DOCUMENT RESUME

ED 093 389 IR 000 900

AUTHOR Halley, Fred S.; And Others
TITLE Computer Applications in Teaching and Learning.
INSTITUTION State Univ. of New York, Brockport. Coll. at Brockport.
PUB DATE 23 Oct 73
NOTE 23p.; Paper presented at the Rochester Area Colleges Workshop on the Use of Computers in Instruction (Rochester, New York, October 23, 1973)

EDRS PRICE MF-$0.75 HC-$1.50 PLUS POSTAGE
DESCRIPTORS *Computer Assisted Instruction; *Computer Oriented Programs; *Computers; *Evaluation; *Grading

ABSTRACT

Some examples of the usage of computers in teaching and learning are examination generation, automatic exam grading, student tracking, problem generation, computational examination generators, program packages, simulation, and programing skills for problem solving. These applications are non-trivial and do fulfill the basic assumptions necessary for education. These assumptions and goals are that the individual is of greatest importance, cost per pupil should be minimized, cheating is bad, the student should know what is expected of him, the normal curve is an inappropriate approach to grading, and cooperation should be encouraged. An item pool, known to all students, from which individual tests are generated would fulfill these conditions and lessen the fear of computers in many students' minds. (WH)

COMPUTER APPLICATIONS IN TEACHING AND LEARNING

By

Fred S. Halley
SUNY, Brockport

with

Ronald L. Koteskey, Asbury College
James M. Swanson, University of Texas, Austin
Robert J. Tracy, DePaul University

Pearson and Reicken (1970) have stated that, "Computer technology, perhaps more than any other means at the present, holds promise that we can deal with the educational problems of today and tomorrow in a significant fashion." This view is not shared by many who feel the computer has a depersonalizing and alienating effect. The position of this paper is that the computer is a tool, neither good nor evil; and, like the slide rule or calculator, can be used efficiently to save a great deal of time and do things that for all practical purposes would be impossible without it; or it can be a waste of time and money, like using an expensive calculator to multiply a number by 1000 (Oettinger and Marks, 1969; Mesthene, 1970). This paper attempts to encourage the use of available computer technology by reporting successful, yet nontrivial, computer applications to teaching and learning.

Assumptions and Goals

At this point, let us make a series of assumptions about education. These are in the form of unsupported (in this paper) value judgments which could be easily challenged. These will be used to evaluate the traditional educational practices and as a basis for judging the computer at the end of the paper.

The individual is of greatest importance. "Traditional lecture discussion, lock-step classes are potentially very depersonalizing" (Keller, 1968). Each student is treated exactly the same; therefore, the slow are lost, the fast are bored, and the fortunate "average" few students learn the most. Everyone (the prepared and unprepared) is studying the same thing, at the same time, going at the same rate of speed, etc. Of course, to receive credit in a course, the students should have studied the same content; but it may be inconvenient or impossible to study it correctly when the teacher says so. The student may not have the necessary background, or he may already know it and have nothing to do until a new topic is introduced. If everyone were the same, the lock-step method would be fine, but it completely ignores the importance of the individual. A solution to this problem is to have a very low student-teacher ratio or to use student proctors as in the Keller plan.

Cost per pupil should be minimized. In this day of rising teacher salaries and tightening educational budgets, attempts to lower student-teacher ratios to increase individual attention are soon halted. On-line computer assisted instruction (CAI), is at the present time expensive (PLATO) and has not proven to be superior to other methods of instruction including programmed textbooks. Kemeny (1971) has pointed out that CAI is often a misuse of expensive computer time. Efficient and effective use of the computer in the educational process is a critical need since traditional education will become increasingly costly per pupil as teachers' salaries increase.

Cheating is bad. Aside from guilt feelings and lowering (or raising, depending on the value system) of self-esteem, the student who cheats is unlikely to learn something essential. The traditional method of teaching

with everyone studying the same material, working the same problems, writing papers or reports on the same topic, and taking the same tests encourages cheating and makes it, from the student's viewpoint, a practical enterprise. Answers are compared, papers and reports are copied or purchased, and test questions move quickly along the student grapevine if not already in the files of fraternities and sororities. Students pass courses, get good grades, but learn little.

The student should know what is expected of him. Many teachers do not specify what they expect of the student other than that he "know everything" from the textbook and lectures. So, of course, the student does not know specifically what he should be responsible for. The teacher who has not specified behavioral objectives decides that the "ideal question" is one that discriminates the "good" from the "bad" students, i.e., half the students miss it. The basic strategy in making up a test question is not "What should the student know?" but "Will about half the students miss this?" Of course, this means that the teacher has failed with half his students. Therefore, teachers resort to footnotes, figure captions, insignificant details, etc. The best thing to do is to keep the student confused as to what he is to know. This situation may be avoided through proper use of criterion-referenced grading procedures (Holtzman, 1970).

The normal curve is an inappropriate approach to grading. Since the instructor typically has not specified behavioral objectives and students miss about half the questions on the typical well-constructed test, he resorts to the normal curve to determine grades. Although the normal curve is very useful in inferential statistics, often class grades obtained from a highly selected and usually small sample such as college students do not conform to a normal curve. Furthermore, if the student is in a slow class,

he may get an "A" for the same work that would result in a "C" in a bright class. An alternative is to use a criterion-referenced procedure instead of normative grading (Holtzman, 1970).

Cooperation should be encouraged. In the traditional classroom competition often means that when someone wins (gets an "A"), someone else must lose (get an "F"). Students who have been exposed to normative grading may become reluctant to help others because they realize their grade is depending on their relative position in the class rather than on how much they know. As a result of normative grading, peer teaching and learning is minimized even though this is a valuable and effective pedagogical technique. The clear specification of behavioral objectives makes the teacher an ally rather than merely a judge in the attainment of these objectives.

Summary. The value judgments implicit in the above discussion are shared by many who use traditional educational methods. It is the experience of the authors that the goals stated above are not incompatible with computer technology and, in fact, computer applications may enhance the attainment of these goals.

Types of usage for computer applications in teaching and learning

While advances in computer and educational technology have created attractive new possibilities for the application of new knowledge in an effective manner, it has been stymied by strong resistance to change (see Grayson, 1972; Oettinger, 1966). The applications given below have been used successfully in actual classroom settings by one or more of the authors of this paper. There is no set pattern for implementing these techniques; the specific applications are dependent on the type of course, the instructor's goals, and student acceptance. The applications may play a major or minor

part in a course; they may be used singularly or jointly to achieve instructional goals. Regardless of techniques opted, successful implementation is dependent upon both the instructor's and the student's understanding of how these techniques are to be used in the attainment of beneficial instructional goals. The applications listed below are intended as unobtrusive means to an instructional end rather than ends in themselves.

1. Examination Generation. The computer may be used to generate exams which can test student comprehension of subject matter in a variety of disciplines. This is most useful in courses where student understanding may be measured by means of multiple choice, true-false, or completion questions. Under control of the instructor, a question pool is organized into (groups representing) course content areas and stored in the computer. Programs are available which randomly sample within each content area to produce different, but equivalent exams for each student and separate answer keys for the instructor (Wagener, 1973; Koteskey, 1972).

A variant of the above technique is to design the question pool so that it may be added to by several instructors rather than remaining the product of a single instructor. This is most useful when introductory sections of a course are being taught by different instructors and each may review and add to the question pool. To generate exams the instructor specifies content areas to be sampled and can even specify specific questions from each content area. This cooperative technique involving several instructors eliminates the need for examination security since such a large pool of questions makes cheating without knowing the subject matter impossible. Some instructors even make the total question pool available to the students as a study aid.

2. Automatic Exam Grading. For multiple choice examinations produced

by any method, the computer may be used to score and evaluate student performance. To do this students write exam answers on optically scannable sheets or mark-sense cards which can be processed by the computer. This procedure is applicable when a single exam is given to an entire class or when each student receives a different exam; in either case, answer keys produced during exam generation are used to score the student's tests (Towle, et al., 1973). Given the decline in support for graduate assistants, computerized grading is especially convenient for the instructor and in large enrollment classes is often a necessity. Automatic grading programs can be used to provide summary statistics on student performance such as histograms, standardized scores, and item analyses which may be desired by the instructor.

3. Student Tracking. In large classes it often becomes difficult to maintain a record keeping system which enables the instructor to keep track of the progress of his students. The computer may be used as a dynamic grade book and filing system which may be updated and corrected by the instructor. It is particularly useful in self-paced or self-instructional courses (Bruell, 1972). Such systems can produce graphs of class performance, compute scores, standardize test scores, and assign grades. The student data bank may be interrogated to find students who may be having difficulty with the course so that they may be offered extra help (Castellan, 1973).

The above applications of the computer can be used in practically any course; the following applications are better suited to statistical or methodology courses since they are based upon numerical methods.

4. Problem Generation. Often it is desirable to supply individual students with separate sets of data for purposes of analysis and interpre-

tation. Using the computer, it is possible to generate data sets by drawing random samples from populations, a procedure which produces a unique exercise for each student and thus discourages plagiarism, while at the same time offering considerable convenience for the instructor (Halley, 1972). Programs may be elaborate so that entire word problems are printed on computer paper ready for student use, or more simple programs may be used which create numeric outputs to be attached to pre-printed problems.

For each student exercise that is printed, a corresponding solution can be produced for the instructor's use in checking and grading. The solutions can contain not only the final answer to problems, but also intermediate summary statistics so that students' computation errors may be located and partial credit assigned. The printed output of the program can be designed so that the solutions can be conveniently separated from the exercises and the exercises easily separated from one another for distributing to students.

The programs are designed for the convenience of the instructor, and typically require no preparation of raw data. Instead, the instructor enters such information as the number of exercises to be produced, the difficulty level of exercises, and the parameters (e.g. μ, σ) of the populations from which the data values are randomly generated. The parameter values can be chosen to produce marginal statistical significance, causing some of the generated exercises to yield statistically significant results and others to be nonsignificant.

5. Computational Examination Generators. The above applications are primarily designed for homework, drill and practice. However, the same general techniques may be utilized for exam generation and evaluation of

student performance. This use of computers offers many of the same advantages as the problem generation procedure discussed above, specifically to produce individualized exams, discourage cheating, and to make exam production more convenient for the instructor. Koteskey (1972) has further developed these procedures as a means of grading students solely in terms of the amount of material they master. Students are allowed to proceed to a new course topic only after they have mastered earlier material. Realizing this, the students study to learn the material well. The students take a test only when they are ready, so there is no such thing as a make-up test. The instructor meanwhile does not have to spend time making up tests and keys since these are produced by the computer.

6. Program Packages. Program packages, often referred to as "canned programs" are generalized computer programs written and stored on a computer for use by persons not familiar with a programming language (Anderson and Coover, 1972). For each program, there is documentation which instructs the person how to use the program. Program packages may be designed to run in batch or interactive modes, the latter mode being the most useful for student learning (Dixon, 1971). Some are quite extensive and do most any type of data analysis or statistical testing, while others are more specific and handle special applications.

Packaged programs are most useful in courses where it is not desirable to teach students computer programming, but it is desirable to have students use the computer. Once students are exposed to the utility of packaged programs to accomplish their objectives, they become interested in learning more about computers and computing languages.

Program packages may be applied in research methods courses where students analyze data they have collected. In courses where simulation

packages (see discussion below) are used, students may use "canned" statistical programs to analyze the outcome of simulated experiments (see Halley, 1972; Swanson, 1973). Also in courses where it is desirable to analyze textual material, programs for content analysis have been developed (Sedelow & Sedelow, 1972).

7. Simulation. Computer programs have also been used to simulate certain phenomena. In the area of statistics, for example, programs have been used to conduct sampling experiments based on random samples drawn from normal populations with specified parameters (e.g., Appelbaum & Guthrie, 1970; Garrett, 1970; Lohnes & Cooley, 1968; Wikoff, 1970). It is possible to systematically vary sample sizes and different parametric values to study their effects on a number of theoretical propositions. Such experiments, perhaps conducted in class, might be used to provide students with an intuitive understanding of theoretical concepts.

In addition, simulation techniques are being used to train students in experimental design in specific content areas (Main & Head, 1971). This technique is used to introduce students to research design in a wide spectrum of the social sciences, encompassing areas in which research facilities may be limited (subject pools, apparatus, space, time, and cost constraints). The emphasis is on sequential experimental design in which students design subsequent studies based on prior experimental outcomes. For example, students are allowed to experiment with a model of the "cocktail party problem", concerning the ability of a person to attend to the conversation of one speaker in the midst of one or more other speakers. A computer model has been constructed to simulate current knowledge of this phenomenon and students are allowed to choose experimental designs and to input the design parameters into the model. For example, in one version of the model,

students are allowed to manipulate such variables as: (1) the number of Ss to be run; (2) whether the accepted message (the message to be attended to) and the ignored message are presented to both ears or whether the two messages are directed to different ears; (3) the relative loudness of the messages; and (4) the content of the two messages. The model simulates the performance of the requested number of Ss under the specified conditions.

To complete the experiment, one or more additional groups of Ss are simulated using different settings of the independent variables. The model outputs performance scores for each S in each group, and the student may be required to interpret or to analyze the data in order to test his hypotheses. There are a number of other simulations which have been developed, dealing with imprinting, schizophrenia, and obesity.

8. Programming Skills for Problem Solving. In this application, the student actually writes programs in a computer language. The availability of easy to learn programming languages frees the student from "busy work" at desk top calculators and enables the learning and understanding of statistical procedures to proceed at a more rapid rate. The implementation of these techniques is dependent upon the availability of easy-to-learn languages (BASIC) and access to time-sharing terminals (Kemeny, 1971). In addition, interpretative languages have been developed (OMNITAB, SMIS, Hogben, 1970; Castellan, 1970; Swanson, et al., 1973) which may be used to write programs and include access to packages.

Critique of Computer Applications

As can be seen from the variety of techniques above, computers allow wide variation in how the courses are conducted. Even though the

courses are conducted differently, they all embody the basic philosophy stated in the first part of this paper.

Individualization. Contrary to what has been predicted, intelligent use of computers does not force people into the same mold. Individualization may, in fact, be enhanced by individualized testing which allows students to proceed at their own speed, repeatable testing which allows individual mastery of materials, unique problem generation which encourages each student to do the required work, and the provision of program packages which allow individual research and experimentation (statistical packages and simulation routines).

However desirable individualization may be, there are possible disadvantages. Individualization places extreme responsibility on the student to pace himself. Without the traditional prodding of the instructor, students often procrastinate and get hopelessly behind with the attendant consequences that they receive lower grades. Some would argue the desirability of student responsibility being reflected in course grades. Others would maintain that people need external pacing for attainment of difficult but important tasks. A method of accomplishing this latter is by providing a series of semi-threatening deadlines. With the exception of these deadlines and the end of the quarter, each student moves at his own speed.

Furthermore, a disadvantage exists for the instructor as he may become less involved with his more interested and highly motivated students. The better students tend to complete individualized courses without contacting the instructor; the less able and motivated students tend to have extensive contact with the instructor, thereby distorting his view of overall student performance. A possible solution to this difficulty is to enlist the aid of the better students as tutors.

Cost Factors. A disadvantage of the techniques discussed above is that their initial implementation is costly in terms of time and expertise. The expertise factor may be less of a problem since many packages are currently under development and hopefully these packages can be transported to the user's system. There still remains, however, the time expenditure for the individual instructor who must familiarize himself with available materials and decide, by consulting with his computer center staff, whether available packages can be implemented on his particular system. This is particularly problematic since these efforts are not recognized by colleagues as legitimate professional activities and thus are not rewarded.

Once the packages are up and working on the local computer system, the packages can have a dramatic effect on reducing instructional costs, especially if large class sections are being taught. Less instructor time is required in preparing exams. Less secretarial service is needed since exams are computer generated, and fewer assistants are needed because the computer scores the tests and keeps student records.

Cheating Minimized. Because exams and exercises are individualized, copying another student's answers is impractical. Furthermore, if the question pool is published in advance, there is no profit in students keeping examination files.

Computer Applications Encourage Specification of Course Goals. When the instructor goes through the process of designing a question pool to be placed in a computer file, he is forced to be specific concerning what the course covers. This clarification process enables him to more effectively communicate course expectations to students. With mystery removed, the student can now concentrate on learning the important material in the course. Similarly, mystery concerning the method of grading is removed

as the criteria for grading is clearly specified. Moreover, use of the computer allows more rapid feedback on student performance.

One must be cautioned that a typical clarity in statement of course goals and standards makes the instructor a ready target for criticism by those who are less definite.

Criterion Grading. Student grades now depend on absolute achievement, not on how well the rest of the class has done. The standards are set for each grade, and anything from all "A's" to all "F's" may be given. When the student receives a given grade, it means that he can do certain specified things, not that he fell at a given point in some unspecified class; thus, the grade has a different connotation than traditionally.

Cooperation Emphasized. Since there is no competition between students using these methods, cooperation is encouraged. Peer teaching and learning occurs openly and freely without the threat of one student's grade being enhanced at the expense of another's. Such cooperation is not encouraged under normative referenced procedures.

Without the threat of normative grading, the students feel that the professor is more fair and not to be flattered, out guessed, or fooled; rather, the professor becomes an ally to be cooperated with in the attainment of clearly stated course goals. In this setting without the distraction of conflict, rivalry, and mysterious grading procedures, *esprit de corps* is likely to develop within the class.

Use of the computer produces a familiarity which promotes a positive attitude toward computers and their usage. The following note was received by an instructor during a course using these techniques:

"Dear Computer:

Congratulations! You couldn't have picked a harder eight questions. May your transistors corrode, your wires melt, and your ability to pick easier questions improve.

Your truly human friend,

Doug"

References

Appelbaum, M. I. and Guthrie, D. Use of Computers in Undergraduate Statistics Instruction. Conference on Computers in the Undergraduate Curricula, 1970. Iowa City: Center for Conferences and Institutes, The University of Iowa, 1970.

Anderson, Ronald E. and Coover, Edwin R. Wrapping Up the Package: Criticat Thoughts on Applications Software for Social Data Analysis. Computers and the Humanities, 7, 2, 1972.

Bruell. CUMREC - An Interpretive Program for Cumulative Records in the Classroom. Project C-BE, Technical Report #4, University of Texas, 1972.

Castellan, N. John, Jr. SMIS/SIS: The Symbolic Matrix Interpretive System and Statistical Interpretive System. Common: Seattle Proceedings, 1970, 64-73.

Castellan, N. John, Jr. GRADER: An Automated Gradebook System. Educational Technology, 1973, 13, 4, 56-60.

Dixon. (1971)

Garrett, H. G. SAMDS: A Program to Generate Empirical Sampling Distributions of the Mean. Conference on Computers in the Undergraduate Curricula, 1970. Iowa City: Center for Conferences and Institutes, The University of Iowa, 1970.

Grayson. Science, 1972.

Halley, Fred S. Individualized Instruction in Basic Statistics: An Experiment in Computer Managed Instruction. Proceedings of the 1972 Conference on Computers in Undergraduate Curricula, 1972, pp. 533-538.

Halley, Fred S. A Social Science Program Library for Small Computers. Proceedings of a Fourth Conference on Computers in the Undergraduate Curricula, 1973, pp. 232-235.

Hogben, David, Peavy, Sally, and Varner, Ruth. OMNITAB II: User's Reference Manual. National Bureau of Standards, Technical Note #552, 1971.

Holtzman, Wayne H. Computer Assisted Instruction, Testing and Guidance. Harper and Row, 1970.

Kemeny. Uses and Misuses of Computers in Instruction. Mimeograph, 1971.

Kemeny. Computers at Dartmouth.

Lehman, Richard S. The Use of the Unknown in Teaching Statistics. E.P.A. Convention. Boston, Massachusetts, 1972.

References (Continued)

Lohnes, P. R. and Cooley, W. W. Introduction to Statistical Procedures with Computers Exercises. New York: John Wiley and Sons, Inc., 1968.

Keller. Goodbye, Teacher. 1968.

Main and Head. Computer Simulations in the Elementary Psychology Laboratory. Conference on Computers in the Undergraduate Curricula, 1971. Hanover, New Hampshire: University Press of New England.

Mesthene. How Technology Will Shape the Future. Science, 161, 1968, 135-143.

Oettinger, A. G. and Marks, S. Run, Computer, Run. Harvard Studies in Technology and Society, No. 1, Cambridge, Massachusetts: Harvard University Press, 1969.

Oettinger, A. G. A vision of Technology and Education. Communications of the ACM, IX, 489, 1966.

Pearson and Reicken. Computer-Assisted Instruction, Testing and Guidance. (Wayne H. Holtzman, Ed.) New York: Harper and Row, 1970.

Sedelow, Sally Y. and Sedelow, Walter H. Language Research and the Computer. University of Kansas, Lawrence, Kansas, 1972.

Swanson, James M., Riederer, Stephen and Weekly, Howard. Using Omnitab to Teach Applied Statistics. Proceedings of a Fourth Conference on Computers in the Undergraduate Curricula, 1973, pp. 1238-134.

Towle, Nelson J., Cohen, Perrin S. and Cohen, Leila R. A Case Study in the Use of Computers to Personalize Instruction. Proceedings of a Fourth Conference on Computers in the Undergraduate Curricula, 1973, pp. 135-138.

Wagener, Jerrold L. CAGES: An Approach to Computer-Assisted Generation and Evaluation of Examinations. Mimeograph. Brockport College, Brockport, New York, 1973.

Wikoff, R. L. Using the Computer in Basic Statistics Courses. Conference on Computers in the Undergraduate Curricula, 1970. Iowa City: Center for Conferences and Institutes. The University of Iowa, 1970.

APPENDIX A

COMPUTER ASSISTED TEACHING AND LEARNING APPLICATIONS

by James M. Swanson, UT, Austin

I. Repeatable Tests in a Human Learning Class.

In this upper-division undergraduate class, I use the computer to generate multiple choice exams which each student must take (and retake if necessary) until a criterion of 90% is attained. In the context of the grading scheme of this course, completion of this mandatory requirement earns the student a grade of "C"; to receive a higher grade ("B" or "A"), a student must complete one of five optional special projects for each letter increment in grade. These options include summaries of speakers who are visiting the Texas campus and who are relevant to the course material -- a paper on a topic that is personally interesting to the student which was covered in class, an in-depth coverage of one psychologist's contribution to our knowledge of the field of human learning (to include from 5-10 references to journal articles by that psychologist), etc. Topics vary somewhat from year to year. Students who do not complete the mandatory test requirement receive either a "D" or an "F", depending upon their actual performance level.

The class records are kept by computer in this course using a program developed by Dr. Jan Bruell of the Psychology Department at Texas. This program (CUMREC -- Cumulative Record) not only keeps records, it also scores exams which are taken using mark-sense answer sheets. This scoring and record-keeping relieves me and my teaching assistant of the drudgery of keeping a record book of all of the accomplishments of each student. Since the enrollment of this course varies from 100 to 150, this alone is ample reason to be enlightened and to use the computer.

The most important part of this course is the repeatable testing as far as pedagogical strategy goes. Students are required to read a text and gain the basic knowledge about the subject matter of human learning, and are directed along this path by receiving feedback from the computer concerning the question missed, concept covered by that question, and page number in the textbook where the concept was discussed. This allows me to use a different sort of approach in lecture. I use lectures (a maximum of two a week) as a motivating force just as Keller has suggested in his outline of the proctorial system of instruction which he has popularized. Students are not typically tested on material covered in lecture (although I do try to summarize the assigned text material briefly, and I do answer questions in class about problem areas in the test). However, due to the optional requirements set up for attaining a high grade in the course, students do attend lecture regularly -- and in this context, do get ideas for an optional topic.

One important technique I use to encourage preparation for tests (so a student will not be tempted to take a test before actually reading or studying the material covered) is to give a letter increase in the final grade if a student averages 90% on tests on the first attempts (usually with one drop allowed for unavoidable screw-ups).

II. A Computerized Laboratory for Statistics.

I teach students (both graduate and undergraduate) how to use the language omnitab in the context of statistics courses. Students attend lectures and work simple homework problems just as in a regular course in statistics, but they also attend a lab session once a week. Each lab session has two purposes: (1) some principle of statistics -- like the sampling distribution of the mean, for example -- provides the subject matter for the lab, and this topic is made clear to the student before attending lab; and (2) additional features of the Omnitab language are added as the lab sequence progresses to make the students more sophisticated and efficient programmers.

When a particular Omnitab exercise is selected by a student, the following events occur: (1) an Omnitab program (written by the instructor) is automatically executed to generate data and to present instructions to the student user; (2) then, control is transferred to the student who solves the lab problem by writing and executing an Omnitab program; and (3) output of the student's program is turned in to the instructor for evaluation.

The Omnitab language is particularly well suited for this application because the initial instruction time required to get students on the machine is very small. Input and output is format free as far as the novice programmer is concerned, and actual statements are issued as commands using simple English sentences. My use of Omnitab in the statistics class is almost exclusively in the time-sharing mode, and error messages are given when syntactic errors are made. This allows a programmer to make the correction immediately, and complete the run without starting over at a future time.

III. Use of Packages in a Free Consulting Service.

I usually recruit the five best students from my graduate statistics class to enroll in a class called applied statistics and consulting. In this class, we (the five students and myself) solicit real data from the faculty and graduate students in the department. We take only data that is ready to be inputted to a program (or we tell the researcher how to put the data into the appropriate form); our basic role is to direct the users of this service to the packaged program relevant to his or her needs and to interpret the output of the program statistically.

This course serves two purposes. First, it gives the five students taking the course valuable experience with a wide variety of data; and second, it provides the department with a free, centralized source of statistical expertise and some confidence in an area in which most are deficient.

APPENDIX B

COMPUTER ASSISTED TEACHING AND LEARNING APPLICATIONS

by Ronald L. Koteskey, Asbury College

1. Introductory Statistics.

Each student reads a standard textbook and works the problems in the accompanying workbook. He is also required to attend about one lecture each week in which each chapter is discussed in a rather intuitive way so that he will at least be exposed to all topics in the course. Sample copies of the computational exams are posted, and the student can take a test over a given unit whenever he has completed that section of the workbook and feels prepared to take the test. He must complete the test perfectly, with the exception of one calculation error. When he has passed this computation test, he is given a computer generated multiple choice "conceptual" exam. The student's conceptual ability seems more limited than his calculation ability, so he is only required to get about 80% correct on this test before going on.

Grades are given on the basis of the number of units passed. There are six units in the course, and a final exam which covers basic material in each of the first five units. The final exam must also be done perfectly. A complete description of the course, its unity, how it is conducted, and grading is found in Koteskey (1972).

2. Social Psychology.

The students read a text and attend lecture-discussions three days each week. They are given all the questions (all essay) on the tests, about 500 for the entire course. A few of the questions are factual (list, describe), but most contain words and phrases, such as explain, compare, contrast, what are the implications, give examples from your own experience, etc., with the answers not found directly in the book. Four tests are given in the course with each test made up of eight questions selected in a stratified random manner from about 125 of the questions. Students can take or repeat tests one day each week; those not taking tests ask questions and engage in discussion with the instructor. The grading scale is distributed at the beginning of the course, and students take tests whenever they are prepared and repeat them until they achieve the score they want on that test.

3. Physiological Psychology.

The students read a text and are given a 200 page workbook which includes completion, multiple choice, and essay questions covering every concept in the text. Six tests and a comprehensive final exam are given with each test consisting of six completion questions, three essay questions, and three multiple choice questions. The completion and essay questions on the test are taken directly from the workbook; the multiple choice questions are ones the students have not seen before, but are similar to the ones in the workbook.

Each day the class consists of a half-hour lecture-discussion and a 20-minute period when students can take tests or talk with the instructor individually. The grading scale is posted at the beginning of the course; however, to discourage unnecessary repetition of tests, points are subtracted each time the test is repeated. That is, an "A" cannot be made on a given test on the second repetition or after that.

4. Experimental Psychology, Advanced Experimental Psychology, and Psychology of Motivation.

The computer application is nearly the same in all these courses with only the number of tests given being different in each course. Of course, the laboratory parts of the courses differ greatly. The grading scale is posted, and the tests are fully repeatable three times only, with each test consisting of six essay questions. They are allowed only the limited number of repetitions to keep each student from getting too large a sample of the questions. Since these are smaller classes, students are allowed to take tests any time they wish, and class time is not used for testing. No penalty is given for repeating tests since they are all essay, and the students have not seen the questions. To keep the questions from circulating, students are asked to sign a statement concerning this each time they take a test. The reason for not giving out all the questions is that there is still not a large enough pool of questions available covering everything in these courses.

5. General Psychology.

This course has a test over each chapter consisting of twelve multiple choice questions drawn from the pool of questions that come in the instructor's manual with the text. Since chance can play a large part in a test score, only two repetitions are allowed, and there is no penalty for repeating except that the student's final score on each test is the one he got the last time he took it even if it is lower than when he took it earlier. Class time is again divided into a 30-minute lecture-discussion and a 20-minute testing-grading-discussion time. Since these are multiple choice tests, and the computer makes up a key for each test, they can be graded and returned to the student at once.

APPENDIX C

COMPUTER ASSISTED TEACHING AND LEARNING APPLICATIONS

by Robert J. Tracy, DePaul University
Chicago, Illinois

Four programs were designed to aid in developing student's statistical skills by generating exercises on: (1) z scores; (2) Pearson r correlation with regression or matched groups t tests; (3) independent groups t tests; and (4) chi square. The programs generate exercises by drawing random samples from populations, a procedure which produces a unique exercise for each student, and thus discourages plagia For each student exercise that is printed, a corresponding solution is oduced for the instructor's use in checking and grading. The solutions contain not only the final answers to problems, but also intermediate summary statistics so that students' computation errors may be located and partial credit assigned.

The programs have been used for a number of years to produce assignments for elementary statistics courses, taken by psychology, sociology, and nursing majors at DePaul University. The exercises have been used in a fashion similar to that presented in Figure 1. For this particular exercise, the same instructions and experimental situation dealing with the effect of TV violence are presented to all students -- though it should be noted that appropriate parametric alteration can also produce data suitable for the simulation of a variety of experimental studies. Irrespective of the situation described, each student receives a different set of data generated by the computer. Usually the population parameters are chosen so that about half of the exercises require that the null hypothesis be rejected and the other half require it to be accepted. An important aspect of this exercise is that students are not told what statistical test to perform. Instead, they must decide for themselves which statistical technique is appropriate, a task which will confront some of them as future researchers.

EXERCISE 2

X:	116	92	95	91	77	107	78
Y:	43	54	58	45	62	45	60

1) State all the assumptions necessary to perform the statistical test. Do these assumptions appear to be met? (Even if the assumptions do not appear to be met, answer the following questions.)

2) State the null and alternate hypotheses.

3) Give your decision rules.

4) Show the necessary computation. Since p lal credit may be earned, it is to your advantage to show the appropriate formulas having substituted numerical values for the symbols, e.g.,

$$S = \sqrt{\frac{N \Sigma X^2 - (\Sigma X)^2}{N(N-1)}} = \sqrt{\frac{11\ (293)-3049}{11\ (11)-1}}$$

5) Give your decision regarding your acceptance or rejection of the null hypothesis.

Figure 1. Sample Exercise

Some computational practice is beneficial for students. I believe that these exercises promote the learning of statistical terminology and operations, a more accurate use of statistical formulae, and perhaps a better understanding of statistical theory as it relates to the formulae. It is clear, however, that computational practice should not be excessive, least it become drudgery and detract from learning other statistical concepts. As a student becomes proficient in a certain computation, he can be introduced to canned programs which produce the same computations. This approach allows a student to check his calculations with those of the computer. The comparison will heighten appreciation of the computer's speed and accuracy, a powerful incentive to use the computer as a computational tool.

Rival claims have been made about the effects of watching violent TV shows on children. Some believe that the shows stir up aggressive feelings while others suggest the opposite -- that children get rid of their aggressions vicariously by watching TV. A child psychologist decides to test these two hypotheses. He selects as his measure of aggression a projective type of test which consists of a set of pictures of boys in ambiguous situations. For each picture, the subject is to pick one of five alternatives, one of which has an aggressive theme, about what is going on. The test is administered to a group of ten-year-old boys. Immediately after finishing the test, the boys are shown a sixty-minute Western movie which depicts a good deal of violence--barroom brawls, gunfights, etc. After the movie, the boys are given the projective test again. (In previous research, it has been found that children given the test before and after seeing a movie with no aggressive content achieve comparable scores on the two occasions.) The difference between the two test scores is determined for each boy.

On the basis of the fictitious data above, determine whether the violent movie had any influence. The "X" scores are for the initial testing. The "Y" scores are from the test presented after the movie.

APPENDIX D

COMPUTER ASSISTED TEACHING AND LEARNING APPLICATIONS
by Fred S. Halley

As indicated earlier in this paper, computers may be applied in various ways to meet instructional goals. It is the purpose of this appendix to provide a brief description to two such applications.

The first application uses several techniques to produce instructional and testing materials for students in a large (up to 417), required, self-instructional basic statistics course. Each student uses a study guide, a programmed text, a set of homework problems and a computer generated individualized data set. All the students have the same homework problems, but each student has a unique computer generated data set. In this manner, students may maximize cooperation and help each other with their homework, but simple plagiarism is prevented. The computer is used to produce the correct answers for each student's homework so that laboratory assistants may evaluate student progress. Also the computer is used to generate both comprehensive and computational final examinations. (A more detailed description of this course may be found in Halley, 1972.)

A second application of the computer uses a program package in a sociology research methods course. In this course, students formulate a theoretical problem, develop operational definitions for its major concepts, construct measurement devices and collect data. After data collection, the students analyze the data and write a research report. Given the amount of work in this course for students, it would be unreasonable to require the students to learn a programming language for the analysis of their data. At the same time, it is desirable for the students to experience using a computer. To achieve this goal, students are provided with a program package which is easy to set up and execute without prior knowledge of computers or programming languages. In this manner, students may experience the use of a computer in the analysis of their own data without having to become programmers. (The program package used in this course, SOLIB, is described in Halley, 1973.)