DOCUMENT RESUME

ED 084 834 EM 011 655

AUTHOR Bentz, William F.; Ambler, Eric E.
TITLE Learning to Analyze and Code Accounting Transactions in Interactive Mode.
INSTITUTION Kansas Univ., Lawrence. School of Business.
PUB DATE 73
NOTE 13p.

EDRS PRICE MF-$0.65 HC-$3.29
DESCRIPTORS *Accounting; Attention Control; Business Administration; Business Education; *Computer Assisted Instruction; *Computer Programs; Feedback; Higher Education; Input Output; *Interaction; *Man Machine Systems; Program Descriptions; Student Attitudes
IDENTIFIERS Cathode Ray Tube; CODE; Coding; Results; TEACH; TRANS; Transactional Analysis in Accounting; Update

ABSTRACT

An interactive computer-assisted instructional (CAI) system, called CODE, is used to teach transactional analysis, or coding, in elementary accounting. The first major component of CODE is TEACH, a program which controls student input and output. Following the statement of a financial position on a cathode ray tube, TEACH describes an event to which the student responds by typing his coded version of the accounting entry necessitated by the event; the program evaluates the responses, provides feedback, updates the financial position and begins a new cycle. Other components of CODE are: 1) TRANS, a data file which stores instructions, statements, events, and entries; 2) UPDATE, a program which creates new data files and modifies existing ones; and 3) RESULTS, a data file which monitors student performance. The hypothesis that coding can be well taught by an interactive CAI system is supported by the findings in learning theory that systems which provide feedback, offer attention direction and develop positive attitudes will be most effective. Preliminary testing shows that CODE meets these requirements; formal experimental testing of CODE has been scheduled for the fall semester of 1973 at Kansas University. (PB)

Learning to Analyze and Code Accounting Transactions in Interactive Mode

William F. Bentz
and
Eric E. Ambler*

School of Business
University of Kansas
Lawrence, Kansas 66044
(913) 864-4666

*Assistant Professor and Graduate Student.
Address all correspondence and inquiries to Professor Bentz.

INTRODUCTION

The purpose of this paper is to describe a computer-extended-instruction system, called CODE, that facilitates the learning of transaction analysis in elementary accounting. In order to develop some perspective, the first section of this paper is devoted to a discussion of the significance of coding (transaction analysis) in accounting. Next, a complete description of the CEI system called CODE is presented. This description of CODE includes both a description of its operation from a student's perspective, and a more technical description of the methodology used in its construction. Some of the fundamental principles of learning that stimulated the development of CODE are outlined in section three. The plans for experimentally testing CODE are described in the final section of the paper.

THE SIGNIFICANCE OF CODING IN ACCOUNTING

Accounting students quickly learn that accounting practice is primarily concerned with observable financial transactions.[1] They also learn that some events result in an accounting entry, while other events are not reflected anywhere in the accounting system -- even though they may be of tremendous importance to the firm. Further, students learn that events and transactions cannot be recorded themselves; symbols are recorded and stored. These symbols characterize the extent to which each transaction possesses certain attributes or qualities. In accounting, those qualities or attributes most often recorded and stored are item counts, prices, names, addresses, dollar amounts to be paid or received, etc.

Frequently, students are introduced to the coding process by being asked to determine the effects of transactions on a limited set of account

balances. This task involves determining which accounts are affected, the extent of the effect (dollar amount), and the direction of the effect (increase or decrease). Many texts introduce this coding process, sometimes called transaction analysis, before introducing bookkeeping conventions such as debits and credits.[1] In many other texts, debit-credit conventions are introduced immediately so that all coding is done within the debit-credit framework.[2]

During the process of coding accounting entries, accounting students learn several accounting concepts and conventions. Since account balances are the cumulative result of a series of transactions, students learn which accounts are affected by each transaction. Second, they learn what the effects of a transaction are on the statements of earnings and financial position. As a result, many texts emphasize these statement effects from the very start by preparing updated financial statements after the analysis of each of several transactions. Third, they learn that each transaction must be coded so that the continuous balance of the accounting equation is maintained. And fourth, they learn that the chart of accounts is a classification scheme that is not free of ambiguities. While this list is not exhaustive, it does include several concepts and propositions that are learned in the process of coding financial transactions.

DESCRIPTION OF THE CODE SYSTEM

The purpose of CODE, an interactive computer-extended-instruction system, is to facilitate the development of the transaction coding skills of novice accountants. A general description of the way CODE works is developed first from the perspective of students, and then in more technical

detail. The more technical discussion contains a description of the function of each software component of CODE, and a description of how the system is administered from the instructor's perspective.

Each student receives a handout which describes the sign-on procedures for a cathode ray tube-type terminal (CRT), and the commands necessary to access and execute a program called TEACH. Once accessed, TEACH controls the flow of information that is presented to the student as well as receiving student responses. Thus, TEACH assumes input-output control throughout the student's session with CODE.

Immediately after TEACH is accessed by the student, a set of instructions appears on the CRT. These instructions indicate how the student is to respond when events are described on the CRT, and the method that should be used to code the accounting entries that are necessitated by these events. The instructions remain on the CRT until the student indicates that he is ready to proceed.

An abbreviated statement of financial position is included in the set of instructions. When the student indicates that he is through with the instructions and ready to proceed, the statement of financial position appears in the upper half of the CRT, where it remains for the rest of the session. The statement of financial position in Figure 1 appears as it does on the CRT. Headings are omitted and the statement is in unclassified form due to the severe space restrictions imposed by the use of a CRT terminal.

The next four stages are repeated for each learning unit that is presented to the student. First, an event is described in three lines or less. Second, the student types his coded version of the accounting entry

FIGURE 1

ABBREVIATED STATEMENT OF FINANCIAL POSITION
AS IT APPEARS ON THE CRT

1 - CASH	66000	9 - ACCRUED PAYROLL	29000
2 - ACCOUNTS RECEIVABLE (NET)	435000	10 - ACCOUNTS PAYABLE	375000
3 - MARKETABLE SECURITIES	120000	11 - OTHER LIABILITIES	50000
4 - INVENTORIES	502000	12 - BONDS PAYABLE	295000
5 - PREPAID EXPENSES	12000	13 - COMMON STOCK	350000
6 - LAND	110000	14 - PAID-IN CAPITAL	109000
7 - BUILDINGS (NET)	57000	15 - RETAINED EARNINGS	114000
8 - OTHER ASSETS	20000	16 - INCOME ACCOUNTS	0
TOTAL ASSETS	1322000	TOTAL EQUITIES	1322000

that is necessitated by the event just described on the CRT. His answer is compared with the correct answer and a feedback response appears on the CRT. Correct answers are just acknowledged, but incorrect answers are corrected. At this stage in the interaction, the CRT contains: (1) a statement of financial position, (2) the description of an event, and (3) the correct coding of an accounting entry to record the financial consequences of the event.

The fourth stage is possible only because the system operates in time-sharing mode with a CRT terminal. The cursor moves to those account balances that are affected by the accounting entry just coded and these balances are updated, along with any column totals that are affected. The statement of financial position is updated in this manner after each accounting entry has been coded. This four-stage cycle constitutes one learning unit which begins again with the clearing of the bottom half of the CRT while the updated

statement of financial position is retained in the upper half, and a new event is described to the student. This event-coding-response-update cycle is repeated until there are no more learning units on the data file.

The program that controls the student's session with the CODE system is called TEACH. It reads the use of instructions from a data file (called TRANS) and displays them on the CRT at the start of each student's session, and whenever the student asks to see them again.[3] Likewise, TEACH reads the statement of financial position from the data file TRANS and updates the statement after each transaction is coded and entered. Similarly, the events and the corresponding accounting entries are read from TRANS and displayed on the CRT. These are the two inputs required by each learning unit; the description of an event and a properly coded accounting entry are required to record the financial consequences of that event.

The function of the main data file (called TRANS here) is to store instructions, a beginning statement of financial position, and a set of events with their associated accounting entries in coded form (learning units). The set of instructions is easily changed, as is the statement of financial position. The size of the main data file is a function of the number of learning units included. This discussion of CODE assumes one main data file, but the instructor can use as many data files as he desires so long as either students know which file to access, or TEACH is modified to automatically access the proper file.

The program called UPDATE performs a specialized but important function. UPDATE is used to modify existing data files, such as TRANS, and to create new data files. These file activities are important because all of

the learning materials used in the CODE system are stored on the data files. Therefore, the learning material content of CODE is revised and improved by changing the content of TRANS or any similar data file used by TEACH.

Each instructor can construct one or more data files that correspond to the major sections of the text being used in his accounting class. No reprogramming is necessary to accomplish these changes, since only the data files are being changed. Moreover, UPDATE facilitates this revision process by facilitating the construction or modification of these data files. For example, UPDATE accepts unformatted input and creates a file in the format required by TEACH. This feature alone is very helpful.

The last major component of CODE to be described is a data file called RESULTS. During execution, TEACH accumulates the performance of each student on all the learning units. At the end of each student's session, his name and performance are written on one line of the file called RESULTS.

The information contained on the performance file can be used in several ways. First, a simple listing of the file will provide some information on the performance of all those students using CODE. Those students having difficulty with the material will be apparent, as will those students who are unchallenged by the material. Also, the listing indicates who has used CODE!

More importantly, the file of the results for an entire class can be used to perform an item analysis on each learning unit included in TRANS. Some units will prove to be too simple or too difficult, while others may be confusing. In any case, the data necessary to perform several types of

analysis is provided on the data file called RESULTS, after being accumulated by TEACH.

At the present time, no information is collected about the types of errors being made by students. Entries are either coded correctly or they are coded incorrectly. There is no provision made for detecting different degrees of correctness. As a result, the data contained in the performance file called RESULTS is just a series of ones (1) and zeros (0). A one in position five of line ten indicates a correct response to item five by the student whose name appears on line ten. A zero indicates the student did not code the transaction correctly.

LEARNING PRINCIPLES EMBODIED IN CODE

Previous findings in learning theory form the theoretical basis for hypothesizing that coding can be better taught by an interactive CEI system than by other, non-interactive systems.[4] This rather bold hypothesis is based on many factors, but only the three most important ones are reviewed here.

Feedback

The importance of relatively frequent, non-aversive feedback is well accepted in education. Moreover: "Present evidence also suggests that the efficacy of feedback varies in proportion to its completeness" [1, p. 302]. Providing a correct answer is better than only indicating whether an answer is correct or incorrect. Thus, the feedback provided by CODE is fast, accurate, relatively complete and non-aversive. The one way in which the completeness of the feedback in CODE could be improved is to indicate to each student why an answer is incorrect. This feature is not part of CODE yet.

Attention Direction

An important function of any learning process is the attention directing function. The very nature of CODE directs a student's attention to specific aspects of the coding process and to the specific, sequenced set of events to be coded. But, even more importantly, the use of a CRT terminal lets CODE show every student the effects of every transaction on the financial statements. Both the arrangement of this information on the CRT and the movement of the cursor are attention directing devices that should facilitate learning. In summary, CODE has many attention directing features that cause the student to focus on the important aspects of the coding process.

Attitudes

Since education is directed toward influencing future behavior, the development of positive attitudes toward accounting is extremely important.[5] One way to develop positive attitudes is to clearly define what is required of the student (Mager [6]), and to make sure the student has the requisite knowledge to perform the desired tasks (Gagné [5]). CODE is expected to be helpful in the development of positive attitudes both because students enjoy working with interactive programs and because of the sequential organization of the material to be learned. Also, the learning task is clearly defined, as is satisfactory performance.

Many other learning theory concepts are relevant to the use of CODE, but the three topics mentioned in this section seem to be especially relevant to CODE.

PRELIMINARY AND PROPOSED TESTING

Student reception to CODE and other interactive programs in use at Kansas University has been favorable in terms of developing interest and favorable attitudes. However, the CODE system will not be tested in a controlled, experimental setting before the fall of 1973. Since CODE's success depends on the quality of the transaction material contained on the data file, the plan is to continue to develop and revise the transaction file (TRANS) through the summer of 1973. During the fall semester of 1973, the CODE system will be compared with traditional teaching methods under experimental conditions. Only then will the effectiveness of the CODE system be known with any confidence.

FOOTNOTES

[1]In fact, accounting has been defined (AICPA, 1941) as the art of recording, classifying, summarizing, and interpreting financial transactions and events in money terms.

[2]In some texts, debit-credit terminology is deferred several chapters, as in Johnson [6]. In other texts, the debit-credit terminology comes immediately after the presentation of transaction analysis, as in Burns [3], Fertig [4], and Meigs [8]. Others introduce debit-credit terminology at the start of their texts, as in Schrader [9]. Bruns [2] presents an input-output matrix for recording entries without debit-credit terminology, but then he introduces debit-credit terminology in the next chapter.

[3]A program, of course, does not actually perform I/O or processing functions; it causes them to happen. However, it is convenient to describe the system using active verb forms.

[4]Included among the non-interactive systems are many computerized practice sets or problem sets. Examples of non-interactive, computerized transaction analysis exercises can be found in Wilkenson [10] and in Williams [11].

[5]See Mager [7] and Ansubel [1].

REFERENCES

[1] Ansubel, David P., and Floyd G. Robinson. School Learning: An Introduction to Educational Psychology. New York: Holt, Rinehart and Winston, Inc., 1969.

[2] Bruns, William J., Jr. Introduction to Accounting. Reading, Mass.: Addison-Wesley Publishing Company, 1971.

[3] Burns, Thomas J. and Harvey Hendrickson. The Accounting Primer: An Introduction to Financial Accounting. New York: McGraw-Hill Book Company, 1972.

[4] Fertig, Paul E., Donald F. Istvan, and Homer J. Mottice. Using Accounting Information (second edition). New York: Harcourt Brace Jovanovich, Inc., 1971.

[5] Gagné, Robert M. The Conditions of Learning. New York: Holt, Rinehart and Winston, Inc., 1965.

[6] Mager, Robert F. Preparing Instructional Objectives. Palo Alto, California: Fearon Publishers, Inc., 1962.

[7] Mager, Robert F. Developing Attitude Toward Learning. Belmont, California: Fearon Publishers, Inc., 1968.

[8] Meigs, Walter B., A. N. Mosich, and Charles E. Johnson. Accounting: The Basis for Business Decisions. New York: McGraw-Hill Book Company, 1972.

[9] Schrader, William J., Robert E. Malcom, and John J. Willingham. Financial Accounting: An Input/Output Approach. Homewood, Ill.: Richard D. Irwin, Inc., 1970.

[10] Wilkenson, Joseph W. Accounting with the Computer: A Practice Case. Homewood, Ill.: Richard D. Irwin, Inc., 1969.

[11] Williams, Thomas H., James Wesley Deskins, and J. Stanley Fuhrmann. Information Processing Simulation Model. Homewood, Ill.: Richard D. Irwin, Inc., 1968.