The development of an evaluation plan for microcomputer courseware.

Robert Charles. Pearson

University of Windsor

Follow this and additional works at: https://scholar.uwindsor.ca/etd

Recommended Citation
https://scholar.uwindsor.ca/etd/3831
NOTICE
The quality of this microfiche is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Previously copyrighted materials (journal articles, published tests, etc.) are not filmed.

Reproduction in full or in part of this film is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30.

THIS DISSERTATION HAS BEEN MICROFILMED EXACTLY AS RECEIVED

AVIS.
La qualité de cette microfiche dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

Les documents qui font déjà l'objet d'un droit d'auteur (articles de revue, examens publiés, etc.) ne sont pas microfilmés.

La reproduction, même partielle, de ce microfilm est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30.

LA THÈSE A ÉTÉ MICROFILMÉE TELLE QUE NOUS L'AVONS REÇUE
THE DEVELOPMENT OF AN EVALUATION PLAN FOR MICROCOMPUTER COURSEWARE

by

Robert Charles Pearson

A Thesis submitted to the Faculty of Graduate Studies and Research through the Department of Communication Studies in Partial Fulfillment of the requirements for the Degree of Master of Arts at the University of Windsor

Windsor, Ontario, Canada

© 1986
Permission has been granted to the National Library of Canada to microfilm this thesis and to lend or sell copies of the film. The author (copyright owner) has reserved other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without his/her written permission.

L'authorisation a été accordée à la Bibliothèque nationale du Canada de microfilmer cette thèse et de prêter ou de vendre des exemplaires du film. L'auteur (titulaire du droit d'auteur) se réserve les autres droits de publication; ni la thèse ni de longs extraits de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation écrite.

I hereby declare that I am the sole author of this thesis.

I authorize the University of Windsor to lend this thesis to other institutions or individuals for the purpose of scholarly research.

Robert Charles Pearson

I further authorize the University of Windsor to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

Robert Charles Pearson
The University of Windsor requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.
THE DEVELOPMENT OF AN EVALUATION PLAN FOR MICROCOMPUTER COURSEWARE

by

Robert Charles Pearson

ABSTRACT

The rapid growth in the use and availability of microcomputer courseware has created a need for reliable procedures whereby users may assess its quality. The researcher developed and implemented an evaluation plan for microcomputer courseware. The plan was derived from a review of evaluation models and instructional theory. The evaluation models suggested approaches to courseware evaluation while the instructional theory suggested design prescriptions that could be used as evaluation criteria. The evaluation took place in two stages. First, a critique was undertaken to assess the courseware's adherence to sound principles of instructional design. In the second phase a quasi-experimental design was used to assess a number of learner outcomes that the evaluation plan deemed important. In addition to the development of an evaluation plan, the study demonstrated the importance of prescriptive theory in evaluating courseware and the shortcomings of a representative CAI course.
TABLE OF CONTENTS

ABSTRACT .......................................................... iv
LIST OF TABLES ..................................................... viii
LIST OF FIGURES .................................................. ix
LIST OF APPENDICES ............................................... x

CHAPTER

I. STATEMENT OF THE PROBLEM .............................. 11
   Introduction ............................................... 12
   Rationale for Study .................................... 13
   Definition of Terms ................................... 15
   Goal Statement ......................................... 16
   Overview of Pilot Study ................................ 17

II. REVIEW OF LITERATURE .................................. 20
   Evaluation Models and Methods ....................... 21
      Definitions ........................................... 21
      Evaluation Models ................................ 22
      CIPP Model .......................................... 23
      Discrepancy Evaluation Model .................... 24
      Goal Free Evaluation Model ....................... 26
      Stake's Evaluation Model ......................... 27
   Evaluation Methods .................................. 30
      Examinations ....................................... 30
      Attitude Questionnaire ............................ 31
      Observation ......................................... 31
      Archival Methods ................................ 32
      Interview ........................................... 32
      Checklist ............................................ 33
      Examples of Studies ................................ 34
      Summary .............................................. 38
   Instructional Theory .................................. 41
      Component Display Theory ......................... 43
      Performance Categories ............................ 43
      Content Categories ................................ 45
      Primary Presentation Forms ....................... 47
      Secondary Presentation Forms .................... 48
   Prescriptions ........................................... 51
      Performance-Primary Presentation Form Consistency .................. 51
      Content-Primary Presentation Form Consistency .................. 52
<table>
<thead>
<tr>
<th>V. DISCUSSION</th>
<th>130</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objectives of Evaluation</td>
<td>132</td>
</tr>
<tr>
<td>The Evaluation Method</td>
<td>132</td>
</tr>
<tr>
<td>Major Findings</td>
<td>134</td>
</tr>
<tr>
<td>Description of Sample</td>
<td>134</td>
</tr>
<tr>
<td>Student Performance</td>
<td>135</td>
</tr>
<tr>
<td>Student Attitudes</td>
<td>139</td>
</tr>
<tr>
<td>Limitations of Pilot Study</td>
<td>139</td>
</tr>
<tr>
<td>Conclusions and Implications</td>
<td>141</td>
</tr>
<tr>
<td>Suggestions for Future Research</td>
<td>145</td>
</tr>
</tbody>
</table>

APPENDIX...................................... 146

REFERENCES..................................... 162

VITA AUCTORIS.................................. 166
<table>
<thead>
<tr>
<th></th>
<th>LIST OF APPENDICES</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Demographics Questionnaire</td>
<td>146</td>
</tr>
<tr>
<td>2</td>
<td>Attitude Towards Computers (Pre)</td>
<td>147</td>
</tr>
<tr>
<td>3</td>
<td>Pretest</td>
<td>150</td>
</tr>
<tr>
<td>4</td>
<td>Posttest</td>
<td>151</td>
</tr>
<tr>
<td>5</td>
<td>Instructions for Performance Tests</td>
<td>152</td>
</tr>
<tr>
<td>6</td>
<td>Performance Tests</td>
<td>153</td>
</tr>
<tr>
<td>7</td>
<td>Attitude Towards Computers (Post)</td>
<td>157</td>
</tr>
<tr>
<td>8</td>
<td>Attitude Towards Instruction</td>
<td>160</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

1. Composite Evaluation Plan ........................................ 40
2. Performance-content Matrix ........................................ 47
3. Parts and Kinds Conceptual Structures ......................... 60
4. Procedural Organizing Structure ................................ 61
5. Theoretical Organizing Structure ................................ 62
6. Design Prescriptions ............................................... 72
7. Summary of Program Content .................................... 86
LIST OF TABLES

1. Description of Sample............................ 110
2. Source of Variation in Baseline Battery........ 112
3. Student Achievement on Pretest.................. 115
4. Baseline Battery................................ 117
5. Student Achievement on Posttest................ 122
6. Performance Test One.................................. 123
7. Performance Test Two................................ 123
8. Performance Test Three............................. 125
9. Performance Test Four............................... 126
10. Student Attitude Towards Computers (Post)....... 127
11. Student Attitude Towards Instruction............ 129
CHAPTER I

STATEMENT OF THE PROBLEM
Introduction

In this study, the researcher developed an evaluation plan for microcomputer courseware and used the plan to evaluate a representative piece of courseware.

Chapter I provides a rationale for the study, which includes a general statement of the problem, an indication of the problem's significance and a definition of the terms "computer-aided instruction" and "courseware". The chapter concludes with a brief description of the courseware used and the procedures followed in the evaluation.

Chapter II covers a review of the literature. The first section traces the evolution of evaluation models. In each case the strengths and weaknesses of these models are given. Evaluation methodologies appropriate to each model will also be criticised. A review of computer-aided instruction evaluation research follow and finally, a plan for evaluating courseware is proposed. The second section outlines
theories of instruction that are pertinent to the
design of computer-aided instruction. These theories
are prescriptive rather than descriptive in nature and
suggest a number of instructional design principles
that may be used as evaluation criteria.

The evaluation method is outlined in Chapter III.
Chapter IV presents the results of the evaluation which
includes a critique of the courseware and data from a
quasi-experiment. A discussion of the results and
conclusions are presented in the last chapter.

Rationale for Study

Since the introduction of the microcomputer less
than a decade ago, computer hardware and software have
proliferated at a rapid rate. Advances in technology
and production methods have allowed manufacturers to
market sophisticated computers at relatively low
prices. Educational institutions, in particular, have
rushed to purchase such equipment for their classrooms.
As the demand for computer skills continues to grow so
will the demand for hardware and software by schools,
colleges and universities.
The microcomputer has also brought about the rebirth of computer-aided instruction (CAI) (Tenner, 1984). Developed in the late 50's and 1960's, CAI held great promise, but high costs and technical problems all but ended research and development by 1970. The recent accessibility of microcomputers has suddenly created new demand for CAI courseware—software that delivers instruction. By the end of 1983 it was estimated that approximately 100,000 micros were in use throughout the U.S. university system. Some estimate that between 300,000 and 650,000 units will be in place by 1986 (Tenner, 1984). At the University of windsor, 45 IBM personal computers were recently installed in a specially designed microcomputer lab. The use of micros by schools is even more pronounced. Of the 82,420 schools in the United States, 24,640 (or about 30%) were using microcomputers in 1983. This was a 56% increase over the year before.

As the amount of courseware increases so does the need for reliable evaluation methods (Hickey, 1974). Educators and training personnel in industry have been flooded with thousands of CAI courseware products
produced by many companies. Much of what is available is of inferior quality (Wager, 1982a). Courseware is often marketed on the basis of visual appeal rather than instructional effectiveness (Wager, 1982b). Bright, glossy ads promote high resolution graphics, animation, color and sound. "Hence poor but 'fancy' programs many get more respect than effective but 'plain' programs," (Wager, p. 2). Diem (1982), stresses that because so much software is commercially available to the "neophyte" computer user, effective evaluation schema must be formulated.

Definition of Terms

Ragsdale (1982a) defines computer-aided instruction (CAI) as instruction that is delivered via computer. Computer-aided learning (CAL) and computer based learning (CBL) are terms used synonymously with CAI. Courseware is a term used to refer to instructional software. To further complicate this nomenclature problem, the term computer-managed instruction (CMI) is often confused with computer-aided instruction. In CMI, the computer serves only as a
diagnostic and prescriptive tool, which directs students to appropriate modes or units of instruction.

The application of computer-aided instruction can in turn be broken down. Zemke (1984) divides CAI into four categories: drill and practice, tutorial, simulations and games. The most common is drill and practice. These programs place the emphasis on review rather than instruction. A body of knowledge is presented by a teacher and then the drill and practice courseware is used to build student competency.

Tutorial programs are the second type described by Zemke. These programs present a body of information and may incorporate some form of drill. The third type, simulations, attempt to recreate some aspect of the "real world" through the computer. Simulations allow students to experience situations that otherwise would be too expensive, too time consuming or too dangerous to encounter in reality. The last type of program is a game which may be a form of drill and practice, a simulation or a tutorial. Games are distinguished from these other programs by the addition of a competitive element such as a running score of correct answers.
Goal Statement

The general objective of the present study was to develop an evaluation plan and to use it to assess a piece of courseware which is representative of products currently available. Specifically, the objectives of the study were:

1) Develop a plan to evaluate microcomputer courseware.
2) Generate a list of design criteria pertinent to microcomputer courseware.
3) Develop and execute a pilot study, based on the evaluation plan, to critically assess a piece of representative courseware.
4) Suggest ways to empirically test the validity of the evaluation plan.

The evaluation literature suggests a number of variables for investigation: student achievement, attitudes about personal computers and attitudes about instruction. The design criteria consist of prescriptions derived from two instructional theories and a model of motivational design. One instructional theory prescribes how course material should be
selected and sequenced, the other prescribes how single concepts should be presented within a lesson. The model of motivational design suggests how instruction should be structured to help motivate a student.

Overview of the Pilot Study

A commercially available computer literacy CAI training program was selected for evaluation. The courseware: "Teach Yourself PC DOS" (A.T.I., 1984), trains users to operate D.O.S., the disk operating system of the IBM PC. The evaluation took place in two stages. First, a critique was undertaken to assess the courseware's adherence to sound principles of instructional design. The second phase employed a quasi-experimental design to assess a number of learner outcomes that the evaluation plan deemed important. Two third year communication studies classes took part. The treatment group worked through the courseware during a regularly scheduled class. The treatment group also completed an attitude questionnaire about computers and a pretest prior to the treatment. A posttest, a performance test, a second attitude questionnaire about computers and an attitude
questionnaire about the quality of the courseware were administered after the treatment. The control group completed the computer attitude questionnaires, the pretest and the posttest.
CHAPTER II

REVIEW OF LITERATURE
Evaluation Models and Methods

Instructional evaluation has been defined and operationalized in many ways since the beginning of the century. This chapter briefly describes a number of representative evaluation models and methods. In each case, the strengths and weaknesses of these models and their appropriate methodologies are given. Finally, a composite model for evaluating computer literacy courseware is suggested.

Definitions

Four major trends in evaluation may be identified: evaluation as measurement; evaluation as professional judgement, evaluation as information gathering; and formative evaluation. During the early part of the century, evaluation was viewed strictly as measurement—formal tests (achievement and IQ) assessed congruence between a set of stated objectives and observed outcomes (Tyler, 1949). Over the last twenty years however, evaluators have stressed a need for evaluation to actively describe and judge a product, person or process (Scriven, 1971 & Stake, 1967).
Stufflebeam (1971) offers a third perspective by questioning whether evaluators should actually determine the "worth" of something and advocating that the evaluation process should simply provide information for decision-makers. Scriven's distinction between formative and summative evaluation is a fourth trend in evaluation. Scriven describes summative evaluation as occurring after completion of a product or process and for the benefit of some external audience or decision-maker (Scriven, 1981). Scriven's notion of summative evaluation is very similar to Stufflebeam's perspective. Formative evaluation is used during the development of a product and provides information that may change the design. Pilot testing is an example of formative evaluation.

Evaluation Models

A number of evaluation models—a more specific set of procedures for carrying out an evaluation which illustrate the various evaluation perspectives may be described. Despite the existence of many models, Borich and Jemelka (1976) identify four models that reflect the general definitions of the evaluation
process. These models include: the CIPP Evaluation Model developed by Stufflebeam et al., (1971); Provus' Discrepancy Evaluation Model (1971); Scriven's Goal-Free Evaluation Model (1974); and Stake's evaluation model (1967).

CIPP Evaluation Model

The CIPP evaluation model developed by Stufflebeam et al. (1971) divides the evaluation process into four stages: context evaluation, input evaluation, process evaluation and product evaluation. This renders the acronym CIPP. The CIPP model is concerned with ensuring that observed outcomes match program objectives at various stages in the development process. In this regard, Stufflebeam's model adheres to Tyler's general definition of evaluation. The CIPP model, unlike Tyler's narrow notion of congruence, assessed both the process and finished product. This is a major strength of the model. To use Scriven's terminology, the CIPP model has a strong formative aspect to it.
Stufflebeam's reliance on program objectives as a means of determining program success offers several advantages. Objectives provide a meaningful context for evaluators. They clearly identify what the program developers feel are important goals. This is consistent with Stufflebeam's feeling that evaluators should not judge the overall worth of a program but rather, provide information for decision-makers.

The reliance on objectives as an evaluation standard is however, viewed as a major drawback by many evaluators. The most obvious problem is that the CIPP model ignores learning not anticipated in the program objectives. Problems also arise if objectives are poorly stated or perhaps change during the development process.

**Discrepancy Evaluation Model**

The discrepancy evaluation model developed by Provus (1971) assesses a program against a predetermined standard of excellence. Standards are set jointly by program developers and evaluators. This comparison reveals "discrepancies" between outcomes and
a predetermined standard. Provus divides the process into five stages which include program description; field observations; analysis of student behavior both during and after instruction; and finally a comparison between the program and an alternative. At each of these stages evaluators and project developers assess program outcomes against standards of excellence (Provus, 1971).

Provus' model is similar to the CIPP model. Both have a strong formative element. The first three steps in the discrepancy evaluation process occur during program development. The major difference between the models is that Provus substitutes "standards" for "objectives". This approach makes the evaluation process more flexible. Programs may be assessed in ways other than observing the learner's terminal behavior. For example, a program may be assessed using a set of design criteria.

The discrepancy evaluation model, like the CIPP model, does not attempt to assess incidental learning caused by a particular program. This remains a serious limitation of both evaluation models.
Goal Free Evaluation

Michael Scriven describes an evaluation model intended to overcome the deficiencies of goal-oriented evaluation (Scriven, 1972). Goal-free evaluation is concerned with assessing the learning that occurs in addition to that which is intended. Scriven notes that an evaluator's bias, as cast by the program objectives, hinders an objective evaluation. Scriven mentions the blind and double-blind tests used by the pharmaceutical industry as an analogy. In a blind test, some patients are administered a placebo to control for the imagined feeling of well-being sometimes associated with drug taking. It was found that researchers were predisposed not to find effects in the control group. To counteract this bias, double blind tests, where both the observers and the patients are unaware which group is the control group and which is the experimental group, were initiated (Scriven, 1981) to fully counteract experimenter bias.

Scriven lists a number of advantages to goal-free evaluation. The most important is that it examines all the learning which takes place. Scriven advocates that
evaluators as in a double-blind test, be unaware of a program's learning objectives (Scriven, 1981). Goal-free evaluation also avoids the problems created when objectives are not clearly stated or perhaps change during the development process. In addition, Scriven warns that objectives often reflect what developers want rather than what is really needed (Scriven, 1981).

Goal-free evaluation does provide methodological problems for the evaluator. Unintended effects are more difficult to detect and measure. Stufflebeam (1971b) criticizes the model because he views the analysis of objectives and outcomes as an integral part of any evaluation. Stufflebeam, nevertheless, agrees with the importance of measuring unintended effects but feels it would be better subsumed under a more comprehensive evaluation model.

Stake's Evaluation Model

Stake believes any evaluation must fulfill two major functions: it should describe and it should judge (Stake, 1967). To undertake these two functions, Stake
identifies three categories in which evaluation information should be gathered: antecedent data, transaction data, and outcome data.

An antecedent is any "condition existing prior to teaching and learning which may relate to outcomes" (Stake, 1967, pg 96). Antecedents include such items as student aptitude, previous experience, interest and socio-economic status. Transactions describe the "encounters" between students and the instruction while outcomes refer to changes in student achievement, attitudes and aspirations that result from the instruction.

Stake places these three categories within two data matrices: a description matrix and a judgement matrix. The description matrix notes differences between intents and observations, while the judgement matrix notes standards and judgements.

The evaluator processes the description data in two ways: by analyzing the relationships or "contingencies" between antecedents, transactions and outcomes and the degree of congruence between intents
and observations. Stake emphasizes that discerning
congruence "does not indicate that the outcomes are
reliable or valid but that what was intended did
occur", (Stake, p. 99). Determining the relationship
between antecedents, transactions and outcomes helps
point out where improvements may be made in the
instruction: "the evaluator's task is one of
identifying outcomes that are contingent upon
particular antecedent conditions and instructional
transactions", (Stake, p. 99).

Stake describes two ways to judge a program:
judging with respect to an absolute standard and with
respect to the characteristics of a comparable program.
Judgements made relative to a standard of excellence
are more useful since they point out specific program
deficiencies rather than the superiority of one program
characteristic over another.

Stake's model offers many advantages. It allows
evaluators to compare program intents with program
outcomes—an important aspect of any evaluation. In
addition, student antecedents such as attitude and
level of motivation can be measured before and after
the instruction. This is similar to goal-free evaluation. Finally, Stake's model emphasizes the use of pre-established standards as a means of determining program worth.

Evaluation Methods

In addition to evaluation models a number of evaluation methods may also be described. Models provide an evaluator with a perspective from which to work. Evaluation methods prescribe procedures for carrying out an evaluation. The design of an effective evaluation therefore, entails careful consideration of what the study is trying to evaluate (model) and how the study will be executed (method). Hazen (1980) outlines five evaluation methods and discusses their relative advantages and disadvantages.

Examinations

The most widespread student evaluation method is the examination. Objective final examinations and achievement tests are useful because they purportedly measure accurately whether performance objectives are met. Tests are also easy to administer and easy to
grade. Examinations, however, cannot assess all the learning that goes on during a course since they are geared to predetermined standards (Hazen, 1980).

**Attitude Questionnaires**

Student attitudes and opinions may be estimated through the use of questionnaires. Opinions and attitudes are measured with semantic differentials, paired comparisons, multi-dimensional scaling or Q-sorts. The most common scale used on evaluation questionnaires is the Likert scale. Questionnaires are the most inexpensive means of gathering data on a large number of people. The major disadvantage in using questionnaires is that they are an obtrusive form of measurement (Hazen, 1980).

**Observation**

The unobtrusive nature of observational techniques is perhaps this method's greatest advantage. Researchers note facial expressions, eye movement and posture as clues in determining attitudinal responses to instruction. The major disadvantage associated with naturalistic methods is that evaluators might spend a
great deal of time observing without any interpretable data. In addition, it is difficult to eliminate observer bias completely (Hazen, 1980).

Archival Methods

Archival methods are an easy way to gather data. Student attrition rates provide a simple means of assessing student interest. Another means of measuring interest is to monitor how much time each student spent working. Aside from being easy to perform, archival methods are also an unobtrusive form of data collection. They provide a relatively easy means of performing long term studies. The major disadvantage of this method is the relative inaccuracy of interpretation available for the data collected. For example, excessive time spent working with a computer may be the result of a system malfunction, a student's lack of knowledge or lack of interest (Hazen, 1980).

Interview

The open-ended, unstructured interview is another useful evaluation method. Student feedback, a form of interview, is used extensively by developers as a means
of improving and updating instruction. Unstructured interviews, however, require skilled interviewers and often interview data cannot be easily analyzed (Hazen, 1980).

Checklists

Another popular means of assessing courseware is through the use of checklists. A checklist summarizes a number of design criteria. Evaluators note discrepancies between the design of courseware and the principles listed in the checklist. Two of the most popular checklists are MicroSIFT (Microcomputer Software and Information For Teachers) published by the (American) National Institute of Education and an evaluation guide prepared by The (American) National Council for Teachers of Mathematics (NCTM) (Ragsdale, 1982).

Roblyer (1982a) discusses the merits of using checklists to perform CAI evaluations, but she points out that most criteria developed through reviews of existing courseware are arbitrary. Roblyer (1981b) attempts to resolve this problem by suggesting that
some criteria are necessary to all good instruction, regardless of the medium used. Three categories of courseware criteria are given: essential characteristics, differential characteristics and aesthetic characteristics.

Essential characteristics are those criteria which are derived from research on how students learn. Wager (1982) and Briggs (1982) also stress the need for a theory base in courseware evaluation. Differential characteristics are criteria important within the context of a specific learning situation. Of concern here would be factors that are dictated by the learning style of the students. Roblyer suggests that aesthetic characteristics such as the use of color, animation and sound are the least important.

Examples of Studies

Many research studies have been undertaken in an effort to assess a particular piece of courseware. Gross and Griffen (1982) evaluated a computer-aided course in musical aural skills. The study examined the effects of the courseware on three learner related
variables: subject matter achievement, attitudes about the CAI course, and attitudes about computers in general. Gross and Griffen administered pre and posttests to a first year, ear training class at the University of Minnesota. The tests measured students' knowledge of the material covered in the course. In addition, students completed attitude surveys at the beginning and end of the course. Statistical tests were used to determine whether posttest scores were significantly different from pretest scores. The attitude questionnaire about computers was reviewed in a similar fashion. The post-treatment attitude questionnaire measuring students' opinion about the CAI music course was reviewed for positive and negative comments. Gross and Griffen determined that the computer-aided courseware was a success.

The major problem with the Gross and Griffen design is that they did not have a control group against which to compare the test and attitude data from the treatment group. This design makes it difficult to determine whether other factors, in addition to the courseware, caused the changes in
attitude and achievement. Gross and Griffen also neglected to evaluate the courseware with respect to accepted design criteria. This step is important since the criteria provide a rationale for problems highlighted by achievement and attitude measures.

Anderson et al. (1981) examined the effects of a 20 to 30 minute CAI science lesson on the attitudes and performance of high school students. 340 high schools students were randomly assigned to three treatment groups and a control group. Each treatment group received different versions of the course: a module with an enriched display (colour and graphics); a module with a planned malfunction and a module without an enriched display.

Anderson demonstrated that both affective and cognitive benefits resulted from exposure to the CAI course. Students in all treatment groups demonstrated significant knowledge gains even after six months. The computer learning situation left students with more favourable feelings about computers. Students also appeared to be more comfortable with computers after the course. Scores on the anxiety scales were
significantly lower immediately after the treatment and six months later. Both the enriched display and computer malfunction had no significant effect either on performance or attitude.

The Anderson study does an effective job of assessing the impact of computers on student attitudes. Unfortunately the researchers dwell on what Roblyer refers to as aesthetic characteristics, which are the least important to effective courseware design (as the results demonstrated). Like the Gross and Griffen study, Anderson neglected to assess the program in terms of accepted instructional design principles.

A study conducted by Sakamoto et al. (1979) attempted to develop a systematic method for evaluating CAI materials. Five different courseware products, all produced by independent authors, were evaluated using a number of techniques. A total of 106 students took part in the study. These students were randomly assigned to six treatment groups—one for each product and a control. The evaluation procedure included a thorough review and critique of each product by a panel of five experts. Immediately before each CAI exercise, students
were required to complete both a baseline test, a pretest and a motivational questionnaire. During the CAI exercise questionnaires designed to measure the students' level of anxiety and attitudes about computers were administered. Immediately following each CAI exercise, students completed a posttest, a questionnaire designed to assess feelings about the instruction, a second motivational questionnaire and a second attitude questionnaire.

Sakamoto's study is the most thorough of the evaluations reviewed. All of the important evaluation criteria—performance, attitudes, motivation and critique are included.

**Summary**

Borich and Jemelka (1976) stress that no single model is appropriate for a specific evaluation problem. Evaluators must review a number of models, selecting those attributes that suit their needs:

The problem of choosing the correct evaluation model derives from a somewhat natural tendency to see an evaluation model as more than it is; a methodology
for conducting the evaluation; instead of a meta-methodology or framework into which more specific constructs and methods must be plugged (Borich and Jemelka 1981, p. 177).

The review of evaluation models suggests three valid approaches to the process. Stake's model is the most flexible since it integrates each approach: professional judgement, measurement of student outcomes and assessment of incidental learning. The CIPP model emphasizes the degree of congruence between intended outcomes and observed outcomes. This type of evaluation quickly diagnoses problems in a design. The discrepancy model advocates the use of standards of excellence as a means of judging worth. Discrepancy evaluation helps provide answers for the problems uncovered during the first phase of evaluation. Finally, Scriven's goal-free model stresses the need for evaluators to carefully assess the side-effects of any instruction.

Each of these evaluation perspectives implies specific methodologies. An analysis of intended versus observed outcomes is best achieved through the use of
achievement tests. Judging the merit of courseware against pre-established standards may be accomplished by carefully describing every major element of the program. As suggested by Roblyer, a set of learning principles derived from research would be used as the standard against which the courseware would be judged. Finally, incidental learning may be assessed by attitude questionnaires, learning style inventories, interviews, observations and archival techniques.

**Figure 1: Composite Evaluation Plan**

<table>
<thead>
<tr>
<th>PHASE ONE</th>
<th>FUNCTION</th>
<th>METHOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assess congruence between intended outcomes and observed outcomes</td>
<td>To diagnose problems</td>
<td>Performance tests</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PHASE TWO</th>
<th>FUNCTION</th>
<th>METHOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Judge the courseware relative a standard</td>
<td>To help explain problems found in phase one</td>
<td>Critique Checklist</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PHASE THREE</th>
<th>FUNCTION</th>
<th>METHOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assess side-effects</td>
<td>To identify learning that</td>
<td>Attitude questionnaires</td>
</tr>
</tbody>
</table>
Instructional Theory

This section will examine theories and models that provide guidance for structuring and sequencing instruction. It is important to note that these theories are prescriptive rather than descriptive in nature (Reigeluth, 1983). Descriptive theories of instruction explain how students learn while prescriptive theories of instruction suggest under what conditions specific instructional strategies are most effective. Prescriptive theory is based on descriptive theory. For example, Ausubel (1962) suggests (descriptive theory) that knowledge is organized within the learner's memory primarily in a hierarchical fashion. General, more inclusive knowledge subsumes newer, more specific and more concrete knowledge. Reigeluth (1979) therefore suggests (prescriptive theory) that lessons should be sequenced in a simple to complex order with general, all-inclusive concepts being dealt with before more specific and concrete concepts.
Two instructional design theories will be presented: Merrill's Component Display Theory (1983) and Reigeluth's Elaboration Theory (1979). Two theories are necessary since at present no single theory is sufficiently comprehensive to include all instructional situations. In addition, Keller's (1983) model of motivational design will be used. The rationale for each theory is firmly grounded in various cognitive learning theories (Ausubel, 1963; Bruner, 1960; Gagne, 1965). The emphasis in this chapter will be on describing each instructional theory.

Elaboration Theory and Component Display Theory (CDT) may be distinguished in that Component Display Theory makes prescriptions at the "micro" level of instruction and Elaboration Theory makes prescriptions at the "macro" level (Reigeluth, 1983). "Micro" level theory prescribes strategies which are applicable only to the teaching of single concepts. For example, CDT would prescribe methods for teaching a third-grader how to classify animals as being mammals. "Macro" theories prescribe strategies applicable at the course level. Content selection and sequencing are typical
macro-level concerns. Elaboration theory would help a teacher determine whether the concept "mammal" should be taught before or after the concept "vertebrate".

Component Display Theory

Component Display Theory is objective-based. The theory allows designers to clearly conceptualize and define the nature of each objective and prescribe instructional methods and test items according to these definitions. Component Display Theory classifies objectives along two distinct dimensions: performance level and content type. Subsequent instruction is geared to these initial classifications.

Performance Categories

Performance level describes the complexity of a particular task. Asking a learner to recall the parts of a certain computer is an easier task than asking a student to explain how computers store information. The former simply requires recall while that latter requires an understanding of how computers work. Merrill proposes three performance levels: "remember", "use" and "find".
"Remember" level performance is the lowest level of student performance. A "remember" level task requires a student to recall some item of information that was previously memorized. This is a low level task since the student may simply recite the material verbatim without a meaningful understanding of it.

"Use" is the next level of performance. It requires the student to apply a rule to a specific case. For example, teaching a student how to balance a chemical equation is a "use" or "application" level task. "Use" level tasks are more difficult since the learner must apply rules to previously unencountered questions.

The "find" level is the most difficult. A student must actually discover, through observation, the rule or rules necessary in an "application" level task. Here are examples of test items at each performance level:

1) Recall the chemical formula for carbon dioxide.  
   (REMEMBER)

2) Identify the characteristics of a maple leaf that
help it receive more sunlight. (USE)

3) Invent a method to determine whether glucose exists within a plant. (FIND)

Content Categories

Content categories describe the kind of material that is being taught. Merrill classifies all content into one of four categories: facts, concepts, procedures and principles. A fact is an arbitrarily associated piece of information such as a proper name, a date, an event, the name of a place or a symbol used to name objects or events. The capital of Canada, or the year World War II ended are both facts.

A concept is a set of objects, events or symbols that share common characteristics and can be labelled with the same name. Insects, automobiles, and trees are all examples of concepts since the terms classify a set of objects that share clearly identifiable characteristics.

A procedure is an ordered set of steps that leads to a specific goal. Procedures may describe how to do something or how something works. Dividing whole
numbers and describing how the human heart functions are both procedural tasks.

A principle is an explanation or prediction about why things happen in the world. They typically involve some form of change relationship and in many of the sciences are stated as formal laws. The law of supply and demand is a principle that describes what happens to demand when there is change in supply (and vice versa). Here are examples of test items within each content classification:

1) What do the letters D.O.S. stand for? (FACT)

2) What characteristics distinguish micro computers from mainframe computers? (CONCEPT)

3) What are the steps in copying a file from one disk to another? (PROCEDURE)

4) Explain why a disk must be formatted. (PRINCIPLE)

Objectives and test items can therefore be described in terms of their performance level and content type. For example, a test item such as "classify the following computers as either micro, mini or mainframe, is clearly a concept question since it
requires the student to classify examples of computers as belonging to one of the three categories. If the examples are new to the student, then the task is at the "use" level since a rule must be applied in a new situation. If the examples are not new, then the task is at the "remember" level since the student must simply recall what kind of computer the example is.

**Figure 2: Performance-content Matrix (Merrill, 1983)**

```
\begin{array}{c|c|c|c|c|c}
\hline
\text{TASK} & \text{Find} & \text{Use} & \text{Remember} & \text{Fact} & \text{Concept} \\
\hline
\text{LEVEL} & \text{Concept} & \text{Proc.} & \text{Proc.} & \text{Proc.} & \text{Proc.} \\
\hline
\end{array}
```

**TYPES OF CONTENT**

**Primary Presentation Forms.**

Regardless of the kind (content type) or difficulty (performance level) of the task, instruction may be "presented" in four basic ways: generality form, instance form, expository form and inquisitory form. Merrill refers to these four forms as "primary presentation forms" (Merrill, 1983). For example, the
concept "computer" can be described at a general level by listing its critical characteristics (storage capacity, input device, output device, etc.) or it can be described at a particular level by presenting a specific example of a computer (an IBM PC). The general presentation form of instruction is a "generality" and the particular presentation form is an "instance". Therefore, a concept, procedure, and a principle may be presented as either a "generality" or an "instance". Facts, by definition, do not have generalities and can only be described in "instance" form.

Instruction may also be presented in "expository" and "inquisitory" form. These presentation forms relate to how the student will respond to the instruction. For example, a "generality" or an "instance" may be presented in "expository" form by telling or showing the student critical characteristics or examples. Instruction may also be presented in "inquisitory" form by asking the student to respond to a question which asks for a critical characteristic or an example. Most instruction uses both forms.
Expository generalities and examples are given first and then followed by "inquisitor" practice.

Secondary Presentation Forms

Secondary presentation forms provide additional help for the learner by elaborating upon primary forms. Secondary forms may give definitions of key terms in the generality that are unfamiliar to the learner or provide background information important to the lesson. Giving the learner the definition of "computer storage" and providing a brief history of the computer are examples of secondary presentation forms. Memory aids to help the learner remember a generality ("remember, D.O.S. rhymes with BOSS") and the use of arrows, colour, boldface type etc. to focus student attention on important aspects of the generality are other secondary presentation forms. The way generalities and examples are represented is yet another secondary presentation form. Generalities and examples may be represented in words or as diagrams, photographs or models.
When generalities and examples are given in "inquistory" form, the most important secondary presentation form is "feedback". When a student answers a question, "feedback" provides information about the nature of the student's response. Feedback may be as simple as stating: "yes you're correct" or as complex as a reworking of the problem complete with the correct answer and an indication of where the student made errors.

Merrill also describes two other secondary presentation forms. "Process displays" consist of instructions or directions given to the student that suggest how they should process the information that is presented. For example, an instruction like: "close your eyes and try to say the definition in your own words", is a process display. Procedural displays tell the student how to manipulate or operate the medium being used to present the instruction. Directions like "press return" or "turn the page now" are examples of procedural displays.
Prescriptions from Component Display Theory

Performance-Primary Presentation Form Consistency.
Component Display Theory outlines three taxonomies: a performance taxonomy, a content taxonomy and a presentation taxonomy. The prescriptive element of the theory proposes when and how the various components of each taxonomy should be used. The first set of propositions concern instructional consistency between "performance classifications" and the "primary presentation forms". It is suggested that, different kinds of instructional outcomes require different instructional treatments. Teaching a student to recall the provinces of Canada, and to solve a certain type of mathematical problem requires very different instructional strategies.

At the "remember level" only one generality or instance is presented. Practice and test items consist of inquisitorial presentations which require the student to recall the generality or instance. At the "use level" a generality is presented and then followed by examples of that generality. Practice and test items consist of a set of new, previously unencountered,
examples to which the learner responds. Instruction at the "find level" contains only practice and generalities from which the student must discover a generality.

**Content-Primary Presentation Form Consistency.** A second set of propositions prescribe how presentation forms should differ according to the type of content being taught. The generality for a concept is labelled by the term "definition". A definition consists of a concept name, the name of the class to which the concept belongs (superordinate class) and defining characteristics. An example of a concept definition is: "a dog (concept name) is an animal (subordinate class) with four legs, fur and a cold nose (defining characteristics)". An instance of a concept is labelled by the term "example". An "example" is a specific object, event, or symbol that is a member of the class under consideration. The "example" consists of a concept name and the object or a representation of the object in question. An example of the concept "dog" would be a picture of German Shepherd.
The generality for a procedure is labelled by the term "process" and is characterized by an ordering relationship. The generality identifies an outcome which is the result of a certain process, the order in which the process steps occur and a distinction between process steps and decision steps. Decision steps are used when there is more than one way to perform a procedure. An example of a procedural generality is: "to start a car (goal), press down on the accelerator and turn the ignition key (steps). If the car is cold, pump the accelerator (decision step)". An instance for a procedure is called a "demonstration" and consists of the name of the procedure being demonstrated, a statement of the desired goal, the materials and/or equipment to be used and an actual or simulated execution of the procedure. A procedural instance may be a film demonstrating the steps in starting a car.

The term "proposition" is used to label a generality for a principle. A "proposition" includes the name of the principle, an indication of the component concepts involved in the principle and the
nature of the causal relationship between the concepts. In mathematics and science, principles often take the form of formulas. The instance of a principle is called an explanation. An explanation should contain the name of the principle, a description of the problem situation and a description of how the concepts related by the principle interact and change over time.

**Practice.** When teaching a student to remember a fact or generality, practice should require the student to recall that fact or generality. For example, the student would be given a country and asked to give its capital. Students learning concepts practice classifying behaviour. The student is given an example and must indicate whether it is a member of the concept class in question. Students learning a procedure practice by demonstrating a particular process. The student is given the goal and name of the procedure and the materials or equipment necessary to carry out the demonstration. Students learning a principle practice by explaining what will happen in a given situation. At the "find" level, practice consists of "examples", "demonstrations" and
"explanations" and the student is required to invent or discover the appropriate "concept", "procedure" or "principle".

Secondary Presentation Forms. Secondary presentation forms will often be needed even if content type and performance level are consistent with primary presentation forms. Feedback should always accompany practice regardless of whether the answer is incorrect or correct. At the "remember" level, memory aids (mnemonic elaboration) are recommended for both generalities and facts. At the "use" level, attention focusing (arrows, coloured text, etc.) is recommended for both generalities and instances and alternate representations (pictures, diagrams, models, etc.) may be used for instances and generalities at either the "use" or "remember" levels.

The degree to which secondary presentation forms are used depends on the complexity of the instruction relative student ability. Merrill refers to this relationship as the instruction's "richness level" (Merrill, 1983). The more complex the content the richer the instruction and the greater the need for
secondary presentation forms. For example, a particularly bright student may only require the definition of a concept to be able to correctly classify examples. A slower student by contrast, might need the definition, examples, practice, attention focusing, and alternative representations. When combined with a management system, Component Display Theory can afford a measure of learner control for the student. Bright students can be "branched" around unnecessary primary and secondary presentation forms.

**Relationships between Primary and Secondary Presentation Forms.** Instruction should clearly differentiate between primary and secondary presentation forms. The nature of a generality and what is an elaboration of the generality must be clear to the student. Instances of generalities should be as different as possible and when teaching concepts matched non-examples (examples that have all but one critical characteristic) are recommended. Non-examples are not recommended for demonstrations since the student may learn the incorrect procedure. Non-examples should be used sparingly when teaching
principles. They may prove useful in pointing out incorrect interpretations or they may confuse the student. Caution is recommended when using non-examples in any case. Finally, instances should be ordered in a simple to complex sequence so that students can master less difficult items first and then progress to more difficult ones.

Elaboration Theory

An Elaboration Sequence

Unlike Component Display Theory, Elaboration Theory (Reigeluth, 1979) operates strictly at the macro level of instruction and does not deal with either delivery or management strategies. The theory does however, build from the conceptual structure of Component Display Theory. Elaboration Theory is concerned exclusively with the organization of instruction at the macro level. Organizational concerns are broken down into four general problem areas: selection, sequencing, synthesizing and summary of subject matter. In general terms, Elaboration Theory prescribes that instruction begin with a special kind
of overview called an "epitome" and the remaining course material be organized in a simple-to-complex sequence--subsequent lessons "elaborate" upon the material presented in the "epitome".

Elaboration Theory is comprised of strategy components: (1) an elaborative sequence; (2) learning prerequisite sequences; (3) summarizers; (4) synthesizers; (5) analogies and (6) a learner-control format.

An elaborative sequence is a special kind of simple-to-complex sequence. The first lesson of an elaborative sequence epitomizes the content which is to follow. An "epitome" is not a summary which presents many ideas at a superficial and abstract level. An "epitome" teaches the most fundamental and representative ideas of the course at an application ("use") level. The remaining lessons provide more complex knowledge about these fundamental ideas. For example, the epitome lesson for a course on mammals would provide a series of divergent examples of mammals (whales, polar bears and man). Subsequent lessons would deal with the various "types" of mammals.
The material within a course is organized with respect to one content type only. Thus, an elaborative sequence may be described as having either a conceptual organization, a procedural organization or a theoretical organization. The other two types of content and facts make up the supporting content of a course.

A conceptual organizing structure shows superordinate, coordinate and subordinate ideas among concepts. There are two basic types of conceptual structures. A "parts" structure shows concepts that are components of a given concept. A "kinds" structure shows concepts that are types of a given concept. Figure 3 is an example of a "parts" and a "kinds" conceptual organizing structure. Each box in the diagram may represent one lesson's worth of material.
Figure 3: Parts and Kinds Conceptual Structures
(Reigeluth, 1983)

A procedural organizing structure shows relationships among steps in a procedure. As with conceptual structures there are two types of procedural organizing structures. One is a procedural-order organizing structure which shows in what order(s) the
steps should be performed and a procedural-decision structure which shows when to use a specific series of steps. Flowcharts are often used to display procedural organizing structures (see figure 4).

**Figure 4:** Procedural Organizing Structure (Reigeluth, 1983)

A theoretical organizing structure shows the relationships between a series of events in the real world. The intent of this type of structure is often to describe the results of a specific change. For example, if the demand for cars goes down so does the
demand for steel and the demand for coal (see figure 5). Theoretical organizing structures may also be prescriptive in nature. Such a structure identifies desired outcomes and how to achieve them (see figure 5).

Figure 5: Theoretical Organizing Structure
Prerequisite Learning Structure

A prerequisite learning structure is a strategy component that shows what prerequisite skills and/or knowledge must be mastered before a given piece of content can be learned. For example, before a student can learn how to use computer software, the student must know how to operate the computer at some basic level. Prerequisite learning structures should exist for every step in an elaboration sequence.

Summarizers

A summarizer is a strategy component that systematically reviews material that has been learned. A summarizer should provide a concise statement of each idea and/or fact that has been taught. In addition, an example of each idea and some practice items should also be presented. There are two kinds of synthesizers: an "internal" synthesizer which occurs at the end of a lesson and summarizes material from that lesson only and a "within-set" synthesizer which summarizes material from a set of lessons. A set of
lessons includes a lesson and those lessons which elaborate upon it.

Synthesizers

A synthesizer, unlike a summarizer, interrelates and integrates content that has been taught. A synthesizer typically takes the form of a content organizing structure (see figures 3 thru 5), examples, and some practice items. Like summarizers, there are two kinds of synthesizers: "internal" and "contextual". Internal synthesizers relate ideas within lessons and contextual synthesizers relate ideas between lessons.

Analogy

An analogy is a strategy component that relates ideas presented within a lesson to ideas already familiar to the learner. The greater the similarities between the analogy and the idea in the lesson, the more successful the analogy will be. If the number of differences is great, the analogy may prove to be more confusing than helpful.
Learner Control

In general terms (Merrill, 1980), learner control refers to students' abilities to select and sequence instructional content at their own pace. In addition, students should be able to select appropriate strategy components and cognitive strategies. An elaborative instructional sequence allows for learner control over selection, sequencing of content and selection of strategy components and cognitive strategies. With respect to selection and sequencing of content, an elaborative sequence permits students to select an aspect of the epitome lesson that is of interest and then continue to seek more detail in that area. Clearly labelled strategy components (synthesizers, summarizers etc.) facilitates learner control by permitting the student to select a strategy component when appropriate.

Prescriptions from the General Elaboration Model

As stated earlier, elaboration theory prescribes one of three models depending upon the kind of content (concepts, procedures, principles) the course presents.
In addition, the theory prescribes how the various strategy components should be used with each model. There are prescriptions however, which are common to each model and can be subsumed under a general elaboration model. The most basic prescription of the general model is that instruction must begin with an "epitome". Typically this lesson would begin with an analogy and then be followed with organizing content. Each piece of organizing content would be immediately preceded by its learning prerequisites. After the organizing content had been presented, all relevant supporting content would follow and the lesson would end with an internal summarizer and a synthesizer.

After an "epitome" lesson, the general model prescribes a series of elaborations. Each elaboration is referred to as a new "level" of instruction. Reigeluth (1983) suggests that there may be anywhere from four-to-eight level-one lessons. These lessons provide more detailed information about a some aspect of the "epitome" lesson. Each level one lesson contains all the strategy components of the epitome lesson but with the addition of a contextual summarizer.
and a contextual synthesizer which reviews and integrates previous material. Level one elaborations in turn serve as epitome lessons for level two lessons—lessons which provide more detailed information about material presented in level one lessons. The general model continues to prescribe more detailed and complex lessons until the students have reached mastery on the course objectives.

Some variations must be made to the general model depending upon the type of organizing content. The "epitome" for a conceptually organized course contains the most general or inclusive concept. Subsequent elaborations present progressively narrower and more detailed sub-classifications of the broader concept. Conceptual synthesizers are unique in that they utilize a conceptual structure (see figure 3). A procedural organizing structure would begin with the simplest most applicable process. Keigeluth refers to such a process as the "shortest path" sequence. Procedural elaborations teach increasingly more detailed and situation-specific versions of the epitome process. Procedural synthesizers often take the form of a
procedural flow chart (see figure 4). Finally, theoretical content is organized around a fundamental principle. Elaborations present more complex and content-specific variations of the fundamental principle. Theoretical synthesizers often take the form of a series of cause-and-effect statements (see figure 5).

It should be noted that a course could be taught from any one of the three content perspectives—such a decision is based upon the goals of the course. Supporting content (for example: facts, concepts and principles in the case of procedural organizing content), must be integrated into the course along with appropriate strategy components.

The ARCS Model: Motivational Design

The two previous models present prescriptions for instructional design at the course and lesson level. The models however, lack motivation strategies. Keller (1983) proposes four basic categories (and a number of subcategories) of motivational conditions that must be addressed in all instruction. The four categories are attention, relevance, confidence and satisfaction.
Keller uses the term attention to mean curiosity arousal as defined by Maw and Maw (1968) and Berlyne (1965). Thus, Keller views the first step in motivating students as getting and maintaining their interest in the content. Subsumed under the broad category of attention are three strategy components. 1) Perceptual arousal: attention may be gained and maintained through the use of novel, surprising, incongruous or uncertain events during instruction. 2) Inquiry arousal: information-seeking behaviour may be stimulated by having students ask questions or solve problems related to the content. 3) Variability: the nature of the instruction must vary in order to maintain interest.

Once a student becomes curious about a certain content area, that student will begin to ask: "why is this important to study?" Content relevance is an important part of motivational design. Keller proposes three strategy components to help make instruction more relevant to the learner. 1) Familiarity: the use of language and examples that relate to a learner's own experiences. 2) Goal orientation: the instruction should clearly identify both objectives and course
utility for the student: 3) Motive matching: use teaching strategies that match the goals and aspirations of each student.

Even if instruction raises curiosity and is relevant, a student may not proceed if they believe there is a likelihood of low achievement. Every student must be sufficiently confident of success throughout the instructional process. Keller again proposes three strategy components to promote learner confidence. 1) Expectancy of Success: the students should be aware of how they are to be evaluated. 2) Challenging setting: each student should be able to select an appropriate performance criterion to ensure a challenging but not overly difficult learning environment. 3) Attribution molding: the student should be provided with feedback that supports ability and effort as the causes of success.

Students may become unmotivated if the outcomes of the instruction do not meet their expectations. Keller proposes three strategy components that will maintain student satisfaction. 1) Natural consequences: the student must have the opportunity to use the new skills
and knowledge in a meaningful context. 2) Positive consequences: the student must receive positive feedback whenever possible. 3) Equity: consistent evaluation criteria and rewards should be maintained for each student.

Summary

Merrill (1983), Reigeluth (1979) and Keller (1983) propose a number of instructional prescriptions that may be used as evaluation criteria. Merrill's theory suggests prescriptions at the "lesson" level while Reigeluth's theory suggests prescriptions at the "course" level. Keller's model prescribes motivational strategies:
Figure 6: Design Prescriptions

Lesson Level

1) Objectives, instruction and test items should be consistent with respect to content type (concepts, procedures and principles) and task level (remember, use and find).

2) A lesson taught at the application level should use generalities, examples, and practice (primary presentation forms).

3) A lesson taught at the application level should provide feedback, diagrams, memory aids, alternative representations and background information (secondary presentation forms) where appropriate.

Course Level

4) Course material should be selected according to the entry level of students.

5) Course material should be presented in an elaborative (simple to complex) sequence.

6) A course should begin with an "epitome" lesson.

7) Each lesson should contain summary and synthesis of previous material.

Motivational Design

8) Instruction should gain student attention.

9) Instruction should be relevant to students.

10) Instruction should build the confidence of students.

11) Instruction should be satisfying for students.
CHAPTER III

PROCEDURES
Target Population and Sample

The treatment group consisted of university students taking third year communication studies courses in a social science faculty. Students from another communication studies class acted as a control group. Both courses were offered during the 1985 summer session.

Description of Stimulus Material

The treatment group was required to work through a computer-based training program entitled "Teach Yourself PC DOS". The courseware was produced by American Training International (ATI, 1984). ATI markets approximately 60 training programs on many topics. 'Teach Yourself PC DOS' is designed for the 'computer novice'. The package comes with one 5 1/2 inch floppy diskette and a 60-page user's handbook.

The ATI courseware package is menu driven, i.e. users select topic areas from a list which is displayed at the beginning of the program. Although any unit can be selected at any time; the program recommends that
the student work through each unit in order. The program will not run unless DOS has been loaded into the computer first. The instruction handbook makes this point, but does not however, tell the user how to load DOS.

Once the ATI program is loaded, the user is asked whether a colour display is required. The user simply types a 'Y' or an 'N'. If another character is inadvertently typed, a red warning message flashes at the bottom left of the screen instructing the user to strike the proper key(s). This form of error correction is maintained throughout the course. Once a 'Y' or an 'N' is selected the program automatically advances to a series of four introductory frames (FRAMES 1 to 4). These frames describe what DOS is and tell users that the program is divided into a beginner and advanced section. Users advance from frame to frame at their own pace by pressing the space bar. This procedure is maintained throughout the course except where specific answers are requested. In addition to advancing the program frame by frame, users have the option of
returning to the main menu by pressing the escape key (ESC).

The main menu for the first section of the course in divided into 10 units:

1) How to use your training program efficiently.
2) Load PC DOS.
3) Display the directory.
4) Check the status of a disk.
5) Prepare a disk.
6) Copy a disk.
7) Copy a file.
8) Display the contents of a file.
9) Rename a file.
10) Erase a file.

The second part of the course covers:

1) How to use EDLIN (text editing program) to prepare sample letter.
2) Use EDLIN to modify file contents.
3) Combine command in batch files.
4) Create a tree system.
5) Work on files in a tree system.
6) Expand the tree system.
7) Manipulate files.
8) Use your printer and other devices.

Unit A: How to Use Your Training Program

This unit begins by reminding users that the ATI training program contains 2 components: the training disk and the user's handbook (FRAMES 6 to 8). The handbook, which summarizes the commands covered in the disk, does not contain any instruction and is intended only as a reference manual (FRAME 12).

Next, the program explains how material is presented throughout the course. ATI uses what it calls a "split screen" approach. The top half of the screen displays an actual example of software while the bottom half contains an explanation or instructions. In the colour version of the program, the top portion of the screen is displayed in light blue while the bottom is displayed in orange (FRAMES 9 to 11).
Unit B: Load PC DOS

The second unit of the course shows users how to load DOS into the computer and check the disk directory. The first frame (FRAME 16) stresses that DOS must be loaded every time you use the computer and tells users how to load the operating system—"just insert the disk into your computer and turn the system on".

The next three frames (FRAMES 17 to 19) describe the series of prompts the user receives after loading the program. The first prompt asks for the date and the next for the time. These prompts can be over-ridden by pressing the enter key. The third prompt is designated by a 'A>' symbol which is introduced in FRAME 20. The A> symbol tells the user that DOS has been loaded and that the computer is waiting for a command. The next frame introduces the cursor.
Unit C: Display the Directory

This unit begins by describing the concept of a file (FRAMES 23 to 25): "All the information on a disk is stored in files. A list of files on a disk is similar to the table of contents for a book" (FRAME 25). The next four frames (FRAMES 26 to 29) show the user how to list the names of files stored on a disk by using the DIR (directory) command. Frame 30 displays an example of a file and explains that the number following the file name measures the amount of space used by that file. Frame 31 shows the user how to find when the file was created or last updated. Frame 32 introduces users to the term 'external commands' which are simple DOS commands stored as files on the disk. Frame 33 explains that some DOS commands are automatically loaded into computer memory. These 'internal commands' are not listed as files and do not appear in the directory.

The next four frames (FRAMES 34 to 37) let the user practice listing the file directory on drive B. Frame 39 explains that three letters--called extension letters--often follow the file name in a directory and
are widely used to classify files and documents into categories. Frame 40 reminds the user that the \( A \Rightarrow \) prompt indicates that drive A is still the active drive. The last two frames (FRAMES 41 to 43) list the concepts covered in the unit.

Unit D: Check the Status of a Disk

The fourth unit of the program describes the check disk (CHKDSK) command (FRAME 43). The check disk command tells the user how space is used on a disk: how many files are stored on the disk and how much space is left on the disk (FRAME 44). Frame 45 introduces the concept of computer memory by briefly mentioning that the 'check disk' command shows how much storage space the files on the disk are using. Frame 46 allows the user to check the storage capacity of a disk by typing the CHKDSK command. Frame 47 explains the term byte by stating it is like a character such as a letter or number. Frame 48 points out that the CHKDSK command also gives the computer's total memory and frame 49 introduces the user to the term kilobyte (K): "one K is exactly 1024 bytes".
Frames 51 to 54 show how to determine whether a disk is in good working order by using the CHKDSK A (checkdisk) command or the CHKDSK B command. A faulty disk or an unformatted disk (this term is not explained until the next unit) would result in an error message. Frame 55 reviews the major points covered in the unit.

Unit E: Prepare a Disk

Frames 57 to 60 show how to use the format (FORMAT) command. Frame 58 explains that this command initializes a disk so it can be used to store information. Frame 61 stresses that formatting a disk erases all the information already stored on it and frames 62 to 66 allow the user to go through the process of formatting a disk by typing the FORMAT command and pressing the enter key.

The last section of the unit (FRAMES 67 to 73) describes how to copy DOS onto a formatted disk by typing a '/S' after the format command. Frame 69 reminds the user that formatting a disk will erase all the information stored on it. Frame 71 shows that the
computer will display a "system transferred message" when DOS is transferred to a formatted disk. It is stressed that this newly formatted disk can be used to start the computer in the future. Frame 73 shows the user how to format single sided disks by placing a '/1' after the FORMAT command. The unit does not explain what a single or a double disk drive is. Frame 79 reviews the main points covered in the unit.

Unit F: Copy a Disk

The next unit shows users how to make copies of a disk. Frame 76 suggests that the always prepare backup copies of a disk to guard against loss or damage. Frame 77 states that the DISKCOPY command can be used in a computer with either one or two disk drives. Frames 78 to 84 present a step by step explanation of how to copy a disk using two disk drives by typing DISKCOPY and pressing the enter key. Frame 84 states that the copying process with one disk drive is the same "except for some changing of disks".

Frame 85 recommends the user check the disk copy by comparing it with the original. The is done by
using the DISKCOMP command. Frames 86 to 90 show how to use the disk compare command by typing DISKCOMP and pressing the enter key. Frame 88 indicates that a "special message would appear if the disks were not the same. Frame 91 reviews the main points covered in the unit.

Unit G: Copy a File

Frame 94 stresses that two disk drives are needed to use the copy command fully. The ATI training program nevertheless does not require two drives. Frame 95 introduces an imaginary file called "History.doc" and shows how to make a backup copy by using the COPY command. Frame 97 emphasizes that the COPY command does not effect the original file. Frames 98 to 100 go through the copy process for a second time.

The last two frames of the unit (FRAMES 101 to 102) show the user how to copy a group of files from one disk to another by using "wild card" commands. The user simply types two asterisks after the group of files. Frame 103 summarizes the main points covered in the unit.
Unit H: Display the Contents of a file

This unit shows the user how to access and list a specific file. Frame 106 shows how to select the appropriate disk drive by typing the disk letter (A, B, C and so on) followed by a colon. Next, a file is listed by using the TYPE command followed by the file name (FRAME 108). The example used is TYPE CHRIS.MEM: Frame 111 shows what happens when a longer file is accessed. The file simply scrolls across the screen until it reaches the last line. Scrolling can be stopped at any point by holding the control key and pressing the NUMLOCK key. Scrolling is started again by pressing any key (FRAME 112). The unit does not allow the user to practice the scrolling controls. Frame 114 lists the main points covered in unit H.

Unit I: Rename a File

Unit I describes how a file many be renamed. The user is shown how to change the file name, 'History.doc' to 'Founders.doc' (FRAME 118). This is accomplished by using the REN command followed by the original file name and then the new file name. The REN
command is activated by pressing the enter key (FRAME 119). Frame 120 lets the user check the directory to see whether the file has been renamed. Frames 121 to 124 show how a number of files can be renamed simultaneously by using the wild card commands. The user types in REN *.bak *.x (FRAME 124) and then checks the directory by typing DIR and pressing the enter key to see that all the files originally listed as *.bak are now listed as *.x (Frame 123). Frame 125 is the review frame.

Unit J: Erase a File

The last unit of the first section shows how to erase a file. Frame 123 asks the user to erase a file named 'History.x' by typing ERASE HISTORY.x and pressing the enter key. In frame 126 the user checks the directory and finds that 'History.x' is not listed. The last three frames of the unit (FRAMES 127 to 129) show how to erase more than one file at a time. The user types ERASE followed by an asterisk and the appropriate extension letter or file names. In frame 129 the user checks the directory to see that the file was erased. Frame 130 is the review frame.
Figure 7: Summary of Courseware Content

<table>
<thead>
<tr>
<th>Unit</th>
<th>Frame</th>
<th>Unit</th>
<th>Frame</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) Introduction</td>
<td></td>
<td>5) Prepare a disk</td>
<td></td>
</tr>
<tr>
<td>a) What is DOS?</td>
<td>3</td>
<td>a) Formatting</td>
<td>59</td>
</tr>
<tr>
<td>b) Time &amp; date</td>
<td>18</td>
<td>b) FORMAT command</td>
<td>67</td>
</tr>
<tr>
<td>c) A prompt</td>
<td>20</td>
<td>c) Formatting (DOS)</td>
<td>67</td>
</tr>
<tr>
<td>d) The cursor</td>
<td>22</td>
<td>d) System transfer</td>
<td>71</td>
</tr>
<tr>
<td>2) How to load PC DOS</td>
<td></td>
<td>6) Copy a disk</td>
<td></td>
</tr>
<tr>
<td>a) Loading DOS</td>
<td>17</td>
<td>a) Why make copies?</td>
<td>76</td>
</tr>
<tr>
<td>b) Time &amp; date</td>
<td>18</td>
<td>b) DISKCOPY command</td>
<td>78</td>
</tr>
<tr>
<td>c) A prompt</td>
<td>20</td>
<td>c) With 1 drive</td>
<td>84</td>
</tr>
<tr>
<td>d) The cursor</td>
<td>22</td>
<td>d) DISKCOMP command</td>
<td>85</td>
</tr>
<tr>
<td>3) Directory</td>
<td></td>
<td>7) Copy a file</td>
<td></td>
</tr>
<tr>
<td>a) Files</td>
<td>24</td>
<td>a) COPY command</td>
<td>95</td>
</tr>
<tr>
<td>b) DIR command</td>
<td>27</td>
<td>b) Copying files</td>
<td>101</td>
</tr>
<tr>
<td>c) Directory layout</td>
<td>30</td>
<td>c) Wild cards</td>
<td>101</td>
</tr>
<tr>
<td>d) Space used by file</td>
<td>30</td>
<td></td>
<td></td>
</tr>
<tr>
<td>e) External commands</td>
<td>32</td>
<td></td>
<td></td>
</tr>
<tr>
<td>f) Internal commands</td>
<td>33</td>
<td></td>
<td></td>
</tr>
<tr>
<td>g) Disk drives</td>
<td>37</td>
<td></td>
<td></td>
</tr>
<tr>
<td>h) Extensions</td>
<td>38</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4) Check disk</td>
<td></td>
<td>8) Displaying a file</td>
<td></td>
</tr>
<tr>
<td>a) CHKDSK command</td>
<td>47</td>
<td>a) Changing drives</td>
<td>105</td>
</tr>
<tr>
<td>b) Byte</td>
<td></td>
<td>b) TYPE command</td>
<td>107</td>
</tr>
<tr>
<td>c) Kilobyte (K)</td>
<td>49</td>
<td>c) Scrolling</td>
<td>114</td>
</tr>
<tr>
<td>d) Fault error message</td>
<td>53</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5) Prepare a disk</td>
<td></td>
<td>9) Renaming a file</td>
<td></td>
</tr>
<tr>
<td>a) Formatting</td>
<td>59</td>
<td>a) REN command</td>
<td>121</td>
</tr>
<tr>
<td>b) FORMAT command</td>
<td>67</td>
<td>b) Erasing files</td>
<td>131</td>
</tr>
<tr>
<td>c) Formatting (DOS)</td>
<td>67</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d) System transfer</td>
<td>71</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6) Copy a disk</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>a) Why make copies?</td>
<td>76</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b) DISKCOPY command</td>
<td>78</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c) With 1 drive</td>
<td>84</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d) DISKCOMP command</td>
<td>85</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7) Copy a file</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>a) COPY command</td>
<td>95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b) Copying files</td>
<td>101</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c) Wild cards</td>
<td>101</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8) Displaying a file</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>a) Changing drives</td>
<td>105</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b) TYPE command</td>
<td>107</td>
<td></td>
<td></td>
</tr>
<tr>
<td>c) Scrolling</td>
<td>114</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9) Renaming a file</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>a) REN command</td>
<td>121</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b) Erasing files</td>
<td>131</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Entry Behaviour

Despite being described as a "training program for computer novices", the DOS courseware requires a number of entry skills. The user must be able to turn on the computer and load DOS. The courseware also requires a
basic knowledge of the IBM keyboard. The user must be able to find the "enter" key and the "escape" key.

Instruments

The study employed a number of instruments to measure the students' attitudes and performance. A baseline battery was used to gather data about the characteristics of the participants in the study. An attitude questionnaire and a knowledge pretest were administered to the treatment and control groups prior to the treatment. Four posttest measures were used: an attitude to instruction questionnaire, an attitude to computers questionnaire, a knowledge test and a performance test.

Baseline Battery

The baseline battery consisted of a six-item questionnaire (see appendix 1) that requested basic demographic information: sex, age, year of study, major and computer experience, a questionnaire about students' attitudes towards computers (see appendix 2) and a pretest about material in the courseware (see appendix 3). The attitude instrument consisted of a
questionnaire developed by Lewis (1985) and a questionnaire developed by the Minnesota, Educational Computing Consortium—MECC (Anderson, et al., 1979) (see appendix 2). Both the pretest and the demographic questionnaire were developed by the researcher.

Knowledge and Performance Tests

Initially, a list of knowledge and performance objectives were derived from the courseware. From these lists of objectives, three instruments were constructed. The knowledge pretest and posttest (see appendix 3 and 4) contained short answer questions that determined whether students knew and/or retained the concepts presented in the courseware. The pretest and posttest required students to define certain terms presented in the course such as a computer file, wild card commands, abort command, internal commands and external commands. In addition to defining these terms, students were required to identify computer commands and procedures. Answers were scored as either right or wrong and a point was scored for each correct answer. These tests were administered to both the control group and the experimental group one week
before the treatment. The experimental group completed the posttest after finishing the first section (units A thru J) of the courseware while the control group completed the posttest during a class meeting earlier on the same day.

Since the courseware was designed primarily to build specific computer skills, the experimental group was also required to do a performance test (see appendix 5 and 6). The students were informed that after they completed three units in the course they were to move to one of three testing stations located at the front of the class. At these testing stations trained observers gave the students tasks to perform. Each observer scored student performance as being satisfactory or unsatisfactory. The first task on the performance test asked students to "boot" a disk and load DOS into the computer. The observer carefully noted whether each student inserted the disk into the drive in the correct orientation—and did not touch the disk through the opening in the cover. Next the observer watched to see if the student pressed the enter button until the "A>" prompt was displayed. This
indicated that DOS was loaded. Each observer was given a checklist to record the student's performance at each stage of the task. The three observers were carefully trained before the experiment.

All the students were allowed to use the handbook during the performance tests since the instruction was designed to be augmented with this reference tool. The class was given three hours to work through the ten units and complete the tests.

**Attitude Questionnaires**

Three attitude measures were administered to the students. The experimental group and the control group completed an instrument that assessed their feelings about computers before and after the treatment. The pre-treatment questionnaire was administered with the baseline battery. The post-treatment questionnaire was administered just before the posttest. The third instrument, an attitude questionnaire assessing the students' opinions about the courseware, was administered with the post-treatment computer attitude questionnaire.
The pre-treatment attitude questionnaire consisted of an instrument designed by the Minnesota Educational Computing Consortium (MECC) for a study of computer literacy (Anderson et al., 1979). It is an instrument designed by Lewis (1985).

The MECC questionnaire contained 20 items. These questions were designed to assess students' attitudes about computers. The items tapped four themes associated with feelings about computers: computer enjoyment—the degree to which students enjoy computers or learning about computers (items 1, 5, 6, 7, 10); computer anxiety—the level of anxiety or stress caused by working with computers (items 2, 3, 4, 8, 9); computer efficacy—the extent to which a student feels they can successfully deal with computers (items 11 to 15); and educational computer support—the degree to which a student desires the integration of computers into the curriculum (items 16-20). The instrument developed by Lewis contained 20 items and assessed a student's orientation towards computer use and student intentions to use the computer as a tool. Students
responded to a five point Likert scale on machine readable forms.

The post-treatment attitude questionnaire was developed by Collis (1984) (see appendix 7) and used in a study of sex differences in secondary school students' attitudes toward computers. The instrument consisted of 42 items which included 24 computer attitude items, 14 items relating to attitudes toward other school subjects and 4 items relating to experience with computers. One question was omitted as it referred specifically to a high school setting. Students responded to a five point Likert scale on machine readable forms.

Attitudes about Instruction

The students' attitudes towards the CAI program were measured with a questionnaire developed by the researcher (see appendix 8). The instrument was derived from the principles of instruction cited in Chapter II. For example, the use of practice and feedback were identified as elements of good instructional design. These principles were translated into test items by asking students whether they had
enough chance to practice their new skills and whether they knew when they had made a mistake. Students responded to a five point Likert scale on machine readable forms.

Research Design

In order to evaluate the courseware, student performance, student attitude towards computers and student attitude towards the program were measured. A quasi-experimental research design was employed to measure both aspects of the students' learning. In addition to these methods, a courseware critique was undertaken. The critique assessed the courseware according to its adherence to sound principles of instruction. This process further helped isolate design problems suggested by the attitude and performance measures.

A non-equivalent control group design was employed to measure student performance and attitude changes. This design involved the assignment of a pretest and posttest to both an experimental and control group. The groups however, were not randomly assigned and hence
were not experimentally equivalent. Random assignment proved impossible in this instance since the stimulus material was part of a course curriculum. Students enrolled in the course were required to complete the computer literacy courseware in order to fulfill the class requirements.

Campbell and Stanley (1963) outline the strengths and weaknesses associated with this research design. It is stressed that the design is not to be confused with the popular pretest-posttest two-group design. Since the groups in the non-equivalent control group design are not randomly assigned, it becomes more likely that performance and attitude differences between the control group and the experimental group are attributable to group differences rather than treatment effects. Nevertheless, the design does satisfactorily control for threats to internal validity such as the effects of history, maturation, testing and instrumentation (Campbell and Stanley, 1963, p. 43).

Campbell and Stanley (1963) warn about the problems associated with intersession history. Intersession effects are caused by uncontrolled events.
that occur during the experiment itself. Intersession effects may be reduced by running experimental and control sessions at the same time. In this study, intersession effects were a possible confounding influence since the experimental group and the control group completed the baseline test, pretest and posttest at different times and in different locations. In addition, different individuals administered these instruments. These factors all contribute to intersession effects. Since the two classes were scheduled on the same days, tests taken by both the treatment group and control group were administered at approximately the same time. Locations varied, but the effects of such differences were minimal due to the similarity of all the classrooms in the building. Also, the students received the same instructions.

Other problems unique to this design include those of regression and possible maturation/selection interaction effects. Statistical regression may occur when the criterion for control group and experimental group selection is an extreme pretest score. Campbell and Stanley point out that many experimenters
erroneously match subjects in this design in an attempt to compensate for nonrandom assignment. This procedure may cause regression effects. No matching took place in this study; therefore, statistical regression should not be a confounding influence.

The non-equivalent control group design may also be susceptible to maturation/selection and maturation/history effects. An interaction effect between maturation and selection may have posed a problem in this experiment since the experimental group enrolled in a course dealing with computers and information technologies. It could be argued that such a group would be more pre-disposed towards learning about computers. Careful examination of the pretest scores from the control group and experimental group reveal any differences between the two classes.

The non-equivalent control group design presents the same threats to external validity as does the pretest, posttest control group design. Both designs are susceptible to a learning effect caused by the pretest. Careful design and pilot testing of any pretest helps to reduce the possibility that students
may learn from this instrument. The threats of an interaction between treatment and selection and the threat from reactivity are present in the design. Such threats are somewhat reduced however, over a true experiment since the non-equivalent control group design allows the experimenter to sample more widely. In addition, quasi-experimental designs tend to be less reactive than true experiments. In this study, the measurement instruments and the stimulus were carefully integrated into the curriculum of each course. This procedure greatly reduced an awareness among the students that they were taking part in an experiment.

Data Analysis

The data collected for the study was analyzed to determine whether differences existed between scores obtained by each group on the pretest, the attitude questionnaire, demographic variables and the posttest. All data analysis was completed on a mainframe computer using SPSS (Statistical Package for the Social Sciences) (1976). T-tests were run to establish whether significant differences existed between the control group and experimental group on the pretest,
posttest and questionnaire results. A series of ANOVA tests were also run to test whether significant interaction effects existed between the test scores and sex, computer experience, and student major. These tests were undertaken to ensure that the treatment and control groups were experimentally comparable. The results of the performance tests and the attitude towards instruction questionnaire were used to further indicate areas where the program was deficient.

Method

All instruments and treatments were administered during the first two weeks of the 1985 summer session (July 2 to August 4). Both classes met twice for three hours each week.

During the first class for each course, students completed the baseline test and the pretest. Students had 40 minutes to complete the instrument. The course instructor administering the questionnaire to the experimental group informed the students that the results would be used to assess the interest in
computers and that the information would be used to help guide curriculum planning in the future. The control group was also told that the results would be used to measure interest in computers since the course instructor was considering using personal computers when the course was taught again.

At the end of the first class, the experimental group was informed they would spend the next class working through a computer-assisted training program that would help familiarize them with personal computers. The instructor stressed that computer literacy instruction would provide a useful introduction to a course about the impact of computer technology.

At the beginning of the next class, each student was provided with a disk containing the courseware. The instructor gave the class some basic instructions on how to turn-on the computer and 'boot' the disk. Students were then allowed to work through the courseware at their own pace.
Students completed the performance tests as they progressed through the courseware. Immediately after each student had completed each unit and task, they completed the attitude questionnaire and the knowledge posttest. The treatment group was told they would be assessed on how well they had assimilated the concepts in the courseware and how they felt about computers. The attitude questionnaire was completed before the posttest. This prevented students from reacting negatively on the attitude questionnaire because of a poor performance on the posttest.

Similarly, the same posttest and attitude questionnaire were administered to the control group at the beginning of the third period. Both the control group and the experimental group were given 40 minutes to complete the baseline battery.
CHAPTER IV

RESULTS OF EVALUATION
Phase One: Courseware Critique

In this section the prescriptions of Merrill's Component Display Theory, Reigeluth's Elaboration theory and Keller's theory of motivational design of instruction (see chapter II) are used to assess the soundness of the program's instructional design. Each prescription is listed and followed by a description of how well the program adhered to the prescription. Instances where prescriptions are violated are cited as possibly being related to poor achievement by students.

Component Display Theory

**Prescription:** Objectives, instruction and test items should be consistent with respect to content type and task level.

The ATI program is internally consistent. The objectives of the program, although not clearly stated, are to train users in the procedures necessary to operate an IBM microcomputer using DOS commands. Such an objective is clearly at the application level since
students must apply the commands learned in a variety of situations. The content is also clearly procedural since the student must master a series of steps that lead to a specific goal. No test items are included in the program, but the performance tests used to assess student performance are consistent with the objectives of the program.

**Prescription:** A lesson at the application level should use generalities, examples and practice.

The ATI program presents procedural generalities and examples but the program does not provide practice or feedback--students are not allowed to perform the procedure and then receive corrective feedback. What the program offers as practice items are really examples--the students simply copy from the example and are prompted immediately if the wrong key is pressed. For example, the program tells the student: "To prepare a disk in Drive B.....* Type Format B:" (ATI, Frame 59). The student copies "Format B:" and the next frame appears..."To complete the command...* Press the ENTER key". The student is given what appears to be feedback--"Good. The disk has been formatted" and the
next procedure is presented. The program should follow such an example with a question like: "How would you prepare the disk in DRIVE B so that it is compatible with the IBM PC?" The student would then attempt the procedure and receive feedback. New practice items with feedback would continue to be presented until the student mastered the procedure.

**Prescription:** A lesson taught at the application level should use secondary presentation forms where appropriate.

Because of the complexity of the ATI program, secondary presentation forms should be used. The ATI program presents prerequisite information such as the definition of DOS, file, directory, memory, byte and so on. The program also contains effective attention focusing components. The program does not, however, provide memory aids or alternative representations. These secondary presentation forms would help students learn the procedural tasks.

Memory aids may be used to help a learner remember the various DOS commands: "remember DOS rhymes with
BOSS". The mnemonics could be located at the bottom of the screen (along with the instruction) but set off from the other text with attention focusing devices.

Attention focusing devices are used quite skillfully. The split-screen format provides instructions and "feedback" in the bottom portion of the screen (in blue type) while a simulated excerpt from an actual DOS file is displayed in the top portion of the screen (in orange type). The learner can clearly monitor what they are typing and the results.

Alternate presentation forms are not used. The only mode of presentation is symbolic. There are several points in the program where diagrams may be used. Static graphics or dynamic presentations would clarify many of the procedural tasks as well as the conceptual and theoretical supporting content. For example, throughout the unit dealing with copying disks, the verbal explanation might be supplemented with pictures which depict each step. In Frame 81, the top portion of the screen reads: "Insert the source diskette in drive A: Insert destination diskette in drive B:". A picture of an IBM PC, with the appropriate
disk drive highlighted beside each statement, would help clarify the procedure.

The ATI program makes no allowance for learner control except the learners may progress through the program at their own pace. No additional primary or secondary presentation forms are included for students who are having difficulty mastering the procedures. In addition, the knowledge and skills presented in the program are not taught to mastery—there are no tests to show the users whether they are learning the skills.

**Elaboration Theory**

**Prescription:** A course should begin with an epitome lesson and subsequent lessons should be presented in an elaborative sequence (simple to complex).

The ATI program begins with an "epitome" lesson of sorts. The first unit deals with getting into and out of the program. This represents the most fundamental procedure. The program also follows a simple to complex sequence—the first lessons deal with simple, commonly used commands. Later lessons provide instruction on more complex and context specific commands. A
procedural decision structure, however, needs to be integrated into the overall sequencing plan. Students are not taught how to determine what command is appropriate in what situation.

**Prescription:** A lesson should contain a summary and a synthesis of previous material.

The ATI program contains lesson summarizers (FRAME 74, for example) but lacks contextual summarizers and synthesizers of any kind. The use of synthesizers, in the form of flow charts, would help students integrate material from one lesson to another and aid in developing the procedural decision skills mentioned above. Contextual summarizers, which provide examples of procedures would offer additional feedback and may have motivational benefits.

**Keller’s Model of Motivational Design**

**Prescription:** Instruction should gain student attention, be relevant to students, build student confidence and be satisfying.
The ATI program lacks important motivational strategies. Motivational strategies are not essential for instruction if the students are highly motivated. In this case, the use of computers as the primary delivery medium is highly motivating (novel). Nevertheless, for the anxious user, additional motivational strategies may be helpful. For instance, there is no attempt to stress the relevance of the instruction. This may be accomplished by briefly listing, at the beginning of the program, what the student will learn and why the skills are important. The program makes the content accessible by explaining technical concepts in simple language.
Phase Two: Experiment

Description of Sample

This section provides a detailed description of the sample used in the study. The sample is described according to the following independent variables: age, sex, year of study, major area of study, computer experience and computer knowledge as measured by a pretest. In addition, the sample is described in terms of attitudes towards computers as measured by a baseline questionnaire.

Age and Sex

There were 21 students in the treatment group; 12 were female; nine were male. The mean age of the treatment group was 23.14 years (see table 1). The oldest student was 49 years old and the youngest was 19. There were 18 students in the control group; six females and 12 males. The mean age of the control group was 25.94 years. The oldest student was 50 and the youngest 19. The samples did not differ significantly on age (t=-1.37, p>.05). Sex did not
have a significant effect on the two computer attitude scales (Lewis and MECC) and the pretest (see tables 2).

Table 1: Description of Sample

<table>
<thead>
<tr>
<th>Variable</th>
<th>Experimental (n=21)</th>
<th>Control (n=18)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age</td>
<td>X=23.14 (SD=5.29)</td>
<td>X=25.94 (SD=7.47) t=1.37</td>
</tr>
<tr>
<td>Gender</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Male</td>
<td>42.9% (9)</td>
<td>33.3% (6)</td>
</tr>
<tr>
<td>Female</td>
<td>57.1% (12)</td>
<td>66.6% (12)</td>
</tr>
<tr>
<td>Major</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Comm. St.</td>
<td>76.2% (16)</td>
<td>66.6% (12)</td>
</tr>
<tr>
<td>Pol. Sci.</td>
<td>5.0% (1)</td>
<td></td>
</tr>
<tr>
<td>Nursing</td>
<td>5.0% (1)</td>
<td></td>
</tr>
<tr>
<td>Vis. Arts</td>
<td>5.0% (1)</td>
<td></td>
</tr>
<tr>
<td>Rel. St.</td>
<td>5.0% (1)</td>
<td>5.5% (1)</td>
</tr>
<tr>
<td>English</td>
<td>5.0% (1)</td>
<td>5.5% (1)</td>
</tr>
<tr>
<td>Psych.</td>
<td></td>
<td>5.5% (1)</td>
</tr>
<tr>
<td>Urban St.</td>
<td></td>
<td>5.5% (1)</td>
</tr>
<tr>
<td>Economics</td>
<td></td>
<td>5.5% (1)</td>
</tr>
<tr>
<td>Social Work</td>
<td></td>
<td>5.5% (1)</td>
</tr>
<tr>
<td>Year of Study</td>
<td></td>
<td></td>
</tr>
<tr>
<td>First</td>
<td>9.5% (2)</td>
<td>16.6% (3)</td>
</tr>
<tr>
<td>Second</td>
<td>33.3% (7)</td>
<td>16.6% (3)</td>
</tr>
<tr>
<td>Third</td>
<td>28.6% (6)</td>
<td>61.1% (11)</td>
</tr>
<tr>
<td>Fourth</td>
<td>19.0% (4)</td>
<td>5.5% (1)</td>
</tr>
<tr>
<td>Fifth</td>
<td>9.5% (2)</td>
<td></td>
</tr>
<tr>
<td>Computer Experience</td>
<td>X=3.86 (SD=0.97)</td>
<td>X=4.06 (SD=0.96) t=0.65</td>
</tr>
</tbody>
</table>

* Note: p<0.05

Year of Study and Area of Specialization

All participants in the experiment were university undergraduates. They had attended university for one to five years. Two students from the experimental group
and three from the control group were in their first year. Seven students from the experimental group and three from the control group were in their second year. Six students from the experimental group and 11 from the control group were in their third year. Four students from the experimental group and one student from the control group were in their fourth year and two students from the experimental group were in their fifth year of undergraduate study. Year of study did not have a significant effect on the computer attitude variables and pretest scores (see table 2).

Students represented 10 areas of specialization within the university: Communication Studies, Political Science, Nursing, Visual Arts, Religious Studies, English, Psychology, Urban Studies, Economics and Social Work. Sixteen students from the experimental group and 12 from the control group were enrolled in Communication Studies. Five students in the experimental group were in Political Science, Nursing, Visual Arts, Religious Studies and English. In the control group six students were enrolled in Religious

**Table 2: Source of Variation in Computer Attitude and Pretest**

<table>
<thead>
<tr>
<th>Variable One (Lewis)</th>
<th>Source of Variation</th>
<th>SS</th>
<th>DF</th>
<th>MS</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sex</td>
<td>13.02</td>
<td>1</td>
<td></td>
<td>13.04</td>
<td>0.37</td>
</tr>
<tr>
<td>Year of Study</td>
<td>58.47</td>
<td>4</td>
<td></td>
<td>14.62</td>
<td>0.41</td>
</tr>
<tr>
<td>Computer Experience</td>
<td>33.22</td>
<td>3</td>
<td></td>
<td>11.07</td>
<td>0.31</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variable Two (MECC)</th>
<th>Source of Variation</th>
<th>SS</th>
<th>DF</th>
<th>MS</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sex</td>
<td>157.65</td>
<td>1</td>
<td></td>
<td>157.65</td>
<td>0.80</td>
</tr>
<tr>
<td>Year of Study</td>
<td>543.60</td>
<td>4</td>
<td></td>
<td>135.90</td>
<td>0.69</td>
</tr>
<tr>
<td>Computer Experience</td>
<td>598.60</td>
<td>3</td>
<td></td>
<td>199.53</td>
<td>1.01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Variable Three (Pretest)</th>
<th>Source of Variation</th>
<th>SS</th>
<th>DF</th>
<th>MS</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sex</td>
<td>0.71</td>
<td>1</td>
<td></td>
<td>0.23</td>
<td>3.48</td>
</tr>
<tr>
<td>Year of Study</td>
<td>0.14</td>
<td>4</td>
<td></td>
<td>0.03</td>
<td>0.17</td>
</tr>
<tr>
<td>Computer Experience</td>
<td>0.43</td>
<td>3</td>
<td></td>
<td>0.14</td>
<td>0.69</td>
</tr>
</tbody>
</table>

Note: *p<.05
Computer Experience

All participants in the study indicated their level of computer experience by checking whether they felt very experienced (1), experienced (2), somewhat experienced (3), not very experienced (4), not experienced (5). The mean level of experience for the experimental group was 3.86 (SD=0.97) and the mean level of experience for the control group was 4.06 (SD=0.94). The samples did not differ significantly with respect to their reported computer experience (t=-0.65, p>0.05) (see table 1).

Pretest

A nine-item pretest, developed by the researcher and based on the major content points in the ATI training program, was administered to the experimental group and the control group. The results on this test were expected to indicate whether the groups already possessed knowledge presented in the program and whether the groups differed in their knowledge of the material. Each item on the pretest tested knowledge of factual information presented in the program. The
factual information required recall of definitions, procedures and principles. Responses to the pretest questions were scored as either right or wrong. 18 items were designed and randomly assigned to a pretest and posttest instrument.

The first question on the pretest asked students to give the meaning of the letters D.O.S. (Disk Operating System). One student in the experimental group and one student in the control group answered the question correctly. The second question asked why DOS was necessary for the operation of the computer. No student from either group answered correctly. The third question asked how a user would know when the computer was ready to be operated. Again no student answered the question correctly. The next question asked the student to define the term "computer file". Two students from the experimental group and none from the control group got the right answer. The fifth question asked for the definition of "extension letters". One student from the experimental group answered correctly. No student from either the experimental group or the control group was able to give the definition of a
"system transfer message" (question 6), explain why it is important to format a disk (question 7), and give an example of a "wild card" command (question 8). One student from the control group was able describe a situation where you would use an abort command (question 9).

The experimental group obtained a total mean score of 0.24 (SD=0.54). The maximum score was 9. The control group obtained a total mean score of 0.11 (SD=0.32). The groups did not differ significantly on the pretest measure (t=0.91, p>0.05) (see tables 3 and 4).

Table 3: Student Achievement on Pretest

<table>
<thead>
<tr>
<th>Test Item</th>
<th>Experimental (n=21)</th>
<th>Control (n=18)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Correct</td>
<td>% Correct</td>
</tr>
<tr>
<td>Dos Definition</td>
<td>1</td>
<td>5.0%</td>
</tr>
<tr>
<td>Need for Dos</td>
<td>0</td>
<td>---</td>
</tr>
<tr>
<td>Computer Ready</td>
<td>0</td>
<td>---</td>
</tr>
<tr>
<td>File Definition</td>
<td>2</td>
<td>9.5%</td>
</tr>
<tr>
<td>Extension Letters</td>
<td>1</td>
<td>5.0%</td>
</tr>
<tr>
<td>System Transfer</td>
<td>0</td>
<td>---</td>
</tr>
<tr>
<td>Message</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Format</td>
<td>0</td>
<td>---</td>
</tr>
<tr>
<td>Wild Cards</td>
<td>0</td>
<td>---</td>
</tr>
<tr>
<td>Abort Command</td>
<td>0</td>
<td>---</td>
</tr>
</tbody>
</table>
Attitude About Computers

The baseline battery included a questionnaire that measured a respondent's attitude towards using microcomputers at school. This section of the battery was made up of two instruments. The first was developed by Lewis (1985). The mean score for the experimental group was 39.24 (SD=4.83). The mean score for the control group was 40.89 (SD=6.26). The samples did not differ significantly with respect to their scores on the Lewis attitude questionnaire (t=0.93, p>0.05) (see table 4).

The second part of the attitude measure consisted of the MECC (Minnesota Educational Computer Consortium) computer attitudes questionnaire. The mean score for the experimental group was 46.43 (SD=12.38) and the mean score for the control group was 45.44 (SD=14.33). The samples did not differ significantly with respect to their scores on the MECC attitude questionnaire (t=0.22, p>0.05).
Table 4: Pretest and Attitude Measures

<table>
<thead>
<tr>
<th>Variable</th>
<th>Experimental (n=21)</th>
<th>Control (n=18)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>SD</td>
</tr>
<tr>
<td>Pretest</td>
<td>0.24</td>
<td>0.54</td>
</tr>
<tr>
<td>Attitude Towards Computers (MECC)</td>
<td>46.47</td>
<td>12.38</td>
</tr>
<tr>
<td>Attitudes Towards Computers (Lewis)</td>
<td>39.24</td>
<td>4.83</td>
</tr>
</tbody>
</table>

Note: *p<.05

Description of Sample: Summary

The results of statistical analysis indicates that the two groups used in the study are experimentally comparable with respect to the variables used. There were no significant differences between the groups with respect to the independent variables of age, sex, year of study, major area of study and computer experience. In addition, no statistical difference appeared between the mean scores on the knowledge pretest. Finally, no significant differences existed between the groups with respect to attitude towards computers.
Evaluation of Performance

Posttest

The first question on the posttest asked students how to specify a disk drive. Eleven students in the experimental group answered the question correctly. No students in the control group answered correctly. The second question asked how space is measured on a computer disk. Fifteen students from the experimental group and five students from the control group answered correctly. The third question asked what prompt would indicate that the computer was waiting for a command. Fourteen students from the experimental group and none from the control group answered correctly. The next question asked what message appears when DOS has been loaded into the computer. Three students from the experimental group and none from the control group answered correctly. The fifth question asked how one would check how many files were stored on a disk. Eleven students from the experimental group and none from the control group answered correctly. The sixth question asked what procedure was required to make a disk compatible with the IBM computer. Eight students
from the experimental group and none from the control group answered correctly. The next question asked what a "Fault Error Message" indicated. Only one student from the experimental group answered correctly. Question eight asked for two reasons why copies of a disk should be made. Seven students from the experimental group and one from the control group answered correctly. Finally, question nine asked what part of a computer file is analogous to the table of contents in a book. Eight students from the experimental group answered correctly.

The experimental group received a mean score of 3.90 (SD=1.92). The maximum score was nine. The control group obtained a mean score of 0.33 (SD=0.59). The groups differed significantly ($t=7.57$, $p<0.05$) on the posttest measure (see table 5).
Table 5: Student Achievement on Posttest

<table>
<thead>
<tr>
<th>Test Item</th>
<th>Experimental (n=21)</th>
<th>Control (n=18)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Correct</td>
<td>% Correct</td>
</tr>
<tr>
<td>Change Drive</td>
<td>11</td>
<td>52.4%</td>
</tr>
<tr>
<td>Space</td>
<td>15</td>
<td>71.4%</td>
</tr>
<tr>
<td>Ready Prompt</td>
<td>14</td>
<td>66.6%</td>
</tr>
<tr>
<td>Loaded Message</td>
<td>3</td>
<td>14.3%</td>
</tr>
<tr>
<td>Store Files</td>
<td>11</td>
<td>52.4%</td>
</tr>
<tr>
<td>Format</td>
<td>8</td>
<td>38.1%</td>
</tr>
<tr>
<td>Fault Error</td>
<td>1</td>
<td>5.0%</td>
</tr>
<tr>
<td>Copy Disk</td>
<td>7</td>
<td>33.3%</td>
</tr>
<tr>
<td>Directory</td>
<td>8</td>
<td>38.1%</td>
</tr>
<tr>
<td>Overall Posttest Scores</td>
<td>X=3.90</td>
<td>SD=1.92</td>
</tr>
</tbody>
</table>

Note: *p<0.05

Performance Tests

Students in the experimental group completed four performance tests as they worked through the courseware. Each test consisted of a number of subtests which assessed the student's ability to perform a set of tasks just presented in the program. Each tested procedural content and required the student to apply material they had learned ("use" level tasks). Both the content type and task level of the performance tests were consistent with the stated objectives of the
courseware. The criteria for assessing student performance were: ability to perform the task (scored as correct or not correct by a trained observer); time on task, compared to an expert user, and whether the student referred to the user's manual. An expert user is defined as someone who has worked with DOS frequently for at least six months. Such a person was asked to complete each task and their times were recorded. Students who took longer on task and looked at the user's manual frequently were deemed to be having difficulty.

The first performance test (see table 6) asked the student to complete the following tasks: 1) respond correctly to the date and day prompts and 2) determine how much space was used on the disk. The student completed the test after completing unit C in the program. In task one, 52.4% of the students correctly responded to the date and day prompts. 23.8% of all students referred to the user's handbook. The mean time on task was 48.57 seconds. An expert user would require approximately 5 seconds to perform the task.
In the second task, 90.5% of the students correctly indicated how much disk space was used. 90.5% of the students selected the DIR command (Subtask One) while 95.5% pressed the enter key after the command (Subtask Two). 14.3% of all students referred to the user's handbook. The mean time on task was 28.57 seconds. An expert user would require approximately 10 seconds to perform the task.

Table 6: Performance Test One

<table>
<thead>
<tr>
<th>Task</th>
<th>% Correct (n=21)</th>
<th>% Using Manual (n=21)</th>
<th>Time on Task (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task One</td>
<td>52.4% (11)</td>
<td>23.4% (5)</td>
<td>48.57</td>
</tr>
<tr>
<td>Task Two</td>
<td>95.5% (20)</td>
<td>14.3% (3)</td>
<td>28.57</td>
</tr>
<tr>
<td>(Subtask One)</td>
<td>90.5% (19)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Subtask Two)</td>
<td>95.5% (20)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The second performance test (see Table 7) included 3 tasks. Students were asked to indicate how much space remained on the disk in drive B, format a disk and format a disk with a copy of DOS. In task 3, 38.1% of the students correctly changed disk drives from ">A" to ">B", while 71.4% correctly selected the CHK DISK command. Every student pressed the enter key to activate the command. 71.4% correctly identified how
much space remained on the disk and 47.6% referred to the user’s handbook. The mean time for completion of task 3 was 108.57 seconds.

81% of the students correctly selected the FORMAT command in task 4 and 95.2% remembered to press the enter key. 14.3% needed to refer to the user’s handbook. The mean time on task was 39.52 seconds. When asked to format a disk with a copy of DOS, 66.7% of the students correctly used the FORMAT/S command. 85% of the students pressed the enter key. The mean time for completion of task 5 was 23.57 seconds.

Table 7: Performance Test Two

<table>
<thead>
<tr>
<th>Task</th>
<th>% Correct (n=21)</th>
<th>% Using Manual (n=21)</th>
<th>Time on Task (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task Three</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Subtask One)</td>
<td>38.1% (8)</td>
<td>47.6% (10)</td>
<td>108.57</td>
</tr>
<tr>
<td>(Subtask Two)</td>
<td>71.4% (15)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task Four</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Subtask One)</td>
<td>81.0% (17)</td>
<td>14.3% (3)</td>
<td>39.52</td>
</tr>
<tr>
<td>(Subtask Two)</td>
<td>95.2% (20)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task Five</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Subtask One)</td>
<td>66.7% (14)</td>
<td>23.4% (5)</td>
<td>23.57</td>
</tr>
</tbody>
</table>

The third performance test (see table 8) included three tasks. Students were asked to make a copy of a disk; check to see whether the copy was exact and to
make a copy of a single file. In task six, 52.1% of the students used the DISKCOPY command and all the students remembered to press the enter key after typing a command. 85.7% pressed any key to initiate the copying process and 85.7% pressed the 'N' key to indicate that no more copies were to be made. 42.9% of the students referred to the user's handbook for help. The mean time for task six was 97.62 seconds.

Next, the students were asked to compare the new disk to ensure that the copy was exact. 95% of the students used the COMPDISK command and all students pressed the enter key. 19% of the students used the handbook. The mean time for task seven was 44.76 seconds.

On the last task in the third performance test, each student was asked to copy a single file on the same disk. 52.4% of the students used the COPY command and everyone remembered to press the enter key. 33.3% of the students used the DIR command to check whether the file had been successfully copied. 52.4% of the students used the handbook for help. The mean time for the completion of task eight was 88 seconds.
Table 8: Performance Test Three

<table>
<thead>
<tr>
<th>Task</th>
<th>% Correct (n=21)</th>
<th>% Using Manual (n=21)</th>
<th>Time on Task (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task Six</td>
<td>52.4% (11)</td>
<td>42.7% (9)</td>
<td>97.62</td>
</tr>
<tr>
<td>(Subtask One)</td>
<td>100.0% (21)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Subtask Two)</td>
<td>85.7% (18)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Subtask Three)</td>
<td>85.7% (18)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task Seven</td>
<td>95.0% (20)</td>
<td>52.4% (11)</td>
<td>44.76</td>
</tr>
<tr>
<td>(Subtask One)</td>
<td>100.0% (21)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task Eight</td>
<td>52.4% (11)</td>
<td>52.4% (11)</td>
<td>169.00</td>
</tr>
<tr>
<td>(Subtask One)</td>
<td>100.0% (21)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Subtask Two)</td>
<td>33.3% (7)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The last performance test (see table 9) contained four tasks: list the contents of a file; rename a file; rename a number of files at once and erase a file. In performance task nine, students were asked to list the contents of a file on the monitor. 76.2% of the students correctly used the TYPE command and 95.2% used the enter command. 52.4% of the students used the user's handbook for help. The mean time for completion of task nine was 77.86 seconds.

In task ten, students were asked to rename a file. 76.2% used the REN command correctly and 95.2% remembered to use the enter key. 71.4% used the DIR command to check whether the file was renamed and 55.1%
referred to the user's handbook. The mean time on task was 77.86 seconds.

In task eleven, students were requested to rename a number of files simultaneously. 57.1% of the students correctly used the REN* command and all the students remembered to press the enter key. 76.2% used the DIR command to check whether the files were renamed and 42.9% used the handbook. The mean time for completion of task eleven was 115.0 seconds.

Table 9: Performance Test Four

<table>
<thead>
<tr>
<th>Task</th>
<th>% Correct (n=21)</th>
<th>% Using Manual (n=21)</th>
<th>Time on Task (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task Nine</td>
<td>52.4% (11)</td>
<td>52.4% (11)</td>
<td>88.0</td>
</tr>
<tr>
<td>(Subtask One)</td>
<td>100.0% (21)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task Ten</td>
<td>76.2% (16)</td>
<td>57.1% (12)</td>
<td>77.86</td>
</tr>
<tr>
<td>(Subtask One)</td>
<td>95.2% (20)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Subtask Two)</td>
<td>71.4% (15)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task Eleven</td>
<td>57.1% (12)</td>
<td>42.9% (9)</td>
<td>115.00</td>
</tr>
<tr>
<td>(Subtask One)</td>
<td>100.0% (21)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Subtask Two)</td>
<td>76.2% (16)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task Twelve</td>
<td>90.5% (19)</td>
<td>14.3% (3)</td>
<td>30.95</td>
</tr>
<tr>
<td>(Subtask One)</td>
<td>100.0% (21)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(Subtask Two)</td>
<td>95.2% (20)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

In the final task, 90.5% of the students correctly used the ERASE command to erase a file and all the students remembered to press the enter key. 95.2% of
the students used the DIR command to confirm the file was erased and 14.3% used the handbook. The mean time on task was 30.95 seconds.

Attitude Change

Attitudes Towards Computers

A second questionnaire (Collis, 1984) measuring student attitudes towards computers was administered to both the experimental and control groups. The experimental group had a mean score of 118.90 (SD=7.78) the control group received a mean score of 115.00 (SD=10.17) on the attitude questionnaire. The scores did not differ significantly (see table 10).

Table 10: Student Attitude Towards Computers (Posttest)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Experimental (n=21)</th>
<th>Control (n=18)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Mean</td>
<td>SD</td>
</tr>
<tr>
<td>Attitude Towards Computers</td>
<td>118.9</td>
<td>7.78</td>
</tr>
</tbody>
</table>

Note: *p<.05
Attitudes Towards Instruction

The last item to be administered to the experimental group was a questionnaire measuring attitudes towards the courseware. The instrument contained 30 items which measured attitudes towards twelve aspects of the courseware's design (see table 11). Questions assessed whether a student believed a specific principle of instructional design was violated in the courseware. The principles included: need for practice, individualization, motivation, readiness, statement of objectives and feedback. In addition, questions assessed attitudes regarding the effectiveness of the instructions, the user's handbook and screen layout. Finally, students were asked whether they liked working with computer-aided instruction and whether they preferred teacher interaction along with the CAI. Students responded to a Likert type scale: (1), strongly disagree; (2), disagree; (3), no opinion; (4), agree; (5), strongly agree.

Students responded positively to questions asking whether the instruction contained sufficient practice. The mean score on this index was 3.38 (SD=1.01).
Students were undecided when asked whether the program provided motivation. The mean score on the motivation index was 3.00 (SD=0.86). Questions concerning learning control were subsumed under an index labeled "individualized". Again, students were neutral in their response to this item. The mean score was 2.96 (SD=0.89). Students disagreed slightly with the statement that they possessed the necessary entry level skills. The mean score on the index, 'readiness' was 2.74 (SD=0.98). Students disagreed slightly with statements indicating that the learning objectives were clearly stated at the beginning of the program. The mean score on the 'objectives' index was 2.83 (SD=1.14). Students strongly agreed that they received sufficient feedback during the program. The mean score on the 'feedback' index was 3.36 (SD=1.02).

Students agreed with statements indicating the clarity of the information presented in the program. The mean score on this index was 3.95 (SD=0.92). Students, however, disagreed with statements of the handbook's clarity. The mean score on this item was 2.38 (SD=1.12).
Students agreed with the statement: "I enjoy learning from computers". The mean score on this item was 3.21 (SD=1.04). Students also agreed with the statements indicating a preference for human interaction along with the computer program. The mean score on this index was 3.14 (SD=1.40). Finally, when asked whether the same instruction could be presented in a textbook, students agreed that it could. The mean score on this item was 3.67 (SD=1.07).

Table II: Student Attitudes Towards Instruction.

<table>
<thead>
<tr>
<th>Attitude Index</th>
<th>Mean (n=21)</th>
<th>Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Practice</td>
<td>3.38</td>
<td>1.01</td>
</tr>
<tr>
<td>Individualized</td>
<td>2.96</td>
<td>0.89</td>
</tr>
<tr>
<td>Motivation</td>
<td>3.00</td>
<td>0.86</td>
</tr>
<tr>
<td>Readiness</td>
<td>2.74</td>
<td>0.98</td>
</tr>
<tr>
<td>Objectives</td>
<td>2.83</td>
<td>1.14</td>
</tr>
<tr>
<td>Feedback</td>
<td>3.36</td>
<td>1.02</td>
</tr>
<tr>
<td>Handbook</td>
<td>2.38</td>
<td>1.12</td>
</tr>
<tr>
<td>Screen</td>
<td>3.95</td>
<td>0.92</td>
</tr>
<tr>
<td>Instructions</td>
<td>3.17</td>
<td>0.85</td>
</tr>
<tr>
<td>Liked CAI</td>
<td>3.21</td>
<td>1.04</td>
</tr>
<tr>
<td>Human Help</td>
<td>3.14</td>
<td>1.40</td>
</tr>
<tr>
<td>Text as Good</td>
<td>3.67</td>
<td>1.07</td>
</tr>
</tbody>
</table>
CHAPTER V

DISCUSSION
Objectives of Study

The objectives of the study were as follows:

1) To develop a plan to evaluate microcomputer courseware.

2) To generate a list of design criteria pertinent to microcomputer courseware.

3) To develop and execute a pilot study, based on the evaluation plan, to critically assess a piece of representative courseware.

4) To suggest ways to empirically test the validity of the evaluation plan.

The Evaluation Method

Initially, an evaluation plan for microcomputer courseware based upon evaluation models and instructional theory was developed. The evaluation plan contained three major elements: assessment of student performance; assessment of incidental learning and a courseware review. The review was guided by prescriptions derived from instructional theory which addressed micro, macro and motivational design.
considerations. These prescriptions served as design criteria against which the courseware was judged.

A commercially available computer literacy CAI training program was selected for evaluation. The courseware: "Teach Yourself PC DOS" (A.T.I, 1984), trains users to load and operate the 'disk operating system' (DOS) used in the IBM personal computer. A treatment group worked through the courseware during a regularly scheduled class and at predetermined intervals, students completed a short "performance test" which assessed how well they had assimilated information presented in a unit of the program. Immediately after completing the courseware, all students completed a posttest, which consisted of a course content test and two questionnaires—one measuring attitudes about computers and the other measuring attitudes about the quality of the instruction. A control group also completed the posttest and the computer attitude questionnaire. Both groups completed a baseline battery which contained a pretest and a computer attitude questionnaire. Demographic information was also obtained from each
participant during the baseline battery. Students' age, sex, major, year of study and amount of previous computer experience were obtained.

Major Findings of Pilot Study

Description of the Sample

Results of statistical tests comparing baseline scores between the control group and the experimental group showed that no statistical difference existed between each group on each measure. The groups, therefore, were experimentally comparable with respect to age, sex, university major, year of study and amount of previous computer experience. The average student was about 20 years old, in either their second or third year of university and enrolled at the University of Windsor. There were similar numbers of males and females in each group. Finally, most students indicated they had little, or no previous experience with computers.

Results from the content pretest indicated that both the control group and experimental group had no previous knowledge of the material in the ATI
courseware. The average score on the pretest was less than one out of nine correct. There was also no difference between the groups on their attitudes toward computers. The mean scores were almost identical on each attitude instrument. On average, students held favourable attitudes towards computers.

Student Performance

Student performance was assessed by means of a posttest and a series of 12 performance tests which were administered during the treatment. Results of the scores on the posttest showed that the experimental group learned significantly more than the control group which did not receive the treatment. Nevertheless, the mean score of the experimental group on the posttest was relatively poor—only 3.9 out of 9 correct answers. In addition, student achievement on the performance tests and the design problems cited in the courseware review, also brings the quality of the courseware into serious question. The procedures presented in the courseware were not mastered and only a few approached mastery level. An effective piece of instruction should bring all students to mastery level by the end
of the course. This means all the students should have successfully completed the performance tests and scored well on the posttest.

It appears the design flaws had an impact upon student scores on the 12 performance tasks. Only half (52.4%) of the students were able to correctly respond to the date and day prompts visible at the beginning of the DOS program. This procedure is essential for a user to get "into" the program and begin working with a particular file. Similarly, only half (52.4%) were able to make a copy of a disk, make a copy of a file and access the contents of a file. Only a third (33.3%) of the students knew how to check the disk's directory—a list of the files on a disk—and only a third (38.4%) could correctly change disk drives.

80% or more of the students were able to successfully complete only three performance tasks. These included formatting a disk (81%), erasing a file (90.5%) and comparing a newly copied disk with the original for accuracy (95%). High achievement on these tests may be explained by the trivial nature of the task. For example, the format command is FORMAT, the
erase command is ERASE and the compare command is COMPDISK. In addition, the command protocol may have been mastered at this point due to learning from previous performance tests.

All students had access to a "user's handbook" that contained the correct procedures for accomplishing each task. The percentage of students using the manual ranged from 14.6% on task two to 52.4% on tasks seven, eight and nine. This suggests that many students had not mastered the task after the program. Many students also indicated that they found the handbook difficult to use. In addition, time on task for each test was substantial, ranging from a mean time of 23 seconds to 169 seconds. The majority of the tasks required over one minute to complete. An expert user is able to complete any of these tasks in five to 15 seconds.

The students, however, rated the ATI program favourably with respect to feedback. The satisfaction index for feedback was 3.36. This result is not surprising since the program was filled with many prompts. These prompts, however, merely gave the correct answer rather than pointing out what the
student had done incorrectly. Similarly, students rated the program favourably with respect to practice. The satisfaction index was 3.38. Again, what appeared to be practice was really copying an example.

The lack of motivational design components was another problem cited in the review. The students were neutral in terms of their response on the motivation index (how motivating they found the courseware, X=3.00). It is, however, difficult to assess this result since the students were probably highly motivated to begin with. The students indicated that they liked CAI (X=3.67) but would also like some human help to supplement the courseware (X=3.14). The students also agreed with the statement that a textbook could provide the same material as effectively as the CAI program (X=3.67). Acceptance of the CAI program is probably due to the novelty of using computers as a mode of instruction, the fact that the class had a prior interest in computers and that the students were highly motivated prior to the course. The effective use of attention focusing devices was reflected by the
students' agreement with the statement that the screen was easy to read and effectively formatted (X=3.95).

Student Attitudes

Student attitudes towards computers and school appeared unaffected by the courseware. The high degree of interest in computers at the onset of the evaluation and the relatively short exposure to treatment may be the cause. In addition, other methods of assessing student attitudes, such as observation and interviews, may have proved more effective. Nevertheless, it is important to assess incidental learning, such as attitudes towards computers and school, when evaluating courseware.

Limitations of the Pilot Study

Several limitations of the pilot study may be cited. The most serious is that students were selected and assigned to treatment and control groups conveniently. Convenient selection of students limits the generalizability of the results. It is difficult to generalize the results of this study beyond third and fourth year students enrolled in a computer skills
course. The lack of random assignment makes it more difficult to attribute posttest performance to treatment effects rather than group differences.

Despite attempts to demonstrate the homogeneity of the subject pool, factors other than those measured may have influenced scores on the posttest and performance tests. The most notable of these factors is the treatment group's favourable predisposition towards computers and technology and their high degree of motivation. This is a major confounding variable. Nevertheless, it tends to strengthen the results of the study. Despite a substantial advantage, the treatment group failed to master any of the procedures in the ATI program.

Since the courseware was selected conveniently, it is more difficult to generalize the results of the evaluation to all courseware. Nevertheless, several similar products have been cited making such generalizations more plausible. A better design would evaluate several pieces of courseware. If an evaluation requires generalizable results, products
should be selected randomly or a representative selection made by a panel of courseware developers.

The small sample size (39) limits the power of the study. Lack of power results in relatively important between group differences not being statistically significant. A small sample size in conjunction with the selection and assignment procedures further reduces the generalizability of the results.

Conclusions and Implications

The study suggests the importance of several methods for evaluating courseware. These methods are useful for developers who may use them to improve a specific courseware design and researchers interested in assessing the effectiveness of a class of products. The use of paper and pencil tests and performance tests clearly show whether students have learned the desired skills. The use of a checklist based on instructional design principles, may provide a reasonable alternative to objective tests when such tests are not feasible. Checklists that translate instructional principles into easily understood criteria for evaluators are few.
More development work in this area is needed. Attitude questionnaires are also valuable in assessing the unintended learning that may occur during instruction. For example, a student may be put off computers by using computer-aided instruction. Attitude questionnaires provide valuable additional information frequently ignored in many evaluations. These evaluation methods should also be employed during the development phase of courseware production.

"Formative" evaluation provides valuable information for developers about problems in courseware design before the product reaches the marketplace.

The limitations of the pilot study also suggest procedures that increase the validity and generalizability of a courseware evaluation. These include random assignment of students, random selection of courseware (where necessary) and sufficiently large sample sizes (at least 20 per group).

In addition, the study identifies some important design strategies that should be incorporated in all CAI programs. The use of practice and feedback at the "lesson" level is vital. There can be no learning
without guided practice. Synthesis and summary of material presented in earlier lessons is equally vital at the "course" level. Students need an opportunity to review material and study the inter-relationship between concepts. Motivational strategies are also important to effective instruction, especially for students who find the computer intimidating. The more refined prescriptions, e.g. secondary presentation forms, suggested by Reigeluth and Merrill, simply make good instruction even better. These prescriptions appear intuitively obvious, but they are often ignored in courseware design and courseware evaluation.

Since the courseware selected for evaluation is representative of products currently available, the quality of much CAI courseware may be suspect. ATI for example, markets over 20 tutorial programs. Other examples include IBM's tutorial program for the IBM PC, Apple's tutorial for the Apple IIC and "Wordstar Professional"--a tutorial designed to teach a word-processing program.

This is alarming given the use of such products in schools, universities and business. This suggests that
the buyers of courseware must become better consumers and the producers of courseware must become better designers. If the situation does not change, institutions may become frustrated as courseware products fail to meet expectations. Ultimately CAI may follow the demise of instructional television for lack of quality software.

The use of sound instructional principles as both design and evaluation criteria is the best way to ensure quality courseware. The results of this study suggest the importance of prescriptive instructional theory in both areas. When courseware is not designed and evaluated systematically student learning is compromised. The quote of Wager's cited earlier: "poor, but "fancy" programs may get more respect than effective but "plain" programs" (Wager, 1982, p 23) unfortunately rings true. The ATI program is marketed on the basis of visual appeal rather than how well it trains computer users.

As evaluation criteria become more standardized and courseware evaluations are more common, consumers will be better able to select quality material. As
consumer awareness grows, manufacturers will be forced to produce better products and with better products the microcomputer will reach its full potential as an instructional tool.

Suggestions for Future Research

Future studies need to empirically test the validity of this courseware evaluation plan. Such a test may be achieved by modifying the ATI program according to the findings of the evaluation and then reevaluate the courseware to see if the students achieve mastery. If the students successfully learn the DOS commands from the modified program, the evaluation criteria are valid.
Appendix 1
Demographic Questionnaire

Thank you for participating in this study. Your help is greatly appreciated. Before you begin this questionnaire, please fill in the information requested on this page. Thanks again for taking part in this study.

AGE: [ ]

SEX: [ ]

YEAR OF STUDY: [ ]

MAJOR: [ ]

COMPUTER EXPERIENCE (CHECK ONE):
VERY EXPERIENCED ( ); EXPERIENCED ( ); SOMewhat EXPERIENCED ( ); LITTLE EXPERIENCE ( ); NO COMPUTER EXPERIENCE ( ).

If you have worked with computers please describe in a sentence or two your experience.
Appendix 2  
Attitude Towards Computers (Pre)  

HOW I LEARN QUESTIONNAIRE

In this section, respond to each question by blackening the letter which most reflects your response. The letters correspond to the reactions as follows:

A - Strongly Agree  
B - Agree  
C - Neutral  
D - Disagree  
SD - Strongly Disagree

115. I like to learn on my own.
116. I don't mind re-doing something because of a malfunction.
117. I like working at my own pace.
118. I am good at reading instructions about how to do something and then doing it.
119. I like doing things precisely.
120. I find it easy to memorize procedures for doing something.
121. I like situations where there is one clear answer.
122. I like to operate electronic devices.
123. I like learning with a few other people.
124. When I'm learning something new, I would like someone to show me how to do it.
125. I don't mind trying to work out a problem by trying several ways of doing it.
126. I find it hard to learn how to do something new.
127. I like situations in which there are several ways to achieve the same goal.
128. When I type papers, I would like to make changes to them.
HOW I LEARN QUESTIONNAIRE

A - STRONGLY AGREE  B - AGREE  C - NEUTRAL
D - DISAGREE  E - STRONGLY DISAGREE

129. Given the opportunity, I would like to learn word-
    processing on a computer.

130. Given the opportunity, I would learn how to use the
    computer to keep records like bibliographies, addresses
    etc.

131. Given the opportunity, I would learn the computer as an
    advanced calculator.

132. I would like to learn certain content using computer
    assisted instruction.

133. I would like to use the computer to locate, research in a
    specific area.

134. I would like to use the computer to obtain up-to-date
    information useful in my daily life, e.g.: gas prices;
    prices on consumer goods, bus schedules, airfares etc.

135. I would feel comfortable sending and receiving
    correspondence using a computer terminal.

136. The computer increases isolation from other people.

137. To use the computer, you must conform to a rigid
    structure.

138. Computers can cause the loss of essential skills, e.g.: ability to add etc.

139. Given the opportunity, I would learn to program the
    computer.

140. If money were not an issue, I would buy a personal
    computer or terminal right now.

141. I feel computers are useful to me but I am scared to
    begin learning about them.

142. I would like to learn more about computers.

143. Working with a computer would probably make me feel
    uneasy or tense.

144. I feel helpless around a computer.
HOW I LEARN QUESTIONNAIRE

A - STRONGLY AGREE  B - AGREE  C - NEUTRAL
D - DISAGREE  E - STRONGLY DISAGREE

145. Computers sometimes scare me.
146. I would very much like to have my own computer.
147. I like the idea of taking computer courses.
148. I think I would enjoy using computers in my classes.
149. Walking through a room filled with computers would make me feel uneasy.
150. I feel uneasy when I am with people who are talking about computers.
151. I enjoy (or think I would enjoy) working with computers.
152. I feel confident about my ability to use computers.
153. It is my guess that I am NOT the kind of person who works well with computers.
154. On the whole, I can cope with computers as well as most others my own age.
155. Computers are gaining too much control over my life.
156. Every university student should know something about computers.
157. Every university student should be able to write a simple computer program.
158. Every university student should learn about the role that computers play in our society.
159. Computers can be useful in learning many subjects besides mathematics.
160. Computers are of little use in education.
This is a test to measure your knowledge of personal computers.
Please answer to the best of your ability by jotting down a brief answer in the space provided beneath each question. Thank-you.

1. What do the letters DOS stand for?

2. Why is the DOS program important to the operation of a personal computer?

3. How do you know when the computer is waiting for a command?

4. Briefly describe what a computer file is.

5. What are extension letters and why are they useful?

6. What is a "system transfer message"?

7. Why is it important to format a disk?

8. Give two examples of "wild card commands"?

9. Describe a situation where you would use an ahort command.
Appendix 4
Posttest

Here is another test of your knowledge about computers. Please jot your answers in the space provided beneath each question. Thank you.

1. How do you specify a certain disk drive on the IBM PC?

2. How is space on a computer disk measured?

3. What prompt indicates that the IBM PC is waiting for a command?

4. What message does the user receive when DOS has been loaded onto a disk?

5. How would you check to see how many files are stored on a disk?

6. What process must be undertaken to make a disk compatible with the IBM PC?

7. When using IBM PC-DOS, what does a "fault error message" indicate?

8. Give two reasons why you should make copies of a disk.

9. What part of a computer file is analogous to a table of contents in a book?
Appendix 6
Performance Tests

PERFORMANCE TEST 1 (TO BE COMPLETED AFTER UNIT C)

Name_________________________ ID Number_________________________

Materials: disk with text files on it.

The student is given a formatted disk containing several text files. The student is asked to do the following:
a) access the directory,
b) give the following information: when the file was created; how much space the file uses; how many files are on the disk.

Observer Checklist for test 1

Task 1) Load the disk:
1) time:
2) does the student respond to the date and day prompts (4e do they enter this information or press the enter key)? Y/N
3) does the student refer to the user's handbook? Y/N

Task 2) Access the directory:
1) time:
2) does the student use the DIR command? Y/N
3) does the student press the enter key? Y/N
4) does the student correctly identify: when the file was created; how much space the file uses; how many files are on the disk? Y/N
5) does the student refer to the user's handbook? Y/N
PERFORMANCE TEST 2 (to be completed after unit E)

Name ______________________ ID Number ________________

Materials: 2 disks containing text files, 1 blank disk.

The student will have access to a disk in the A drive (the A> prompt will be showing on the screen. A second disk will be located in the B drive.

a) ask the student to give the total amount of space used on the disk in the B drive.
b) ask the student how much space is available in 'K'.
c) ask the student how much space remains on the disk.

After the student has completed this task remove both disks and give the student a blank disk and ask them to prepare the disk so information can be stored on it. Then ask the student to reformat the disk with a copy of DOS on it.

Observer Checklist for test 2:

1) time:
2) does the student correctly change disk drives? Y/N
3) does the student use the CHKDISK command? Y/N
4) does the student press the enter key? Y/N
5) does the student correctly identify the total available space, the amount of space used (in 'K'); and the amount of space remaining? Y/N
6) does the student refer to the handbook? Y/N

2) time:
2) does the student use the FORMAT command? Y/N
3) does the student press the enter key? Y/N
4) does the student refer to the handbook? Y/N

3) time:
2) does the student use the FORMAT/S command? Y/N
3) does the student press the enter key? Y/N
4) does the student refer to the handbook? Y/N
PERFORMANCE TEST 3 (to be completed after unit G)

Name_________________________ ID Number_________________________

Materials: 1 disk containing text files, 1 blank disk.

The student is given a blank disk and a disk containing a
text file. The A> prompt is displayed on the screen.

a) ask the student to make a copy of the disk containing
the text files.
b) ask the student to ensure the copy is exact.
c) ask the student to make a backup copy of a single file on
the source disk.

Observer's Checklist for Test 3

Task 1

1) time:
2) does the student use the DISKCOPY A: B: command? Y/N
3) does the student press the enter key? Y/N
4) does the student insert the disk to be copied in the A drive?
   Y/N
5) does the student insert the blank disk in the B drive? Y/N
6) does the student press any key?
7) does the student press the 'N' key? Y/N
8) does the student refer to the user's handbook? Y/N

Task 2

1) time:
2) does the student use the DISKCOMP A: B: command? Y/N
3) does the student press the enter key? Y/N
4) does the student refer to the user's handbook? Y/N

Task 3

1) time:
2) does the student use the COPY A: B: command? Y/N
3) does the student press the enter key? Y/N
4) does the student check the directory to ensure the file has
   been copied? Y/N
5) does the student refer to the handbook? Y/N
PERFORMANCE TEST 4 (to be completed after unit K)

Name __________________________ ID Number __________________________

Materials: 1 disk containing text files.

The student is given a disk with a number of text files on it—the disk is booted and the A> prompt is showing on the screen.

a) ask the student to list the contents of a specific file.

b) ask the student to rename the file.

c) ask the student to rename all the files with the extension letters .ABC to .Y.

d) ask the student to erase a specific file and check the directory.

Observers Checklist

Task 1)

1) test:

2) does the student use the TYPE command? Y/N

3) does the student press the enter key? Y/N

4) does the student refer to the handbook? Y/N

Task 2)

1) time:

2) does the student use the REN command? Y/N

3) does the student press the enter key? Y/N

4) does the student check the directory to make sure the file is renamed? Y/N

5) does the student refer to the handbook? Y/N

Task 3)

1) time:

2) does the student use the REN* command? Y/N

3) does the student press the enter key? Y/N

4) does the student check the directory? Y/N

5) does the student refer to the handbook? Y/N

Task 4)

1) time:

2) does the student use the ERASE command? Y/N

3) does the student press the enter key? Y/N

4) does the student check the directory? Y/N

5) does the student refer to the handbook? Y/N
Appendix 7
Attitude Towards Computers (Post)

DIRECTIONS:
Indicate how much you AGREE or DISAGREE with each of the following statements by marking the appropriate letter on the answer sheet. Mark "a" if you STRONGLY DISAGREE with the statement. Mark "b" if you DISAGREE with the statement a little. Mark "c" if you are UNDECIDED about whether you agree or disagree with the statement. Mark "d" if you AGREE with the statement a little. Mark "e" if you STRONGLY AGREE with the statement.

As an example, if you AGREE a little that computers are noisy, then mark "d" on the answer sheet as shown below:

Computers are noisy  A  B  C  D  E

Or, if you are UNDECIDED about whether computers are noisy, mark "c" on the answer sheet as shown below:

Computers are noisy  A  B  C  D  E

If you have any questions, ask the test administrator.

1. I think that a home computer can be very interesting.
2. People managed before without computers, so computers are not really necessary now.
3. Mathematics is one of my best subjects.
4. I would like to learn how to use a computer.
5. When I hand in an essay, I feel I'm going to do poorly.
6. People who like computers are often not very sociable.
7. Computers are exciting.
8. I want to learn all I can about science.
9. I would not expect a good athlete to like computers.
10. Computers will never interest me.
11. I look forward to writing down my ideas.
12. I would be embarrassed to tell my friends that I would like to join a computer club.

13. If I don't see how to do a mathematics problem right away, I never get it.

14. If you like science you will like computers.

15. The world would be better off if computers were never invented.

16. I hope I have a job where I have to use science.

17. Working with computers is not my idea of fun.

18. Typing would be the biggest problem I would have in learning to use a home computer.

19. No matter how hard I try, I cannot understand mathematics.

20. Computers do not interest me.

21. You have to be smart to like computers.

22. I feel confident in my ability to clearly express my ideas in writing.

23. Computers are fun.

24. Microcomputers are easy to use.

25. I am proud of the work I do in mathematics.

26. If my family had a home computer, I would probably use it more than anyone else.
27. Females have as much ability as males when learning to use a computer.

28. I sometimes write stories at home even if they are not assigned for school.

29. Learning science is just as important for girls as for boys.

30. I am concerned that people might make computers too powerful in the future.

31. I would rather spend an evening doing something new with a computer than go out with my friends.

32. I never find myself thinking about science.

33. I enjoy working with computers.

34. Using a computer in math class would make math more fun.

35. It would be hard for me to learn how to program a computer.

36. Computers are boring.

37. Girls are just as good as boys in science.

38. I do not enjoy writing stories or essays.

39. I have studied about computers in school.

40. I have used a computer outside of school to play games.

41. I have used a computer outside of school to do something other than play games.
Appendix 8  
Attitude About Instruction

Here are a number of statements which describe the "ALL" training program you worked on. Please indicate whether you agree or disagree with the statement. Mark "a" if you strongly disagree; mark "b" if you disagree; mark "c" if you feel neutral; mark "d" if you agree and mark "e" if you strongly agree with the statement.

1) The program allowed me to practice the new skills I was being taught.

2) New concepts and skills were explained in a number of different ways.

3) I was able to work through the program at my own pace.

4) The program encouraged me to continue.

5) I felt I needed to know more about computers to work through the program successfully.

6) As I started the program, I quickly understood what I would learn.

7) The program made me feel that learning about "DOS" was important.

8) I knew quickly when I made a mistake.

9) There were times when I wanted to work more quickly or more slowly but the program wouldn't let me.

10) I felt the user's handbook was easy to follow.

11) I found the screen easy to read.

12) Material was presented on the screen in a confusing manner.

13) The instructions telling me how to use the program were adequate.

14) As I worked through the program, I felt I was unprepared.

15) When I made a mistake I immediately was told what I had done wrong.

16) I enjoyed learning from a computer.
17) There were times when I needed further explanation or required a demonstration.

18) I found the program readily suited the way I like to study and learn.

19) I looked forward to working through the 2nd section of the program.

20) During the first few frames, I knew that the material presented at the top of the screen were examples, while the material at the bottom contained specific instructions.

21) The many references to "ATI" training programs during the program annoyed me.

22) I wish I had more time to practice the computer skills presented in the program.

23) The program spent too much time going over the same skill or concept.

24) The program forced me to work more quickly than I wanted.

25) As I worked through the program, I had difficulty seeing how the material was relevant.

26) I had difficulty knowing when I had made an error.

27) As I started working through the program, I was not sure what points would be covered.

28) I found the instructions on how to use the program confusing.

29) I felt I could have learned the same material more easily from a textbook.

30) I would prefer to have the instructor demonstrate the skills rather than learn them from a computer.
References


Vita Auctoris
Robert Charles Pearson

Born: December, 20, 1960
Place: Deep River, Ontario, Canada.

High School: Mackenzie High.
             Deep River, Ontario, Canada.

University: B.F.A., York University.
            Toronto, Ontario, Canada.
Graduated: June, 1983