AN ABSTRACT OF THE THESIS OF

Ernest Giangrande Jr. for the degree of Master of Science
[in Computer Science]

Title: Augmenting Self-Study Materials With Microcomputer-Based Lessons: A Case Study

Redacted for Privacy

Abstract approved: ______________________

Abstract

An experiment was conducted at Oregon State University to evaluate the effectiveness of Computer Aided Instruction based lessons in a self study environment. The 47 student subjects were assigned to one of three groups: a CAI-EXP group which used a CAI lesson for teaching introductory FORTRAN formatted input and output; a PRT-EXP group which used a printed version of this CAI lesson; and a CONTROL group which used no supplemental materials. The criteria for evaluating the effectiveness of a particular condition was (1) the number of retakes on the quiz associated with lesson 4, (2) the number of I/O related questions answered

correctly on this quiz; and (3) the total score on this quiz. The results of the analysis for these measures showed that none of the subjects in the CAI-EXP group had to take a retest on this quiz. Furthermore, the CAI-EXP group significantly outperformed the other two groups on the other measures as well. The conclusion to be drawn is that the CAI materials had a major impact on the performance on the students exposed to them. These findings thus suggest that supplemental CAI materials can have a positive effect on student performance in a self study environment.

Augmenting Self-Study Materials With
Microcomputer-Based Lessons: A Case Study

by

Ernest Giangrande Jr.

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Completed June 12, 1981

Commencement June 1982

APPROVED:

Redacted for Privacy

Associate Professor of Computer Science

Redacted for Privacy

Chairman of the Department of Computer Science

Redacted for Privacy

Dean of Graduate School

Date thesis is presented June 12, 1981

Typed by Ernest Giangrande Jr. for Ernest Giangrande Jr.

TABLE OF CONTENTS

LIST OF FIGURES

Augmenting Self-Study Materials With Microcomputer-Based Lessons: A Case Study

I. Introduction

The effectiveness of a computer-based lesson in a self-paced learning environment is the central theme of this research project. Specifically it is concerned with the evaluation of a self-paced computer aided lesson designed to aid students learning the syntax and semantics of FORTRAN input and output statements and FORMAT statements. The role of the computer as a teaching medium and some studies conducted to evaluate its effectiveness will be discussed in the remainder of this section along with the specific focus and goals of the current study. The computer aided lesson developed for this study will be discussed in section II. A description of the subjects and procedures used for the study will be presented in section III. Section IV contains the results of the analysis carried out on the data collected and a discussion of these results will be given in section V.

A. The Computer as an Instructional Medium

Computer Aided Instruction (CAI) is a broad term that encompasses a variety of approaches to using the computer as an instructional medium. Figure 1 gives a list of some of these approaches. A particular lesson may incorporate one or more of these or some other approach depending on the author's purpose for developing the lesson. This diversity suggests that a lesson's description must be examined or the lesson must be viewed before its content can be determined. The lesson developed for this study incorporates review, simulation, and problem solving.

CAI Approaches:

1. Drill and Practice
2. Gaming
3. Review
4. Dialog
5. Tutorial
6. Simulation
7. Enquiry
8. Problem Solving
9. Experimental
10. Quiz or Testing

Figure 1.

From a student's perspective there are a number of advantages gained through the use of CAI lessons. Among these are:

1) a student may proceed through a lesson at his own pace
2) the system can present repeated review and/or problems for the student to solve
3) a student is made an active participant in the learning process (Pratt and Davis, 1981).

The user of a CAI lesson must respond to every problem posed by the system. This active participation is lacking in traditional classroom environments where a problem is presented and once one student has given the correct answer the lesson proceeds. In such an environment some students benefit by solving problems but others may sit idly by, not attempting to solve the problems at all. Some students may become frustrated because they are never able to produce the answer before the quicker students (Hazen et al, 1979).

The use of simulation and experimentation can enhance a student's understanding of a topic by allowing him to see the results of manipulations that are either dangerous or take too long to observe in the real world. That is, in an experimental lesson the student may manipulate various variables and see the result of such manipulation almost immediately thereby reinforcing the effects of the manipulation in a dynamic way.

Lessons can also be enhanced through the use of graphics, sound, and video presentations (Bailey, 1979; Kehrberg, 1979; Pollack, 1978). These can be used along with textual presentations. These features are available in many microcomputers and may lead to a more extensive use of them in CAI environments. Touch panels and graphics tablets are also available in some systems and allow a student to interact with a lesson without the need to type responses at a keyboard. Kehrberg (1979) suggests that the use of such enhancements make lessons more interesting to a student.

A lesson can be structured so that it presents the goals of the lesson to a user and a means by which he can determine immediately whether he has attained these goals. The lesson can use problems and feedback regarding the student's solution to them. The system can also be designed to direct the student to topics that should be reviewed by him. Some researchers in the field of Artificial Intellegence are attempting to build systems that take a more active role in the learning process (Bregar and Farley, 1980; Brown and Burton, 1974). A more traditional CAI approach is taken in the lesson developed for this study.

In this discussion it has been assumed that the instructor will develop the lessons used by his students. This need not be the case. Many lessons have been developed for a wide range of disciplines and are available commercially (Hoye and Wang, 1973; CONDUIT). The CDC PLATO system is also a source of many lessons in a variety of areas. This system uses specially designed terminals linked by telephone lines to one of the CDC PLATO installations.

The main focus of this study is the effectiveness of CAI in teaching computer science concepts. There have been some attempts at developing lessons in this area. Slaughter and Mortanelli (1975) have developed a lesson that addresses FORTRAN subroutines. This lesson includes the use of the CALL statement, parameter passing, and a simulation of a main program and three subroutines. A series of three lessons developed by Padua et al (1975) introduce the APL programming language. These lessons concentrate on an introduction to APL, scalar arithmetic using APL, and vector arithmetic using APL. A lesson that deals with what data structures are, how they are implemented, and when they are used was developed by Benuenesti and Friedman (1975). This lesson introduces these topics using PL/1. Shapiro (1974) has also developed a lesson that covers the topic of data structures. This lesson introduces stacks and their manipulation, the evaluation of expressions using postfix notation, and the structure and manipulation of linked lists.

One might ask why the use of CAI has not had a greater impact on education than it has had. Nievergelt (1980) points out that the cost of developing portable software has been underestimated. It may take between 100 and 200 man hours to produce a one hour CAI lesson. He also suggests that the type of hardware required to produce effective lessons was also underestimated, with some of the original CAI project developers not realizing the power of the use of graphics, etc. Splittgerber (1979) reports that the lack of qualified authors has also contributed to this problem. Jay (1981), and Chambers and Sprecher (1980) suggest that some professionals not trained in computer science may fear the computer. Jay points out that computerphobia is a subarea of a more general technophobia experienced by some people. He defines this as a negative attitude or anxiety toward the computer and its use. The uncertain results reported by researchers attempting to evaluate the effectiveness of this medium may also contribute to the lack of enthusiasm from some educators. This last point will be expanded below.

B. Focus and Goals

There are a limited number of published reports addressing the effectiveness of CAI and those that are available have reported uncertain results. Bailey and Klassin (1979) report "CAI can be used in some situations to improve achievement scores, particularly with disadvantaged students...". These authors report that the results of a CAI program implemented at four correctional institutions failed to show substantial improvement in terms of performance on achievement scores. They do, however, report that the program was continued because CAI:

1) motivated otherwise unmotivated learners
2) improved problem solving skills
3) provides repetitive drill and practice for low ability students.

Splittgerber (1979), on the other hand, reports on the empirical study of thirty-three CAI programs and concludes that this approach is more successful using the tutorial and drill and practice techniques than when using the problem solving, simulation, and gaming techniques. He also reports that retention is lower for CAI groups and that CAI is most effective for low ability students. Edwards (1978) reports CAI groups performing significantly higher on lesson post tests in a study of a freshman mathematics course. He also reports that students exposed to CAI based materials had a higher positive attitude toward the computer. This is also reported by Crawford (1970) as cited by Edwards.

In a study of a FORTRAN Programming for Business class Hazen (1979) reports that those students using CAI materials spent less time on the course but that there was no significant difference on the final examination scores for the CAI group and one that attended traditional lectures. He further reports no significant attitude difference toward the computer between these two groups. Chambers and Sprecher (1980) after reviewing a number of adjunct CAI studies (that is, supplementing traditional classroom instruction) concluded that low aptitude students using CAI materials performed better than either average or high aptitude students. They also report lower retention in the CAI groups regardless of aptitude level.

One aspect that stands out in the above studies is that they compare CAI based groups with groups in traditional classroom environments. However, CAI is intrinsically a self-paced learning medium, thus a study comparing CAI with a self-paced course might demonstrate its usefulness as an instructional medium. This thesis is concerned with the development of a study to test the hypothesis that CAI is a provably effective medium compared to its self study non CAI counterpart. A self study course (CS190, Self Study Introduction to FORTRAN Programming) at Oregon State University was selected and CAI materials were developed to teach the programming of input and output statements using FORMAT statements. A study was conducted to evaluate the effectiveness of this lesson. The design of this study is a traditional experimental design. The null hypothesis can be stated as follows: there is no improvement in student performance due to the use of supplemental CAI based materials in a self study environment. Performance will be measured in the following ways:

1) the number of attempts on a quiz over the subject matter

2) the number of correct responses to fifteen I/O related questions on the quiz

3) the total score on the quiz.

This hypothesis will be rejected if it is found that the group exposed to the CAI materials perform significantly better than the other groups on these measures. A significance level of .05 must be demonstrated before the null hypothesis will be rejected. Before describing this study in detail a discussion of the structure of the self study course and the CAI lesson that was developed will be presented.

C. CS190 A Self-paced Programming Course

CS190, Self Study Introduction to FORTRAN Programming is a self-paced course. The students enrolled in this course meet with the instructor one time for an orientation. They then work independently, dealing with the clerks in the Math Science Learning Center (MSLC) where they take required quizzes, pick up and turn in programming assignments, and check on their scores and progress in the course. The instructor and student consultants are available for questions on either the quizzes or the programming assignments. The students use a set of course notes designed for this course and a text book (Kreitzberg and Schneiderman, 1975).

The course consists of nine lessons. Each lesson is outlined in the course notes which contains learning objectives, directions to readings in the text, sample problems, and self-administered quizzes. After completing these materials for a lesson the student takes a quiz for that lesson. To pass, a student must obtain a grade of at least 70%. Retakes of the quiz are allowed with no restriction until a passing grade is obtained. The student must also complete six programming assignments which are progressively more difficult. The final assignment requires the student to sort a multidimensional array and merge it with another pre-sorted array. This must be done using subroutines.

The course is graded on a Pass/No Pass basis. To receive a Pass grade the student must pass all nine quizzes and complete all six programs. They may retake the quizzes and redo the programs until they are acceptable. The evaluation of the student is based on performance in all of the areas covered by the course. The percentage of students successfully completing this course has ranged between 50 and 60 percent.

An examination of the grade records maintained for all students enrolled in CS190 indicated that the number of students who actually completed this course successfully with a Pass grade was low. Further examination showed that there are a few key areas where the number of retakes for a quiz was extremely high. Clearly, lessons with a high number of retakes contained material that was difficult for the students to master. In an attempt to isolate what material was causing these problems the grade cards and individual quizzes were studied in greater detail.

First, the course grade cards were examined to determine which lessons had the highest percentage of retakes. The percentage of retakes is considered a more appropriate measure of difficulty than the raw number of retakes because of the high attrition rate. That is, a particular quiz may have a small number of retakes as compared to another quiz but the number of students taking the quiz for the first time may also be lower. This is especially true of later lessons.

Figure 2 shows the results of this examination. Quizzes 4 and 8 have the highest number of retakes with quizzes 6 and 5 following close behind. Figure 2 also illustrates the overall attrition rate for this course.

Quiz	# Taking test	# Taking retest	% Taking retest
1	92	5	5.43
2	86	16	18.60
3	83	18	21.68
4	75	32	42.67
5	67	21	31.34
6	64	24	37.50
7	60	15	25.00
8	57	24	42.11
9	55	3	5.45

Figure 2. Breakdown of Quiz Retakes.

Next the quizzes themselves were examined to determine what material was responsible for the high number of retakes by some students. Figure 2 suggests that the primary areas of interest would be the material covered by quizzes 4, 5, 6, and 8. Since a given lessson does not necessarily cover one topic, each of these quizzes was examined to determine the topics it covered. Those topics which contributed to the retakes for these four quizzes were isolated. Figure 3 lists the topics that resulted in the largest number of incorrect responses by the students on these four quizzes. The original attempt and all subsequent retakes for these quizzes were examined to isolate these problem areas. Figure 3 shows that the problems fall into the area of Input/Output statements and that of looping.

Quiz #4
- -determining values to be stored in memory given data and format specifications
- -determining items to be output given data values stored in memory and format specifications
- -writing READ and PRINT statements that use format statements

Quiz #5
- -correcting, evaluating, and writing logical IF statements
- -writing READ and WRITE statements using device numbers and format statements
- -writing the code for simple loops

Quiz #6
- -comparing segments of code containing DO loops
- -evaluating execution of segments of code containing DO loops
- -correcting incorrect DO statements
- -correcting code in a DO loop
- -determining the number of times a segment of code containing a DO loop is executed

Quiz #8
- -correcting I/O statements that contain implied DO loops
- -repetition and grouping in FORMAT statements
- -writing I/O statements with implied DO loops
- -determining the characteristics of various format specifications (ie. E,F,/)
- -writing FORMAT statements

Figure 3. Problem areas for quizzes 4, 5, 6, and 8.

From this examination it was decided to address the difficulty that students appeared to be having with Input/Output statements. It was further decided to concentrate on lesson 4 as this is the first introduction to formatted input and output, and because approximately 40 percent of the responses to the Input/Output questions on this quiz were incorrect. These questions range from evaluating the result of the execution of format specifications to producing READ, PRINT, and FORMAT statements from given specifications. The I, F, X, and string specifications are introduced in this lesson.

Auxiliary CAI materials were prepared to teach the basic concepts of input and output programming. A detailed description of this lesson will be given below. Features of this supplemental material include:

1) a step by step examination of how input and output statements are processed by a computer
2) practice on the part of the student. This is considered important in developing the skills and insights required to construct input and output statements and their associated FORMAT statements.
3) immediate feedback regarding the correctness of the student's responses to problems (a feature that is virtually nonexistent in a self-paced course).

II. Lesson Development

Two supplemental lessons were developed for this study, one is a CAI lesson and the other is a printed lesson. Both forms were developed so that any effect produced by the use of the CAI supplemental materials could be attributed to this medium as opposed to simply some form of supplement. The following sections contain a discussion of the implementation of the CAI lesson on a microcomputer, a description of the CAI lesson itself, and a description of the printed version of this lesson.

A. Microcomputers in a CAI Environment

The CAI lesson itself was developed on an Apple II microcomputer. This machine presents several advantages in an educational environment. First, it is fairly inexpensive (about $3000 per unit). This includes the microcomputer, disk drives, monitor, and associated systems software. A more fundamental advantage in the development of CAI materials is the ease with which one can program graphics as well as textual presentations on this system. The Apple II uses an adaptation of turtle graphics (Abelson et al, 1974), which allows the user to plot lines and arcs using standard polar coordinates.

The graphic features are incorporated into a package of procedures that can be called from a user's program. The lesson was developed using Apple Pascal, a version of UCSD Pascal (Bowles, 1977). The string processing facilities are helpful in preparing text to be displayed on the graphics screen. Unfortunately the presentation of text in graphics mode is much slower than in text mode. This is because all characters must be displayed by system supplied procedures that retrieve the characters from a system file and draw them on the graphics screen.

Although there are substantial memory limitations (48K maximum) the system does provide for the compilation of "segmented" procedures (overlays) which greatly increase the effective size of the programs that can be run. Each segmented procedure must contain all global variable declarations and a template for all global and segmented procedures. Each such segment is then 'filled in' and compiled. All the segments must then be linked to produce an executable code file. Minor changes in the global declarations and/or structure of the program results in all segments being changed, recompiled ,and relinked. This process is slow enough that the compilation of a large segment may take several minutes. It should be noted that a later release of this system has implemented a feature that allows a user to chain separate programs together which may reduce the need for using segmented procedures. Also the compilation time has been reduced in this new release.

B. A CAI Lesson for FORTRAN I/O

In lesson 4 of the self study course formatted input and output is introduced and explained. There are other concepts that must be clear to a student attempting to understand input and output. In particular these concepts include:

1) the relationship between a variable name and a location in the computer's memory
2) the relationship between the name chosen for a variable and the type of data that is stored in its memory location
3) the difference between the data types that can be processed using a FORTRAN program

Each of these topics is included in this lesson to give a student a solid basis for attempting the construction of input and output statements and their associated FORMAT statements.

The lesson is separated into five parts, each part either contains textual material and examples, or problems that give the student an opportunity to practice the techniques covered in this lesson. This practice is considered to be important in a student's comprehension of the material being presented. Just presenting the various techniques and rules allows a student to passively view the material while the inclusion of problems requires him to take a more active role in the process. Figure 4 gives the names of each of the five parts of this lesson along with a brief description of each. Figure 5 presents the top level pseudo code for the lesson which will be referred to by line numbers in the discussion to follow.

The READ/PRINT-Construction part (figure 5 lines 2-7) begins with a discussion of the data types REAL and INTEGER followed by an explanation of the correspondence between variable names and locations in a computer's memory. The FORTRAN variable naming conventions in relation to these two data types are discussed as well as the rules for forming variable names used in FORTRAN programs. Finally the rules for constructing both READ and PRINT statements are presented along with examples of each. At this point the student is given an opportunity to review this material.

1) <u>READ/PRINT-Construction</u> :
   - -Basic concepts related to input and output
   - -Rules for constructing READ and PRINT statements
   - -Examples of READ and PRINT statements

2) <u>READ/PRINT-Practice</u> :
   - -Student constructs READ and PRINT statements from specifications
   - -System evaluates student's response

3) <u>FORMAT-Construction</u> :
   - -Rules for constructing FORMAT statements used for input
   - -Examples of FORMAT statements used for input
   - -Rules for constructing FORMAT statements used for output
   - -Examples of FORMAT statements used for output

4) <u>READ/FORMAT-Practice</u> :
   - -Student constructs FORMAT statements used for input from specifications
   - -System evaluates student's response

5) <u>PRINT/FORMAT-Practice</u> :
   - -Student constructs FORMAT statements used for output from specifications
   - -System evaluates student's response

<u>Figure 4.</u> Lesson 4 part names and their descriptions.

When the student is satisfied that he understands the material he proceeds to READ/PRINT-Practice (figure 5 lines 8-17) where he is given an opportunity to construct READ and PRINT statements from specifications presented in the form of problems. Figure 6 gives an example of the form in which these problems are presented. Figure 6a shows the specifications that a student must use to construct a READ statement and figure

```
BEGIN (*lesson*)
  REPEAT
    present numeric data types;
    present correspondence between variable
      name and memory location;
    present variable naming conventions;
    present construction of READ and
      PRINT statements;
  UNTIL (student wants to continue);

  REPEAT
    present specifications for construction of
      READ  or PRINT statement;
    (* student enters READ or PRINT statement*)
    evaluate response;
    IF (response correct)
      THEN simulate execution of statement
        ELSE  explain  errors  and  show  correct
                 response;
  UNTIL ((student gets 5 correct OR 10 incorrect)
          AND
         (student wants to end this section));

  IF (student got 10 incorrect AND wants  to  end
         session)
    THEN stop session;

  REPEAT
    present discussion of format specifications;
     present construction of READ and FORMAT sta-
       tements;
     simulate  execution  of READ and FORMAT sta-
        tements;
    present construction of PRINT and FORMAT sta-
       tements
      with discussion of string specification;
    simulate execution of PRINT and  FORMAT  sta-
        tements;
  UNTIL (student wishes to continue);

  REPEAT
     present  specifications  for  constructing
      FORMAT statement for input;
    (* student enters response *)
    evaluate response;
    IF (response correct)
    THEN simulate execution of READ  and  FORMAT
              statements
```

Figure 5 (1 of 2). High level pseudo code for lesson 4.

```
     ELSE BEGIN
            IF (response syntactically correct)
              THEN BEGIN
                      IF  (response  semantically
                            incorrect)
                      THEN BEGIN
                              simulate  execution
                                of students
                                 response;
                                present   correct
                                  response and
                                  correct results;
                          END
                  END
             ELSE present correct response;
   UNTIL ((student gets 5 correct OR 10 incorrect)
          AND
         (student wants to end this section));

   IF (student got 10 incorrect AND wants to  stop
       lesson)
    THEN stop session;

   REPEAT
    present specifications for FORMAT for output;
    (* student enters response *)
    IF (response correct)
       THEN simulate execution of PRINT and FORMAT
             statements
      ELSE BEGIN
             IF (response syntactically correct)
               THEN BEGIN
                      IF  (response  semantically
                             incorrect)
                       THEN BEGIN
                              simulate execution
                                of students
                                response;
                               present   correct
                                response and
                                correct results;
                          END
                  END
             ELSE present correct response;
   UNTIL ((student gets 5 correct OR 10 incorrect)
           AND
        (student wants to end this section));
 END (*lesson*).
```

<u>Figure 5 (2 of 2).</u> High level pseudo code for lesson 4.

```
106 FORMAT(                              )
---------------------------------------
```

CONSTRUCT AN INPUT STATEMENT THAT WILL READ:

AN INTEGER VARIABLE FOLLOWED BY
A REAL VARIABLE FOLLOWED BY
A REAL VARIABLE

* NOTE- USE PROPER SPACING AND CHOOSE APPROPRIATE VARIABLE NAMES AND FORMAT NUMBER

(a)

```
106 FORMAT(                              )
---------------------------------------
```

CONSTRUCT AN OUTPUT STATEMENT THAT WILL PRINT:

AN INTEGER VARIABLE FOLLOWED BY
A REAL VARIABLE FOLLOWED BY
A REAL VARIABLE

* NOTE- USE PROPER SPACING AND CHOOSE APPROPRIATE VARIABLE NAMES AND FORMAT NUMBER

(b)

Figure 6. Sample Problem Specifications for READ and PRINT Statements.

6b shows those to be used to construct a PRINT statement. The student enters his response through the keyboard and it is displayed on the screen. If the response is incorrect the errors are pointed out to the student and a correct response is displayed. If the response is correct the system simulates the execution of the statement using either (1) a card image with data values for a READ statement or (2) values assumed to be stored in memory in the case of a PRINT statement. Appendix A gives a detailed sample of how this is presented to a student.

The READ/PRINT-Practice part is contained within a loop that is terminated when a student has either produced five correct or ten incorrect responses to the problems. If the student makes five correct responses he may proceed or he may opt to try more problems. If ten incorrect responses are made (before five correct ones) the student is given the choice of terminating the lesson or continuing in an attempt to produce correct READ and PRINT statements. Five correct responses would appear to indicate that the student has some mastery of the material, but he can continue practicing at his own option. However, after ten incorrect responses a student could become frustrated, therefore, he may terminate at that time and seek additional help or review the material. If, on the other hand, he feels he understands the material but has been making silly mistakes he may choose to continue in this part. This same approach is taken in the READ/FORMAT-Practice and PRINT/FORMAT-Practice parts discussed below. It should be noted that the system randomly produces all specifications and data values used in these problems. A procedure generates all statements using the syntax rules for their construction. The type and number of specifications are determined randomly but there is no template for the statements that is filled in by the system. This approach differs from generative techniques described by Collins and Duff (1979) and Garcia and Rude (1979) who use a template with particular variables assigned values for each problem. Both approaches avoid the necessity of maintaining a large pool of questions. In this way it is unlikely that any two sessions would be the same. This allows a student to view the lesson any number of times, or to do as many problems as is necessary in one lesson, without becoming familiar with the problems, and perhaps memorizing the correct responses. Thus, a student must use the skills he has learned in solving each of the problems.

Upon completing the READ/PRINT-Practice part the student proceeds to the FORMAT/Construction part (figure 5 lines 18-24). Here a discussion of the format specifications to be used in constructing FORMAT statements associated with READ and PRINT statements is presented. The F, I, X, and string specifications are included in this part. The particular use of the F specification as it applies to both the READ and PRINT statements is also discussed. Simulated executions of both a formatted READ and formatted PRINT statement are also generated in this section. As in the READ/PRINT-Construction part, the student may review the material until he feels comfortable with it before he goes on to the READ/FORMAT-Practice part.

The READ/FORMAT-Practice part (figure 5 lines 25-43) and the PRINT/FORMAT-Practice part (figure 5 lines 44-59) give the student an opportunity to construct FORMAT statements for READ and PRINT statements respectively. For example, the specifications needed to construct a FORMAT statement for an input statement are presented. These consist of the variable name and the columns in which its value will be found on the input data card. Figure 7a shows what the student must use in constructing these FORMAT statements. The specifications for constructing a FORMAT statement used with a PRINT

```
READ 111,K,G
```

CONSTRUCT A FORMAT STATEMENT THAT COULD BE USED TO INPUT THE FOLLOWING ITEMS:

VARIABLE NAME	CARD COLUMNS	DECIMAL PLACES
K	1- 2	
G	5-10	1

(a)

```
PRINT 111,K,G
```

CONSTRUCT A FORMAT STATEMENT THAT COULD BE USED TO OUTPUT THE FOLLOWING ITEMS:

ITEM	LINE POSITIONS	DECIMAL PLACES
K	2- 5	
' ANS= '	8-13	
G	15-18	2

(b)

Figure 7. Sample Problem Specifications for FORMAT statements for READ and PRINT statements.

statement are shown in figure 7b. These are used in the PRINT/FORMAT-Practice part and include the current values for the variables to be output, the columns where the value is to be printed on the output line, and any strings that are to be output along with the data items.

The student's response in both of these parts is evaluated in two steps. First, the syntactic form of the response is evaluated. If any errors are detected they are pointed out to the student. If the statement contains no syntactic errors it is evaluated to determine whether it is semantically correct based on the given specifications. This can result in two outcomes (1) the response is semantically correct and will produce the specified result, or (2) the response is semantically incorrect and will produce some undesirable result. In the case of an input statement incorrect data values would be read from the input data card. For an output statement the result would be an incorrect output line.

As the system randomly generates the specifications to be presented to a student it stores their characteristics (type, field width, etc.) in a linked list. The elements of this list are used in constructing the display presented to a student who enters a FORMAT statement based upon these specifications. The student's response is held in an array which is scanned to determine whether it is syntactically correct. At the same time the system 'breaks' up the FORMAT statement and stores the characteristics of each of its component specifications in another linked list. If the student's response is found to be syntactically correct the system compares the elements of these two linked lists to determine whether the students response is semantically correct. If the response is semantically incorrect the system uses the linked list produced from the students response to simulate the execution of his response. This demonstrates the effect of an incorrect specification. The system generated list is used to display a correct FORMAT statement so the student can compare the two.

If the evaluation finds that the response is correct the student is so informed and a simulated execution of the statements is presented by the system. If the response is found to be semantically incorrect the student is informed that its execution may produce an undesired result. A simulated execution of his response will then be presented by the system to demonstrate what result an erroneous FORMAT statement would produce. This is followed by the display of a correct FORMAT statement and the expected result for the student to use in determining what he has done wrong. Thus, the student sees graphically what result an incorrect format specification would produce coupled with its effect on the rest of the items being read or written. For example, in the case of an input statement not only will an incorrect specification result in an erroneous value being stored in its associated variable but the remainder of the variables being read using that FORMAT statement will also have incorrect values. Presenting the correct and incorrect FORMAT statements along with the result produced by each gives the student an opportunity to examine them and decide what he has overlooked.

As discussed above in regard to the READ/PRINT-Practice part, both the READ/FORMAT-Practice part and the PRINT/FORMAT-Practice part are contained within loops that terminate either when the student has given five correct responses and wishes to stop that part or when he has given ten incorrect responses and wishes to stop the lesson. The program terminates at the end of the PRINT/FORMAT-Practice part.

C. A Printed Lesson for FORTRAN I/O

To ensure that any observed effectiveness of the CAI based materials was, in fact, more likely a function of CAI and not the content of the lesson, a control in the form of a printed lesson has also been developed. This lesson consists of all the textual material presented to the student in the CAI version but it is in a text form. Included in the lesson are all of the discussions and examples outlined above with the exception of the problems that are presented and evaluated in the CAI version. Students using this version would not be presented problems to solve and would not have an opportunity to practice the skills presented in the lesson. Obviously, there would be no feedback regarding their comprehension of the material in this version. As discussed above, both of these aspects are important for a student trying to learn this material.

III. Methodology

The study was conducted using a traditional experimental design consisting of three groups: Group 1 (CAI-EXP) an experimental group that viewed the CAI lesson , Group 2 (PRT-EXP) an experimental group that used the printed lesson, and Group 3 (CONTROL) a control group which used no supplemental materials.

A. Subjects

Subjects for this study were drawn from the students enrolled in CS190, Self Study Introduction to FORTRAN Programming during the winter term of 1981 at Oregon State University. During the orientation meeting the students were asked to volunteer to help in a project that they were told was designed to evaluate their impressions of some new materials designed for this course. The students filled out a slip designating whether they would or would not volunteer. Twenty volunteers were then randomly selected and assigned to the CAI-EXP and PRT-EXP groups. The remaining volunteers were assigned to the CONTROL group. Since twenty volunteers did not remain for this group it was supplemented by randomly choosing students enrolled in the course who had not volunteered for the project.

The students were told to contact the instructor after completing all of the standard materials for this lesson but prior to taking quiz 4. At that time each student was informed of the group he was in and how to access the appropriate materials. He was also told that he would be given a questionnaire to complete regarding the lesson he viewed. Those volunteers who were assigned to the CONTROL group did not use any supplemental material and did not complete a questionnaire and so were told that their assistance was not needed because there were too many volunteers. They were told that they should just continue on with the course.

B. Procedure

Upon contacting the instructor each student was informed of the type of material he would be viewing and told how he could access it. If he was assigned to the CAI-EXP group, arrangements were made for him to meet with an assistant who directed him to the microcomputer lab and stayed with him until he completed the lesson. The assistant gave him the lesson disk along with printed instructions on how to use it and then went about his own work being available if the student had any questions regarding the start up of the lesson. When the student completed the lesson the assistant gave him a questionnaire to fill out. The student filled in the form and left.

Those students assigned to the PRT-EXP group were directed to the resource desk of the Math Science Learning Center at Oregon State University where they received the printed lesson and a questionnaire, returning both on completion. As mentioned above, those volunteers assigned to the CONTROL group were told they were not required to participate in the study and those who had not volunteered never contacted the instructor.

Since one motivation for this research was the high number of students who had to take retakes on some of the quizzes an analysis was first directed at the number of retakes on quiz 4 for the groups and then the scores of these quizzes were analyzed. Two measures regarding the scores were examined, (1) the total score for the quiz which can range from 0 to 100, and (2) the number of questions answered correctly out of the fifteen I/O related questions on the quiz. Also analyzed were some of the questions from the questionnaire and some information from the class enrollment list that was used in concluding that the groups were homogeneous.

It should be noted that the number of students dropped from the original twenty per group. This was caused by two factors. First, only those students who remained active in the course through lesson 4, having taken the quiz at least once, were considered. There were also a few students in the CAI-EXP group and the PRT-EXP group who later asked to be excused from participating in the study. These students were dropped from the study as they had not been exposed to the treatment before they took the quiz for this lesson. The groups eventually ended up with 14 students in the CAI-EXP group, 16 in the PRT-EXP group, and 17 in the CONTROL group. A discussion of the results of the analysis carried out will be given in the next section. Note that all of the analyses were carried out using SPSS: A Statistical Package for the Social Sciences (Nie, 1975).

IV. Results

In analyzing the data collected for this study three statistics were used: the analysis of variance (Anova), the chi square, and the Pearson's correlation. In the analysis of variance test two estimates of the population's variance are computed, one is the variance around the mean within the groups and the other is the variance between the group means. If these two estimates are about the same the null hypothesis should not be rejected. The F value is the ratio obtained by dividing the between group mean square by the within group mean square. The analysis of variance will also be used as an appropriate follow up test to isolate any variance found between the groups (Wood, 1977). In this and all other analyses a $p<.05$ will be required before the null hypothesis will be rejected. The chi square test is used to assess whether obtained frequencies in the various categories differ significantly from the expected frequencies for these categories. A Pearson's correlation determines the degree to which two variables are linearly related. Two measures are taken for each subject, one on each of the variables of interest, and the correlation is computed.

A chi square of the number of tries at quiz 4 by group showed that there is a significant difference ($p<.05$) in the number of tries between the groups. Figure 8 shows that no member of the CAI-EXP group took a retake on this quiz while 50% of the PRT-EXP group and 76.5% of the CONTROL group took one or more retakes.

Chi Square

Group / Try4	CAI-EXP	PRT-EXP	CONTROL	
1	14	8	4	26
2	0	7	9	16
3	0	1	3	4
5	0	0	1	1
Total	14	16	17	

Raw chi square = 19.95191 With 6 Degrees of Freedom. Significance = .0028

Figure 8. Chi square for number of tries at quiz four for all groups.

Analysis of Variance

	D.F.	Sum of Squares	Mean Squares	F Ratio	F Prob.
Between Groups	2	111.4027	55.7013	10.660	.0002
Within Groups	44	229.9165	5.2254		
Total	46	341.3191			

Figure 9. Anova for I/O related questions on quiz 4 for all groups.

An analysis of variance was carried out for the number of correct responses to the I/O questions on the first attempt at quiz 4 by the members of the three groups. Figure 9 shows the results of this analysis. The F ratio of 10.66 ($p<.05$) suggests that there is some variance between the groups on this measure. The analysis of variance was used as a follow up test to isolate this variance. Figure 10a shows a significant difference ($F = 12.401$, $p<.05$) between the CAI-EXP group and the PRT-EXP group on this measure and figure 10b also shows a significant difference ($F = 20.829, p<.05$) on this measure between the CAI-EXP group and the CONTROL group. Since there is no difference ($F = 1.538, p<.05$) between the PRT-EXP group and the CONTROL group on this measure and since figure 11a shows that the mean score for the CAI-EXP group is much higher than the mean scores for

Analysis of Variance

	D.F.	Sum of Squares	Mean Squares	F Ratio	F Prob.
Between Groups	1	53.2149	53.2149	12.401	.0015
Within Groups	28	120.1518	4.2911		
Total	29	173.3667			

(a)

Analysis of Variance

	D.F.	Sum of Squares	Mean Squares	F Ratio	F Prob.
Between Groups	1	107.3597	107.3597	20.829	.0001
Within Groups	29	149.4790	5.1544		
Total	30	256.8387			

(b)

Analysis of Variance

	D.F.	Sum of Squares	Mean Squares	F Ratio	F Prob.
Between Groups	1	9.4342	9.4342	1.538	.2243
Within Groups	31	190.2022	6.1356		
Total	32	199.6364			

(c)

Figure 10. Follow up tests for I/O related questions on quiz 4: (a) Anova for CAI-EXP and PRT-EXP, (b) Anova for CAI-EXP and CONTROL, and (c) PRT-EXP and CONTROL.

either of the other groups this variance can be attributed to the effect from the CAI-EXP groups performance.

Breakdown of Means

I/O Responses for Quiz #4 :

Group	Count	Mean
CAI-EXP	14	11.8571
PRT-EXP	16	9.1875
CONTROL	17	8.1176
Total	47	9.5957

(a)

Total Scores for Quiz #4:

Group	Count	Mean
CAI-EXP	14	83.0714
PRT-EXP	16	68.3750
CONTROL	17	66.4706
Total	47	72.0638

(b)

Figure 11. Breakdown of means for I/O questions and total scores.

Analysis of Variance

	D.F.	Sum of Squares	Mean Squares	F Ratio	F Prob.
Between Groups	2	2445.8946	1222.9473	9.600	.0003
Within Groups	44	5604.9139	127.3844		
Total	46	8050.8085			

Figure 12. Anova for total scores on quiz 4 for all groups.

A similar analysis was applied to the total scores for the first attempts at quiz 4 by the students in the three groups. Again an analysis of variance was applied to these scores. Figure 12 shows an F ratio of 9.600 ($p<.05$) suggesting that some variance exists between the groups on this measure also. The results of follow up tests used to isolate this variance are shown in figure 13. Figure 13a shows a significant difference ($F = 14.332$, $p<.05$) between the CAI-EXP group and the PRT-EXP group. There is also a significant difference ($F = 17.475, p<.05$) between the CAI-EXP group and the CONTROL group shown in figure 13b. An F ratio of 0.204 ($p>.05$) from figure 13c shows no significant difference between the PRT-EXP group and the CONTROL group. These analyses and the mean total scores reported in figure 11b suggest

Analysis of Variance

	D.F.	Sum of Squares	Mean Squares	F Ratio	F Prob.
Between Groups	1	1612.6881	1612.6881	14.332	.0007
Within Groups	28	3150.6786	112.5242		
Total	29	4763.3667			

(a)

Analysis of Variance

	D.F.	Sum of Squares	Mean Squares	F Ratio	F Prob.
Between Groups	1	2115.8039	2115.8039	17.475	.0002
Within Groups	29	3511.1639	121.0746		
Total	30	5626.9677			

(b)

Analysis of Variance

	D.F.	Sum of Squares	Mean Squares	F Ratio	F Prob.
Between Groups	1	29.8935	29.8935	.204	.6548
Within Groups	31	4547.9853	146.7092		
Total	32	4577.8788			

(c)

Figure 13. Follow up tests for total scores on quiz 4: (a) CAI-EXP and PRT-EXP, (b) CAI-EXP and CONTROL, and (c) PRT-EXP and CONTROL.

that the variance can again be attributed to the CAI-EXP groups performance.

Analysis of Variance

	D.F.	Sum of Squares	Mean Squares	F Ratio	F Prob.
Between Groups	1	.0383	.0383	.222	.6428
Within Groups	20	3.4575	.1729		
Total	21	3.4958			

Figure 14. Anova for G.P.A. for CAI-EXP and PRT-EXP

In an attempt to demonstrate that these groups were homogeneous an analysis was applied to two of the questions from the questionnaire, the students G.P.A. and their prior experience with computers. These tests could only be applied to the CAI-EXP and PRT-EXP groups as they were the only ones who filled out the questionnaires. An analysis of variance was applied to the G.P.A.'s for the members of these groups. Figure 14 shows that there is no significant difference between these two groups on this measure ($F = 0.222$, $p>.05$). A chi square test of these groups by prior computer experience shows no difference between these two groups as shown in figure 15. The only measure that could be used to confirm the homogeneity of all three groups was the students classification (ie. freshman, sophomore, etc.). Figure 16 is a chi square of the groups by classification. With a chi square = 10.59305 (p>.05) it suggests that there is no significant difference between the groups on this measure.

Chi Square

Group / Prior	CAI-EXP	PRT-EXP	
No	12	11	23
Yes	1	2	3
Total	13	13	

Raw chi square = .37681 with 1 Degree of Freedom, Significance = .5393

Figure 15. Chi square for group by prior computer experience.

Chi Square

Group / Class	CAI-EXP	PRT-EXP	CONTROL	
FR.	0	1	3	3
SO.	2	1	3	6
JR.	4	4	1	9
SR.	4	4	6	14
PB	1	3	1	5
MS	2	3	3	8
DR	1	0	0	1
Total	14	16	17	

Raw chi square = 10.59305 with 12 Degrees of Freedom, Significance = .5641

Figure 16. Chi square for the class standing by group.

Since quiz 8 also deals with Input/Output a Pearson's correlation was applied to the I/O scores and the total scores for that quiz and quiz 4. A positive correlation (r = .3639, $P<.05$) resulted from the analysis of the I/O scores and a positive correlation (r = .3852, $P<.05$) resulted from the analysis of the total scores. The results reported here will be discussed in the next section.

V. Discussion

Based on the results reported above it is clear that the null hypothesis, namely, that the CAI lesson has no effect on the performance of students exposed to it can be rejected. The performance of the PRT-EXP group was found not to be significantly different from that of the CONTROL group. The aspects of the CAI based materials, therefore, have had an impact on the performance of the CAI-EXP group. This is clear from the fact that both the CAI-EXP and the PRT-EXP groups were exposed to the same textual material. It cannot be concluded that the content of the presentation is responsible for the effect discovered. One can only conclude that the CAI-EXP group's performance was influenced by the opportunity to do practice problems and to have their results analyzed by the system which gave them immediate feedback.

The attitude differences reported by Crawford (1970) and Edwards (1978) for those students using CAI based materials can also be discounted in this study as all students in the study were exposed to computers as part of their course requirements. Hazen (1979) reported no attitude differences in a study of a FORTRAN programming course for business students. This supports the position that the effect demonstrated was due to the aspects of the lesson and not to some effect from exposure to the computer itself.

The average G.P.A. for the members of the CAI-EXP and the PRT-EXP groups combined was 3.10, and as reported above there was no significant difference between these two groups on this measure. This suggests that CAI is a viable medium for students who are not low achievers. This finding runs contrary to that reported by Jameson (1974), Splittgerber (1970), and Chambers and Sprecher (1980). In an attempt to explain this difference it is noted that the studies reported by these authors dealt with comparisons of CAI and traditional lecture classes while our study was conducted in a self study environment. We have also supplemented the existing materials with a CAI lesson as opposed to replacing the existing materials with CAI lessons.

Splittgerber (1979) also reported that the usefulness of simulation and problem solving is questionable in CAI lessons. Again the findings reported here are contrary to his. The only significant difference between the textual materials and the CAI lessons developed for this study was the inclusion of problem solving and dynamic simulation in the CAI lesson.

It can thus be concluded that the reported differences in the effectiveness of these techniques is due to the environments in which the lessons were used.

The findings reported here suggest that the use of CAI as a supplement to existing self study materials warrants further study. Similar lessons for other problem areas in FORTRAN programming could be developed and the effect of their use could be studied. CAI lessons should also be developed to supplement self study courses in other disciplines to determine if the effect reported here would also be found in a non-computer programming environment. Another area for further research is the language used to produce CAI materials. Splittgerber (1979) reports on 33 CAI programs but does not report the languages with which these materials were developed. It is difficult to draw conclusions on the usefulness of one language over another until a large number of lessons and studies have been analyzed. Certainly the use of Pascal provides great deal of computational power that may, or may not, be available in special purpose author languages.

Bibliography

Abelson, H., Goodman, N., and Rudolph, L. LOGO Manual: M.I.T. Artificial Intellegence Memo # 313. , 1974, M.I.T.

Bagley, C.A. and Klassen, D.L. Instructional Computing in Correctional Institutions. Educational Technology , 19,479, 19 pp 37-40.

Bailey, D.E. Ingredients for Excellence in Computer-Based Education Systems. in National Educational Computing Conference , 1979, University of Iowa pp 2-6.

Benuesnesti, M. and Friedman Jr., H.G. Data Structures: An Introduction for PL/1 (PLATO Lesson). Computer-Based Education Research Laboratory, 1975, University of Illinois at Urbana.

Bowels, K.L. Microcomputer Problem Solving Using Pascal. New York: Springer-Verlag Inc., 1977.

Bregar, W.S. and Farley, A.M. Artificial Intellegence Approaches to Computer-Based Instruction. Journal of Computer-Based Instruction , 1980, 6,4 , pp 106-114.

Brown, J.S. and Burton, R.R. SOPHIE-A Pragmatic Use of Artificial Intellegence in CAI. in Proceedings of the Annual Conference of the ACM ,1974, pp 571-579.

Chambers, J.A. and Sprecher, J.W. Computer Assisted Instruction: Current Trends and Critical Issues. Communications of the ACM , 1980, 23,6 , pp 332-342.

Collins, R.W. and Duff, S.J. Computer-Assisted Test Construction via Automatic Program Generation: Using PROBGEN II to Create Individualized Exams and Problem Sets. in National Educational Computing Conference , 1979, University of Iowa.

CONDUIT, P.O. Box 388, Iowa City, Iowa 52240.

Crawford, A.N. A Pilot Study Of Computer-Assisted Drill and Practice in Seventh-grade Remedial Mathematics. California Journal of Educational Research. , 1970, 21 , pp 170-181.

Edwards, L. The Effects of CAI on Achievement and Attitude in The Freshman Survey Mathematics Curriculum. in Ninth Conference on Computers in the Undergraduate Curricula , 1978, University of Denver pp 16-23.

Garcia, A. and Rude, S.A. Design of a CAI Tool to Generatively Teach Ecology. in National Educational Computing Conference , 1979, University of Iowa.

Hazen, M., Daly, C., Embley, D., Nagy, G., & Prange, W. Initial Evaluation Results for an Introductory Programming Course Without Lectures. in National Educational Computing Conference , 1979, pp 150-160.

Hoye, R.E. and Wang, A.C. Index to Computer Based Learning. Englewood Cliffs, N.J.: Educational Technology Publications, 1973.

Jay, T.B. Computerphobia: What to Do About It. Educational Technology , 1981, 21,1 pp 47-48.

Kehrberg, K.T. Microcomputer Software Development: New Strategies for a New Technology. in National Educational Computing Conference , 1979, University of Iowa pp 343-347.

Krittzberg, C.B. and Schneiderman, B. FORTRAN Programming: A Spiral Approach. New York: Harcourt Brace Jovanovich, Inc., 1975.

Moursund, D. Overview of CAI. in Computers in Education Resource Handbook , 1975, University of Oregon, pp V.A.1-V.A.5.

Nie, N.H. SPSS: Statistical Package for the Social Sciences. New York: McGraw-Hill Inc., 1975.

Nievergelt, J. A Pragmatic Introduction to Courseware Design. Computer , 1980, Sept., pp 7-20.

Paden, D.W. and Barr, M.D. Computer-Assisted Instruction for Large Classes in Economics. in National Educational Computing Conference , 1979, University of Iowa, pp 227-233.

Padua, D., Groupe, A., Simkin, R., & Peltz, S. An Introductory Lesson to APL (PLATO Lesson). Computer-Based Education Research Laboratory, 1975, University of Illinois at Urbana.

Pollack, R.A. Computer Augmented Video Education: An Overview. in Ninth Conference on Computers in Undergraduate Curricula , 1978, University of Denver, pp 1-5.

Pratt, L.J. and Davis, L.D. The Use of Computer Aided Instruction in the Teaching of Macroeconomic Principles. SIGCUE Bulletin (ACM) , 1981, 15,1 pp 2-15.

Shapiro, S.C. PLATO Lessons for a Data Structures Course. Technical Report #15 , 1974, Indiana University, Computer Science Department.

Slaughter, K. and Montanelli, R. An Introduction to FORTRAN Subroutines and Subprograms (PLATO Lesson). Computer-Based Education Research Laboratory, 1975, University of Illinois at Urbana.

Splittgerber, F.L. Computer-Based Instruction: A Revolution in the Making? Educational Technology , 1979, 19,1 pp 20-25.

Sorlie, W.E. and Essex, D.L. So You Want to Develop a Computer Based Instruction Project? Some Recommendations to Consider First. Educational Technology , 1979, 19,3 pp53-57.

Steinberrg, R.E. Learner Interaction in Computer-Assisted Instruction. in Computer Based Education: Mission of the Future ,1979, pp 194-202.

Wood, G. Fundamentals of Psychological Research. Boston: Little, Brown, and Company, 1977.

APPENDIX

Appendix A

The following pages contain two examples of the display presented to a student during a simulated execution of FORTRAN I/O statements using FORMAT statements. First, the simulated execution of a READ statement is shown (pages 53 thru 60). Following this the simulated execution of a statement is shown (pages 61 thru 65). In both cases the data values and strings were randomly generated by the system.

```
234739  2 11294612869814 23
-------------------------------------
 105 FORMAT(I2,X,F3.1,2X,I2,I2,2X,F5.1)
-------------------------------------
      READ 105,J,A,N1,N2,R1
-------------------------------------
```

FIRST THE INTEGER J WILL BE READ FROM THE FIRST TWO COLUMNS OF THE INPUT DATA CARD USING THE I2 SPECIFICATION. THIS VALUE IS STORED IN THE LOCATION LABELED J

	MEMORY
J	23
A	
N1	
N2	
R1	

```
234739  2 11294612869814 23
      105 FORMAT(I2,X,F3.1,2X,I2,I2,2X,F5.1)
          READ 105,J,A,N1,N2,R1
```

NEXT THE X FORMAT SPECIFICATION WILL CAUSE COLUMN 3, ON THE DATA CARD, TO BE SKIPPED

	MEMORY
J	23
A	
N1	
N2	
R1	

```
234739   2 1129461286981423
 105 FORMAT(I2,X,F3.1,2X,I2,I2,2X,F5.1)
      READ 105,J,A,N1,N2,R1
```

NEXT THE REAL VARIABLE
A WILL BE READ FORM
COLUMNS 4-6 USING AN
F3.1 SPECIFICATION.

	MEMORY
J	23
A	73.9
N1	
N2	
R1	

```
234739  2 11294612869814 23
105 FORMAT(I2,X,F3.1,2X,I2,I2,2X,F5.1)
    READ 105,J,A,N1,N2,R1
```

THEN THE NEXT 2 COLUMNS ARE SKIPPED BY THE 2X SPECIFICATION.

	MEMORY
J	23
A	73.9
N1	
N2	
R1	

```
234739  2 1129461286981423

 105 FORMAT(I2,X,F3.1,2X,I2,I2,2X,F5.1)

     READ 105,J,A,N1,N2,R1
```

NEXT THE INTEGER N1 IS READ FROM COLUMNS 9 & 10 NOTICE THAT THE TRAILING BLANK IS CONVERTED TO A ZERO WHEN IT IS READ WITH AN I2 SPECIFICATION

	MEMORY
J	23
A	73.9
N1	20
N2	
R1	

```
234739  2 11294612869814 23
      105 FORMAT(I2,X,F3.1,2X,I2,I2,2X,F5.1)
            READ 105,J,A,N1,N2,R1
```

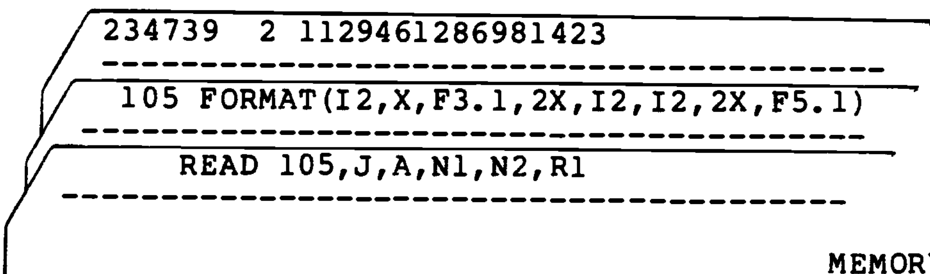
THEN N2 IS READ FROM COLUMNS 11 & 12 USING AN I2 SPECIFICATION

	MEMORY
J	23
A	73.9
N1	20
N2	11
R1	

```
234739  2 1129461286981423
 105 FORMAT(I2,X,F3.1,2X,I2,I2,2X,F5.1)
      READ 105,J,A,N1,N2,R1
```

COLUMNS 13 & 14 ARE
SKIPPED BY THE 2X CODE
IN THE FORMAT STATEMENT

	MEMORY
J	23
A	73.9
N1	20
N2	11
R1	



FINALLY R1 IS READ FROM COLUMNS 15 THRU 19 BY THE F5.1 SPECIFICATION

	MEMORY
J	23
A	73.9
N1	20
N2	11
R1	4612.8

```
  111 FORMAT(X,'N1 = ',I4,' R3= ',F4.2)
```

```
      PRINT 111,N1,R3
```

MEMORY

N1	23
IT	2703
R3	1.03

FIRST WE SEE THAT ONE SPACE IS TO BE SKIPPED ON THE PRINT LINE - THIS IS THE EFFECT OF THE X SPECIFICATION.

```
  111 FORMAT(X,'N1 = ',I4,' R3= ',F4.2)
 ----------------------------------------
      PRINT 111,N1,R3
 ----------------------------------------
```

NEXT THE STRING 'N1 = ' IS PRINTED ON THE OUTPUT LINE IN SPACES 2-6 NOTE THAT THE QUOTE MARKS ARE REMOVED BEFORE IT IS PRINTED.

MEMORY

N1	23
IT	2703
R3	1.03

```
 N1 =
----------------------------------------
```

```
  111 FORMAT(X,'N1 = ',I4,' R3= ',F4.2)
---------------------------------------
```

```
      PRINT 111,N1,R3
---------------------------------------
```

NEXT THE CURRENT VALUE OF THE VARIABLE N1 IS PRINTED USING AN I4 CODE -NOTE THAT THE VALUE OF N1 ONLY OCCUPIES 2 DIGITS, WHILE THE I4 SPECIFICATION WILL PRINT THE VALUE IN THE FIELD RIGHT JUSTIFIED AND WITH BLANKS PRINTED TO THE LEFT.

	MEMORY
N1	23
IT	2703
R3	1.03

```
 N1 =   23
---------------------------------------
```

```
111 FORMAT(X,'N1 = ',I4,' R3= ',F4.2)
```

```
    PRINT 111,N1,R3
```

THE STRING ' R3= ' IS NOW PRINTED ON THE PRINT LINE IN SPACES 11 - 15. AGAIN NOTE THAT THE QUOTE MARKS ARE NOT PRINTED.

MEMORY

N1	23
IT	2703
R3	1.03

```
N1 =   23 R3=
```

```
111 FORMAT(X,'N1 = ',I4,' R3= ',F4.2)
```

```
    PRINT 111,N1,R3
```

FINALLY THE VALUE OF R3 IS PRINTED USING AN F4.2 SPECIFICATION. NOTE THAT THE DECIMAL POINT IS PRINTED AND SO A SPACE MUST BE PROVIDED FOR IT IN THE SPECIFICATION.

	MEMORY
N1	23
IT	2703
R3	1.03

```
N1 =   23 R3= 1.03
```