information processed semantically (such as fraction concepts) is more likely to be retrieved from memory than information that is simply memorized (such as spelling words). Since the words on the spelling software in the present study were not presented with underlying concepts and were probably memorized, student knowledge of how to spell the words was expected to decay after they stopped using the spelling software.

Figure 1 presents a graphical representation of the amount of knowledge the students retained about fractions and spelling two weeks after they had stopped using the software. Only students who used the fractions software (Group I) and the spelling software (Group II) the first two weeks

Table 3

Summary Table of Means for Software Effectiveness and Ratings per Software Program

Type[1]/Program	Effectiveness Test 3[2] Scores	SOFTWARE RATINGS			
		STUDENT		TEACHER	
		N	Learning[3]	N	Recommend[4]
FRACTIONS					
T *SVE Fractions*	50.4 ┐**	72	3.3	3	3.0
T *Eduware Fractions*	38.7 ┘	61	2.9	3	2.0
G *Galaxy Fractions*	38.3 ┐*	40	2.9	3	2.7
G *Fraction Action*	31.8 ┘	32	3.3	3	2.3
SPELLING					
T *Spell It!* tutorial	82.2	44	2.6	3	3.0
G *Spell It!* game	79.8 ┐**	64	3.0	3	3.3
G *Spellicopter*	74.7 ┘	67	3.6	3	3.3
T *A+ Spelling*	73.2	27	3.2	3	3.3

[1] T = tutorial, G = game

[2] Adjusted for Test 1 scores.

[3] Mean student rating for the statement, "I learned a lot," with 4 = Strongly Agree, 3 = Agree, 2 = Disagree, and 1 = Strongly Disagree.

[4] Mean teacher rating for the statement, " I recommend this program," with 4 = Strongly Agree, 3 = Agree, 2 = Disagree, and 1 = Strongly Disagree

* $p < .05$ ** $p < .01$

are represented in Figure 1 because for these students *Test 3* represented a two-week delayed test.

Newly acquired fraction knowledge significantly improved with time (adjusted *Test 2* mean=39.7, adjusted *Test 3* mean=43.7, $F=11.8$, $p < .01$), suggesting that students had semantically processed the fraction information and were developing a well-constructed fraction knowledge base that could be used to process new fraction information presented in the classroom. As predicted, newly acquired fraction knowledge did not decrease over time, but newly acquired spelling knowledge did decrease significantly during the two week period when the students did not use the spelling software (adjusted *Test 2* mean=82.1, adjusted *Test 3* mean=74.1, $F=37.5$, $p < .01$). Student memory for the spelling words decayed with time, suggesting that students had memorized the spelling words and had not processed them semantically.

Thus, information processing methods are important cognitive concepts to keep in mind when developing educational software. To produce optimal learning conditions, educational software must provide students with basic conceptual tools that they can use to build and store more complex concepts. In addition, the software must provide students with exercises that encourage them to process new information semantically. When conceptual tools are not provided and students only memorize new information, memory for newly acquired information can be expected to decline fairly rapidly, as we have seen with the spelling words in the present study.

Gender Effects

In the present study, males and females did not differ in learning fractions and spelling skills. Apparently, the schools have introduced computers equally well to both sexes. The only significant sex effect observed was that girls rated the spelling software significantly higher than boys rated the spelling software. In general, the lack of significant sex effects is a positive finding for females, demonstrating that if given equal opportunities in school, girls can succeed in fields that were previously dominated by males.

Validity of Student Software Ratings

Research by Jolicoeur and Berger (1986) suggests that subjective software ratings do not predict how much students will actually learn when using highly rated software. Popular software review services such as EPIE (Educational Products Information Exchange) and Microsift (from the Northwest Regional Educational Laboratory) generally have three or more software reviewers rate software on a four-point scale from Highly Recommended (4) to Not Recommended (1). To test the validity of software recommendation systems such as EPIE and Microsift, teachers in the present study were asked to respond to the statement, "I recommend this program," on a four-point scale from Strongly Agree (4) to Strongly Disagree (1) for each software program. A total of three teachers rated each program. In addition, students rated the statement, "I learned a lot," for the software they used, using the same four-point rating scale as the teachers. Table 3 compares the teacher and student software ratings relative to the actual effectiveness of each program.

The teachers gave the most effective fractions program, *SVE Fractions* (mean rating=3.0), the highest recommendation. However, *Eduware Fractions* (mean rating=2.0), a program that was significantly more effective than *Fraction Action* (mean rating=2.3), was rated lower than *Fraction Action*. The most effective spelling software, the *Spell It!* tutorial (mean rating=3.0), received the lowest teacher recommendation, and the least effective spelling program, *A + Spelling* (mean rating=3.3), was among those recommended the most highly by the teachers. As predicted, teacher recommendations of software do not appear to represent a valid measure indicating *how much students will learn* using a specific educational software program.

Students were also not able to judge the educational value of the software that they used for two weeks. Partial correlations of *Test 3* scores with student "learning" ratings (adjusting for *Test 1* scores) were calculated to determine the validity of the ratings. There were no significant correlations (partial r=.06 for fractions, and partial r=.01 for spelling). The students were not accurate in their estimates of how much they had learned using the educational software. Since students are not accurate at predicting the educational value of the software, they should not be left on their own to determine the order and parts of a software program they will use. Instead, they should be led through the software in a sequential manner that best matches the intentions of the software developers and the classroom's curriculum.

Summary

The present study demonstrates a system for implementing educational software into schools, incorporating a sound method for objectively measuring the impact of different software programs on student learning. There is a critical need for schools to measure the impact of different software on student learning because this is, currently, the only valid means available for separating effective educational software from noneffective software. A large body of educational and psychological research identifies characteristics of software that should be related to educational effectiveness, but at this point these features have not been systemically evaluated in the context of microcomputers in classrooms. Subjective judgments of the effectiveness of software are surprisingly poor. A previous review by Jolicoeur and Berger (1986) demonstrated that ratings from software review services are not valid indicators of the educational value of software. The current research shows that teachers and students are not able to judge the instructional value of software *even* after they have used it for several hours. Although we have ideas and hypotheses about specific software features that account for high quality educational software, the majority of our ideas and hypotheses have not been verified using objective measures. We know that many studies have shown educational software to be an effective learning medium. However, at this point in time, we still do not know *why* the software is effective.

There is a pressing need, then, for objective information on the effectiveness of educational software programs. When large numbers of educational software programs have been measured for academic effectiveness, it should be possible to identify the software features that make some software programs more effective than others. □

References

Becker, H.J. *The Impact of Computer Use on Children's Learning: What Research Has Shown and What It Has Not* (Report No. 18). Baltimore, MD: Johns Hopkins University, Center for Research on Elementary and Middle Schools, 1987.

Chase, W.G., and Chi, M.T.H. Cognitive Skill: Implications for Spatial Skill in Large Scale Environments. In J.

Harvey (Ed.), *Cognition, Social Behavior, and the Environment.* Hillsdale, NJ: Erlbaum, 1980.
Craik, F.I.M., and Lockhart, R.S. Levels of Processing: A Framework for Memory Research. *Journal of Verbal Learning and Verbal Behavior*, 1972, *11*, 671-684.
Greeno, J.G. The Structure of Memory and the Processes of Solving Problems. In R. Solso (Ed.), *Contemporaty Issues in Cognitive Psychology*. The Loyola Symposium. Washington, DC: Winston, 1973.
Greeno, J.G. Some Examples of Cognitive Task Analysis with Instructional Implications. In R.E. Snow, P-A. Federico, and W.E. Montague (Eds.), *Aptitude, Learning, and Instruction. Volume 2: Cognitive Process Analyses of Learning and Problem Solving.* Hillsdale, NJ: Erlbaum, 1980.
Jolicoeur, K., and Berger, D.E. Do We Really Know What Makes Educational Software Effective? A Call for Empirical Research on Effectiveness. *Educational Technology*, December 1986, *26*(12), 7-11.
Kulik, J.A. Consistencies in Findings on Computer-Based Education. Paper presented at the annual meeting of the American Educational Research Association, Chicago, Illinois, 1985.
Kulik, J.A., Bangert, R.L., and Williams, G.W. Effects of Computer-Based Teaching on Secondary School Students. *Journal of Educational Psychology*, 1983, *75*, 19-26.
Kulik, J.A., Kulik, C.C., and Cohen, P.A. Effectiveness of Computer-Based College Teaching: A Meta Analysis of Findings. *Review of Educational Research*, 1980, *50*, 525-544.
Loftus, E.F., and Suppes, P. Structural Variables That Determine Problem-Solving Difficulty in Computer-Assisted Instruction. *Journal of Educational Psychology*, 1972, *63*, 531-542.
Niemiec, R., and Walberg, H.J. Computers and Achievement in the Elementary Schools. *Journal of Educational Computing Research*, 1985, *1*(4), 435-440.
Niemiec, R., and Walberg, H.J. Comparative Effects of Computer-Assisted Instruction: A Synthesis of Reviews. *Journal of Educational Computing Research*, 1987, *3*(1), 19-37.
Norman, D.A. Cognitive Engineering and Education. In D.T. Tuma and R. Reif (Eds.), *Problem Solving and Education: Issues in Teaching and Research.* Hillsdale, NJ: Erlbaum, 1980.
Rubenstein, M.F. *Patterns of Problem Solving.* Englewood Cliffs, NJ: Prentice-Hall, 1975.
Schoenfeld, H.H. Explicit Heuristic Training as a Variable in Problem Solving Performance. *Journal of Research in Mathematics Education*, 1979, *10*, 173-187.
Thomas, D.B. The Effectiveness of CAI in Secondary Schools. *AEDS Journal*, 1979, *12*, 103-116.
Thornton, C.A. Emphasizing Thinking Strategies in Basic Fact Instruction. *Journal for Research in Mathematics Education*, 1978, *9*, 214-227.

Interactive Technology Book. A new anthology on interactive video and interactive technology in general is now available from ETP. Interactive Video is priced at $24.95 and contains 30 articles published in recent issues of this magazine.