Management Services: A Magazine of Planning, Systems, and Controls

Volume 5 | Number 4 Article 5

7-1968

Computer Programming Approach to the Design of Accounting Systems

R. L. Mathews

This article, describing as it does an application worked out on one of the largest and most expensive computers, may be beyond the scope of most in-house EDP facilities. But for those firms dealing with data centers or university computing centers, it offers real possibilities.

A COMPUTER PROGRAMMING APPROACH TO THE DESIGN OF ACCOUNTING SYSTEMS

by R. L. Mathews
Australian National University

DESPITE the key role of accounting in the provision of information for managerial and other purposes, the revolution in data processing that has resulted from the development of the electronic computer has not had the impact on the accounting information system that might reasonably have been expected. To some extent this may have been due to a reluctance on the part of accountants to substitute new and unfamiliar methods of data processing for techniques that have become hallowed by usage and tradition; but such an explanation is difficult to reconcile with the observable fact that accountants, while apparently loath to change the structure of the accounting system, have been exceedingly active in applying the computer to specialized tasks, such as payroll preparation, invoicing, and the recording of stock movements, which form part of the wider accounting process.

The principal reason for the lag in the application of computer technology to the accounting information system would seem to be incompatibility between the form of the double-entry recording system, which it must be remembered has slowly evolved during the last five hundred years or more, and the data processing qualities of the computer. The nature of this incompatibility will be examined in the following paragraphs.

This article originally appeared in the Australian magazine *Abacus* (Sydney University Press, Sydney, Australia, Volume 3, Number 2, December, 1967). It is reproduced by permission of the author and publisher.

The author records his appreciation to members of the Accounting Division of the Graduate School of Business Administration, University of California, Los Angeles, for their comments on the program which is reproduced in the Appendix to this paper and to Mr. B. W. Smith, of the Australian National University Computer Centre, and members of the University's Department of Accounting and Public Finance for their comments on an earlier draft of the paper.

Accounting information system

The task of the accounting system is to provide information about economic transactions as a basis for planning, decision making and control. The information required includes reports of various kinds, which will be used by the persons receiving them to assess performance, initiate action, or facilitate control. The reports may be retrospective or anticipatory; they may deal with long-run or short-run situations; they may be comprehensive or sectional; but in any case they will be derived (if at times rather tenuously) from the accounting system which has been designed for the purpose of providing the information in question. An examination of the accounting information system that has developed in response to these needs indicates that the different aspects of data processing—collection and recording of input, computation, storage of data, control, and the dissemination of output—have come to be handled in a way that permits the fragmentation of the processing operations (albeit within the integrated framework that is provided by the double-entry system).

So far as the input process is concerned, the traditional system is characterized by a multiplicity of data sources, a multi-tiered classification system, a method of recording diverse transactions that has regard to the chronological order in which they occur, and a disaggregated system of ledger accounts which nevertheless reflects the integrating and equilibrating notion of duality of entry. The recording process that has evolved in response to these conditions inevitably results in transactions being dealt with individually rather than in the mass. This in turn means that one account at a time is selected (two accounts in respect to any one transaction) as the repository of transaction data. Some transactions occur frequently enough to make it worthwhile grouping them for purposes of the recording process by means of specialized journals and ledgers. Group treatment of such transactions followed recognition of this.

The mechanization of the recording process encouraged this tendency for data to be processed in runs of similar transactions, just as it facilitated the concurrent preparation of evidence documents (e.g., receipts or invoices), chronological records (i.e., journals), and ledgers. But the structure of the accounting system, as reflected in the ledger, has remained basically unchanged, and the input process continues to be characterized by the piecemeal approach that has been described.

To the extent that computation is carried out within the accounting system, it usually involves only simple arithmetical calculations and the determination of ledger balances. The accounting system has provided for the storage of data in a form which facilitates ready accessibility and easy reference, having regard to the purposes for which the information is likely to be required. Although one of the main functions of the system is that of control, and although some controls are built into the system (e.g., through the use of the control account technique or the derivation of standard cost variances), the controls which are obtained are not automatic and depend essentially on human review.

The reports or financial statements which constitute the output of the accounting system are heterogeneous in character, and are presented on a multiplicity of forms at different time intervals depending on their purpose. An elaborate summarization process is a necessary prerequisite to the preparation of annual financial statements, but the accounting system is not always designed in such a way as to provide, at least without painstaking analysis outside the formal structure of accounts, information of significance to management (e.g., with respect to cost behaviour in relation to volume) or proprietors (e.g., with respect to current values).

The computer

If the foregoing characteristics of the traditional accounting system are considered in relation to the qualities of the computer with respect to each of its main functions, a number of interesting problems can be seen to exist. So far as input is concerned, effective utilization of the computer demands the feeding in of data as far as possible in homogeneous runs and preferably as a by-product of other operations. The heterogeneous nature of business transactions and the selective method of recording them in the accounting system will produce an input bottleneck of such proportions as to invalidate the use of the computer unless special action is taken to deal with these problems. Such action includes, insofar as specialized accounting applications are concerned, the following:

(a) changes in system design, involving for example the organization of data in runs of transactions of similar accounting significance, e.g., stock purchases, credit sales, cash receipts;

(b) steps to obtain maximum advantage from computer facilities, involving, for example, the use of input media which give direct or "on-line" access to the computer at the time the transaction is first recorded and/or which produce punched cards or tapes either simultaneously with the preparation of conventional employee time cards, invoices, receipts, production reports, etc., or automatically from mark-sensed data.

Despite substantial progress in both these directions, however, complete computerization of input has been retarded by the failure of accountants to modify the design of the basic accounting system as reflected in the double-entry framework of ledger accounts.

Turning now to computation, it is a feature of modern computers that they can handle both complex mathematical problems and repetitive processing combined with simple arithmetical calculations. The first characteristic is not of much significance in relation to accounting, although it does assume importance for purposes of other business applications in the decision-making area. The second characteristic is of fundamental importance for purposes of accounting.

Data storage is not inherently a problem in computer applications because of the possibility of utilizing both the computer's own core storage and external media such as punched cards and magnetic tapes. In practice it becomes a problem in relation to accounting operations because of the need for continuous updating of, and ready access to, stored data. A partial solution to this problem has been found in the use of random access methods of storage, involving, for example, the use of magnetic disks, rather like gramophone records in a juke box, to store accounting data. Individual accounts or items can be selected and updated, thus avoiding the need to search sequentially through the complete tape or deck of cards. But while magnetic disks make it possible to obtain random access to accounts of similar significance, e.g., accounts receivable, cost records, etc., they do not easily provide random access of the two-dimensional kind which would be necessary for purposes of updating a complete set of double-entry accounts.

There is another feature of the computer, or more specifically of computer programing languages, which seems to have been seldom exploited for accounting purposes, but which we shall see has considerable potential with respect to the storage and processing of accounting data. This is the ability of the computer to store and process data in the form of two-dimensional or three-dimensional arrays.

In a computer system, control of data processing operations is achieved by means of a central processing unit which automatically arranges for the execution of the various functions that need to be performed, from input through to final output. This control is achieved by means of instructions which are given to the computer in the form of a stored program. In addition to ensuring automatic control of data as they are manipulated or transferred from one operation or function to another, the stored program can be made to incorporate accounting control in respect of managerial performance, such as is involved in the use of control accounts of standard cost variances. By programing the computer to print error messages or report on variances, the information necessary to achieve such control is provided automatically.

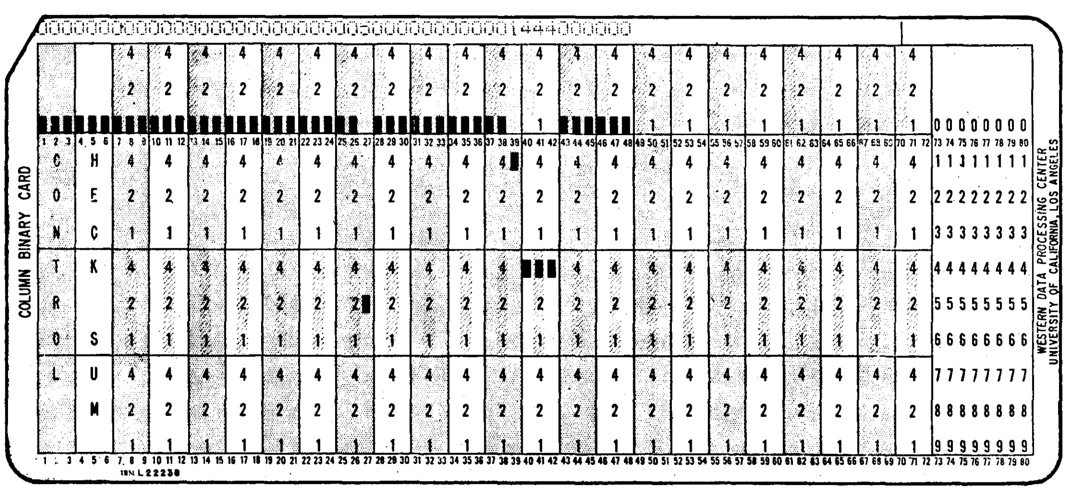
Finally, given the input and the instructions which are contained in the program, the computer will print its output automatically. It may thus be programed in such a way as to carry out without human intervention the elaborate summarization process which is needed to convert transaction data into financial statements such as income statements or balance sheets. There is a problem here, however, which depends for its solution on finding a method of recording all accounting transactions in a system that is compatible both with duality in the accounting record and with the integration of computer processing operations. As we shall see, this in turn requires modifications in the design of the accounting system.

Redesigning the system

Having regard to these qualities of the computer, it is not surprising that accounting applications have usually been restricted to areas where the input, storage, and accessibility problems that have been discussed may be minimized and where the repetitive processing, computational, control, and output features may be used to the fullest extent possible. Such applications include payroll, stocks, accounts receivable, accounts payable, cash receipts, cash payments, but do not typically extend to the complete accounting system. Because of the input and storage features which have been noted, it is difficult to program the computer to record ledger entries in the traditional way. So long as the accounting system is designed to contain a multiplicity of ledger accounts recording a wide range of transactions, programing is likely to be restricted to partial applications and the ac-

FIGURE I

A COMPUTER PROGRAMMING APPROACH TO ACCOUNTING



counting system will not be fully converted to computer operations. Although data may be organized in batches of similar transactions for computer processing, the large variety of transactions inhibits a global or total approach to the computerization of the accounting system, at least so long as the traditional form of the accounting system is retained.

Accountants have recognized the importance of systems analysis and design in relation to computer applications, but so far their work in this area has been mainly restricted to the partial applications which have been listed. They have been slower to recognize the need for a critical examination of the accounting system considered as a whole, involving among other things a review of the traditional basis of recording transactions in ledger accounts. But while it is difficult for the computer to record transactions in this way, there is a particular feature of the computer which suggests the possibility of an alternative solution to the problem, one which continues to record transactions on the basis of a duality concept but which relies on the recording of each transaction by means of a single entry in a matrix system instead of a double entry in a system of disaggregated accounts. This is the computer's ability, which we have noted, to store and process data in terms of two- (or three-) dimensional arrays or tables, that is to say in matrix form.

Matrix accounting

The idea of recording accounting data in a matrix system is not a new one. In business accounting the notion of a "spread sheet" incorporates a matrix approach to the recording of transactions,[1] and the spread sheet has recently been used to illustrate the possibility of applying mathematical optimizing techniques to the task of budgetary planning.[2] Other writers have used the term "double classification bookkeeping" to describe a matrix approach to accounting which they suggested could be adapted to punched card equipment or computers.[3] In national accounting the matrix approach has an even longer tradition, culminating in input-output accounting.[4]

A computer program

In writing a computer program, the programer should proceed logically through the stages of defining the information requirements and deciding on the form of the output before he reviews the input data and determines the processing and control requirements. For purposes of this paper it will be assumed that the information and output requirements are the same as in a conventional accounting system, except to the extent that a ledger matrix is substituted for a system of double entry accounts.

If the accounting system is designed in matrix form, the computer may be programed to set aside a number of fields corresponding to the cells in the matrix. In the Fortran IV programing language this may be done by means of what is known as a dimension statement. The statement

```
DIMENSION KASYS (11, 8)
```

thus instructs the Fortran processor to make provision for a table containing eleven rows and eight columns, or eighty-eight cells. We may think of KASYS as a code representing the accounting system in matrix form.

A second dimension statement

```
DIMENSION IASYS (11, 8)
```

together with an instruction to read in transaction data from the data cards

```
READ (5, 1) ((IASYS (I, J),
J = 1, 8), I = 1, 11)
```

enables the input to be posted to a holding matrix, the data in which are automatically transferred to an accumulating matrix by means of the instruction

```
KASYS (I, J) = KASYS (I, J)
+ IASYS (I, J)
```

(the accumulating matrix will have been cleared previously, i.e., each cell given the value of zero, by means of the instruction KASYS (I, J) = 0).

A READ statement instructs the computer to read data into the matrix from cards that have been punched in accordance with a prescribed FORMAT statement. If, as in Appendix D, the FORMAT statement takes the following form

```
1 FORMAT (8 I 6)
```

this means that, to conform with FORMAT statement no. 1, cards must be punched with eight fields each of six values representing digits or signs. The eight fields in each card may be identified with the eight columns in the matrix that has been established. The eighth column, representing total debits, should be punched with

---

[1] See Eric L. Kohler, *A Dictionary for Accountants*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1952.

[2] See A. Charnes, W. W. Cooper, and Y. Ijiri, "Breakeven Budgeting and Programming to Goals," *Journal of Accounting Research*, Spring, 1963, p. 16. See also Y. Ijiri, F. K. Levy, and R. C. Lyon, "A Linear Programming Model for Budgeting and Financial Planning," *Journal of Accounting Research*, Autumn, 1963, p. 198.

[3] J. G. Kemeny, A. Schleifer, J. L. Snell, and G. L. Thompson, *Finite Mathematics with Business Applications*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1962, pp. 346-351, 358-362.

[4] The earliest presentation of national accounts in tabular form was undertaken by F. Quesnay in his *Tableau Economique*, while W. Leontief was responsible for the development of a formal system of input-output accounting in his book *The Structure of the American Economy 1919-1939*, Oxford University Press, 1941. See also. Russell Mathews, "Business Enterprise Accounts in Relation to Different Kinds of Social Accounting Systems," *The Economic Record*, March, 1960, p. 95 Richard Mattessich, in "Towards an Axiomatization of Accountancy, with an Introduction to the Matrix Formulation of Accounting Systems," *Accounting Research*, October, 1957, p. 328, and *Accounting and Analytical Methods*, Richard D. Irwin, Inc., Homewood, Illinois, 1964, applies the matrix approach to the development of both business and national accounting systems.

FIGURE 2

```
0FORMAT(1H1,28X,13HLEDGER MATRIX/,7X,48H  CASH INVEN ACDEP ACPAY JS
1CAP JSCUR SALES TOTDR/,7H CASH  ,8I6/,7H ACREC ,8I6/,7H INVEN ,8I6
2/,7H FIXED ,8I6/,7H COFGS ,8I6/,7H RENT  ,8I6/,7H WAGES ,8I6/,7H D
3EPRC ,8I6/,7H OTHEX ,8I6/,7H JSDWG ,8I6/,7H TOTCR ,8I6)
```

```
WRITE(6,4)((KASYS(I,J),J=1,8),I=1,11)
```

zeros because the computer will be programed to calculate the totals automatically. If eleven cards are punched representing the eleven rows in the matrix, the READ statement

READ (5, 1) ((IASYS (I, J), J = 1, 8), I = 1, 11)

will ensure that the cards are read in the order of the rows and that the data contained therein are assigned to their appropriate cells. The eleventh card, representing total credits, should be punched with zeros in all eight fields because the program will arrange for the totals to be calculated automatically.

The first week's transactions for the hypothetical business of J. Smith are reproduced in Appendix A on page 40, and a data card is illustrated in Figure 1 on page 34.

It is necessary to comment briefly on the way in which transaction data are accumulated prior to the punching of the cards. In a fully computerized system it is possible to conceive of data being fed into the matrix direct from input media, which simultaneously produce the documents that provide evidence of the transactions. However, in the system that is described in this paper the different groups of transactions are recorded initially in a system of specialized journals (see Appendix B, page 40). Separate computer programs could be written to record these transactions and produce the transaction data needed as input for the main program. Specialized operations such as payrolls could be dealt with in the same way. On the basis of the information contained in these originating records, weekly transaction tables (or if desired daily tables) may be prepared for the general ledger matrix and the two subsidiary ledgers (see Appendix C, page 41). The data cards to be processed in the main system will then be punched, one card for each row, in accordance with the transaction tables.

After the posting is complete (and several sets of cards may be posted in this way), the computer may be programed to print out the data contained in the ledger matrix by means of another FORMAT statement, which specifies the form of the output, and a WRITE statement (see Figure 2, this page).

On the basis of the first week's transactions listed in Appendix A, these instructions would result in a print-out of the ledger in matrix form as reproduced in Figure 3 below on this page.

The ledger matrix has been designed in such a way as to assign rows to accounts receiving what would traditionally be recorded as debit entries and columns to accounts receiving credit entries. Codes have been used for account names as follows:

CASH	Bank
ACREC	Accounts Receivable
INVEN	Inventories
FIXED	Fixed Assets
COFGS	Cost of Goods Sold
DEPRC	Depreciation
OTHEX	Other Expenses
JSDWG	J. Smith Drawings A/c
TOTCR	Total Credits
ACDEP	Accumulated Depreciation
ACPAY	Accounts Payable
JSCAP	J. Smith Capital A/c
JSCUR	J. Smith Current A/c
TOTDR	Total Debits

It will be seen that the first transaction, the bank deposit of $5,000 by J. Smith representing his capital contribution to the business, has been recorded as an entry in the cell corresponding to the CASH

FIGURE 3

LEDGER MATRIX

	CASH	INVEN	ACDEP	ACPAY	JSCAP	JSCUR	SALES	TOTDR
CASH	0	0	0	0	5000	0	1444	6444
ACREC	0	0	0	0	0	0	621	621
INVEN	0	0	0	4064	0	0	0	4064
FIXED	1200	0	0	0	0	0	0	1200
COFGS	0	1246	0	0	0	0	0	1246
RENT	200	0	0	0	0	0	0	200
WAGES	190	0	0	0	0	0	0	190
DEPRC	0	0	0	0	0	0	0	0
OTHEX	0	0	0	0	0	0	0	0
JSDWG	60	0	0	0	0	0	0	60
TOTCR	1650	1246	0	4064	5000	0	2065	14025

FIGURE 4

AC RECEIVABLE
WEEK 1

	BEGBAL	DR 1	DR 2	DR 3	CR	ENDBAL
C.CARTER	0	45	15	0	0	60
J.JACKSCN	0	85	40	94	0	219
N.NORTON	0	95	43	0	0	138
P.PARKER	0	20	0	0	0	20
R.ROYCE	0	0	0	0	0	0
W.WILLIAMS	0	40	95	49	0	184
TOTAL	0	285	193	143	0	621

AC PAYABLE
WEEK 1

	BEGBAL	DR	CR	ENDBAL
B.BARKER	0	0	575	575
G.GRACE	0	0	568	568
H.HOWARD	0	0	561	561
M.MOSS	0	0	1020	1020
S.STEWART	0	0	740	740
T.THOMPSON	0	0	600	600
TOTAL	0	0	4064	4064

row and the JSCAP column; similarly with the other transactions. Some accounts, such as CASH and INVEN, which traditionally receive both debit and credit entries that are offset in other accounts in the system, are represented in both the rows and the columns of the matrix. It is not necessary to include all accounts in both columns and rows, since some accounts do not record transactions involving both debit and credit entries.

Negative entries may be made in cells where accounts normally receiving debit entries need to record credit entries; thus cash received from debtors may be recorded as a negative item in the cell formed by the ACREC row and the CASH column.

The program which is illustrated merely records accumulated entries in each cell. In practice, complete records of transactions affecting each account are likely to be required. These could be obtained either by preparing a new ledger matrix each day and aggregating the results by means of a separate program or by adding a third dimension to the table to provide information about the date of each transaction and to give opening and closing balances. The computer can easily be programed to hold three-dimensional tables. It is not necessary to include summary accounts, e.g., trading and profit and loss or income accounts, in the ledger matrix; separate instructions are included in the program to print out, at appropriate intervals, a trial balance, an income statement, and a balance sheet on the basis of account balances derived from the ledger matrix.

FIGURE 5

TRIAL BALANCE
WEEK 1

	DR	CR
CASH	4794	
ACREC	621	
INVEN	2818	
FIXED	1200	
COFGS	1246	
RENT	200	
WAGES	190	
DEPRC	0	
OTHEX	0	
JSDWG	60	
ACDEP		0
ACPAY		4064
JSCAP		5000
JSCUR		0
SALES		2065
TOTAL	11129	11129

The balances of accounts which include only debit or credit entries are given by the totals of the rows and columns, respectively; the computer is programed to calculate these balances, and a further instruction defines them by reference to their respective cell locations. For example

KACREC = KASYS (2, 8)

indicates where the balance of Accounts Receivable Account is to be found when it is needed for reference or further processing (namely the cell formed by row 2 and column 8).

The balance of an account which appears in both a row and a column is obtained by extending the instructions so as to define the balance as the difference between total debits and total credits, as follows:

KCASHB = KCASHD − KCASHC

where KCASHD has previously been defined as the sum of the debits to Cash Account, as recorded in cell 1, 8 of the ledger matrix and KCASHC has been defined as the sum of the credits in cell 11, 1.

Individual accounts

In the program illustrated in Appendix D on page 42, separate DIMENSION, FORMAT, READ, and WRITE instructions have been included in respect to individual accounts receivable and accounts payable, which thus constitute subsidiary ledgers controlled in the usual way by control accounts in the general ledger. Separate data cards are then provided for accounts receivable and accounts payable. Each debtor's or creditor's account has been designed to record the balance at the beginning of each week, a sufficient number of debit and credit entries (these may be extended as necessary, for example, to provide for separate entries for each day of the week), and the balance at the end of the week (see Figure 4 above). The program could easily be generalized to provide for additional accounts receivable or accounts payable.

At the end of the week a trial balance is automatically printed out in accordance with programed instructions which specify the format of the trial balance, define the items which are recorded in it, and arrange for the calculation of the total debits and total credits (see Figure 5, this page).

The computer is programed to compare the total debits and the total credits in the trial balance

The complete program, illustrated in Appendix D, could easily be extended....

and to print out an error message if the totals do not agree. Further instructions ensure that the balances in the accounts receivable and accounts payable accounts are compared with the sums of the balances in the accounts receivable and accounts payable ledgers (see Figure 4), error messages being printed out if the respective amounts do not agree. Since the ledger matrix for the general ledger and the matrices for the two subsidiary ledgers have been posted from different sets of data cards, this provides a built-in control over the subsidiary ledgers in a manner that is analogous to the control achieved in a conventional accounting system. The program could easily be extended to provide for other subsidiary ledgers.

The program which is illustrated in Appendix D provides for the insertion of a separate data card with each week's transactions to indicate the number of the week (in practice separate data cards would probably be inserted with each day's transactions). At the end of each week, as we have seen, the program provides for a print-out of the trial balance and the subsidiary ledger matrices (this could be done on a daily basis if desired), but the program also provides for the preparation of an income statement and balance sheet at the end of each four weeks' transactions. An additional control statement ensures the accuracy of the program in relation to this task. The print-out in respect of Week 4 is illustrated in Figure 6 below.

The complete program is illustrated in Appendix D. It could easily be extended to provide for a more elaborate accounting system. It was written by the author and run on an IBM 7094 computer at the Western Data Processing Center, Graduate School of Business Administration, University of California, Los Angeles, in December, 1966.

It has been argued in this paper that effective utilization of the computer in accounting applications requires thought to be given, not only to the analysis and design of business systems in relation to particular data processing operations which form part of the accounting function, but also to the design of the accounting system itself. In particular, it is suggested that, for purposes of computer programing, a ledger in matrix form needs to be substituted for the traditional double-entry framework of ledger accounts. A ledger matrix offers scope for the development of a computer program which minimizes the input and storage difficulties which are associated with disaggregated data sources, while permitting maximum use to be made of the computer's qualities with respect to computation, repetitive data processing, automatic control, and the printing of all output.

FIGURE 6

LEDGER MATRIX

	CASH	INVEN	ACDEP	ACPAY	JSCAP	JSCUR	SALES	TOTDR
CASH	0	0	0	-1563	5000	0	3546	6983
ACREC	-185	0	0	0	0	0	3377	3192
INVEN	0	0	0	6467	0	0	0	6467
FIXED	1200	0	0	0	0	0	0	1200
COFGS	0	4426	0	0	0	0	0	4426
RENT	200	0	0	0	0	0	0	200
WAGES	760	0	0	0	0	0	0	760
DEPRC	0	0	10	0	0	0	0	10
OTHEX	126	0	0	0	0	0	0	126
JSDWG	186	0	0	0	0	0	0	186
TOTCR	2287	4426	10	4904	5000	0	6923	23550

TRIAL BALANCE
WEEK 4

	DR	CR
CASH	4696	
ACREC	3192	
INVEN	2041	
FIXED	1200	
COFGS	4426	
RENT	200	
WAGES	760	
DEPRC	10	
OTHEX	126	
JSDWG	186	
ACDEP		10
ACPAY		4904
JSCAP		5000
JSCUR		0
SALES		6923
TOTAL	16837	16837

INCOME STATEMENT
WEEK 4

SALES		6923
COFGS		4426
GROSP		2497
RENT	200	
WAGES	760	
DEPRC	10	
OTHEX	126	
TOTEX		1096
NETPR		1401

BALANCE SHEET
WEEK 4

ASSETS		
CASH		4696
ACREC		3192
INVEN		2041
FIXED	1200	
ACDEP	10	
NETFIX		1190
TOTASS		11119
ACPAY		4904
JSCAP	5000	
JSCUR	1215	
OWNEQ		6215

AC RECEIVABLE
WEEK 4

	BEGBAL	DR 1	DR 2	DR 3	CR	ENDBAL
C.CARTER	151	49	48	0	-60	188
J.JACKSON	580	104	0	0	-125	559
N.NORTON	679	140	0	0	0	819
P.PARKER	243	34	0	0	0	277
R.ROYCE	248	68	148	0	0	464
W.WILLIAMS	595	86	204	0	0	885
TOTAL	2496	481	400	0	-185	3192

AC PAYABLE
WEEK 4

	BEGBAL	DR	CR	ENDBAL
B.BARKER	903	-575	41	369
G.GRACE	781	-568	143	356
H.HOWARD	929	-420	0	509
M.MOSS	1382	0	202	1584
S.STEWART	824	0	92	916
T.THOMPSON	1170	0	0	1170
TOTAL	5989	-1563	478	4904

APPENDIX A

Transactions — First Week

19—			
Feb. 1	Deposited $5,000 capital in bank		
	Paid January rent $200		
	Purchased shop fittings $1,200 for cash		
2	Purchased trading inventories as follows (credit purchases):		
	T. Thompson	$600	
	S. Stewart	450	
	B. Barker	575	
	G. Grace	365	
	H. Howard	420	
	M. Moss	785	$3,195
	Credit sales (cost of goods sold $195):		
	J. Jackson	$ 85	
	P. Parker	20	
	W. Williams	40	
	N. Norton	95	
	C. Carter	45	$ 285
	Cash sales (cost $590):		$1,100
3	Credit sales (cost $106):		
	J. Jackson	$ 40	
	C. Carter	15	
	W. Williams	95	$ 150
	Cash sales (cost $100):		$ 140
4	Credit purchases:		
	M. Moss	$283	
	G. Grace	203	
	H. Howard	141	
	S. Stewart	290	$ 917
	Cash sales (cost $141):		$ 204
Feb. 5	Credit sales (cost $114):		
	J. Jackson	$ 94	
	N. Norton	43	
	W. Williams	49	$ 186
	Purchase returns — M. Moss $48		
	Wages paid		$ 190
	Proprietor's drawings		$ 60

APPENDIX B

Journals — First Week

Cash Receipts Journal

19—	Particulars	Computer posting	Accounts rec.	Cash sales	Other receipts	Total	Bank
			IASYS —(2, 1)	IASYS (1, 7)	—	—	—
Feb. 1	J. Smith	IASYS(1, 5)			5,000	5,000	5,000
2	Sales			1,100		1,100	1,100
3	Sales			140		140	140
4	Sales			204		204	204
5	Total		—	1,444	5,000	6,444	6,444

Cash Payments Journal

19—	Particulars	Computer posting	Accounts payable	Invent.	Fixed assets	Rent	Wages	Other exp.	J. Smith draw.	Total	Bank
			IASYS —(1, 4)	IASYS (3, 1)	IASYS (4, 1)	IASYS (6, 1)	ISAYS (7, 1)	IASYS (9, 1)	IASYS (10, 1)	—	—
Feb. 1	Rent					200				200	200
	Fixed assets				1,200					1,200	1,200
5	Wages						190			190	
	J. Smith drawings								60	60	250
	Total				1,200	200	190		60	1,650	1,650

Credit Sales Journal

19—	Particulars	Computer posting		Amount
Feb. 2	J. Jackson	IREC	(2,2)	85
	P. Parker	IREC	(4,2)	20
	W. Williams	IREC	(6,2)	40
	N. Norton	IREC	(3,2)	95
	C. Carter	IREC	(1,2)	45
3	J. Jackson	IREC	(2,3)	40
	C. Carter	IREC	(1,3)	15
	W. Williams	IREC	(6,3)	95
5	J. Jackson	IREC	(2,4)	94
	N. Norton	IREC	(3,3)	43
	W. Williams	IREC	(6,4)	49
	Total	IASYS	(2,7)	621

Cost of Goods Sold Journal

19—	Particulars	Computer posting		Amount
Feb. 2	Credit Sales			195
	Cash Sales			590
3	Credit Sales			106
	Cash Sales			100
4	Cash Sales			141
5	Credit Sales			114
	Total	IASYS	(5,2)	1,246

Credit Purchases Journal

19—	Particulars	Computer posting		Amount
Feb. 2	T. Thompson	IPAY	(6,3)	600
	S. Stewart	IPAY	(5,3)	450
	B. Barker	IPAY	(1,3)	575
	G. Grace	IPAY	(2,3)	365
	H. Howard	IPAY	(3,3)	420
	M. Moss	IPAY	(4,3)	785
4	M. Moss	IPAY	(4,3)	283
	G. Grace	IPAY	(2,3)	203
	H. Howard	IPAY	(3,3)	141
	S. Stewart	IPAY	(5,3)	290
5	M. Moss	IPAY	—(4,3)	—48
	Total	IASYS	(3,4)	4,064

General Journal
No Transactions in First Week

APPENDIX C

Transaction Tables First Week

IASYS	1	2	3	4	5	6	7	8
1					5,000		1,444	
2							621	
3				4,064				
4	1,200							
5		1,246						
6	200							
7	190							
8								
9								
10	60							

IREC	1	2	3	4	5	6
1		45	15			
2		85	40	94		
3		95	43			
4		20				
5						
6		40	95	49		

IPAY	1	2	3	4
1			575	
2			568	
3			561	
4			1,020	
5			740	
6			600	

Computer Program for Accounting System with General Ledger and Two Subsidiary Ledgers

APPENDIX D

COMPUTER PROGRAM FOR ACCOUNTING SYSTEM WITH GENERAL LEDGER AND TWO SUBSIDIARY LEDGERS

```
C FORTRAN IV PROGRAM FOR ACCOUNTING SYSTEM WITH GENERAL LEDGER AND TWO
C SUBSIDIARY LEDGERS IN MATRIX FORM
C DIMENSIONS OF MATRICES USED FOR POSTING GENERAL LEDGER(IASYS),STORING
C GENERAL LEDGER(KASYS),ACCOUNTS RECEIVABLE LEDGER(IREC) AND ACCOUNTS
C PAYABLE LEDGER(IPAY)
      DIMENSION IASYS(11,8),KASYS(11,8),IREC(7,6),IPAY(7,4)
C INPUT FORMATS FOR GENERAL LEDGER,ACCOUNTS RECEIVABLE AND ACCOUNTS
C PAYABLE LEDGERS
1     FORMAT(8I6)
2     FORMAT(6I6)
3     FORMAT(4I6)
C OUTPUT FORMAT FOR GENERAL LEDGER MATRIX
4    0FORMAT(1H1,28X,13HLEDGER MATRIX/,7X,48H  CASH INVEN ACDEP ACPAY JS
     1CAP JSCUR SALES TOTDR/,7H CASH  ,8I6/,7H ACREC ,8I6/,7H INVEN ,8I6
     2/,7H FIXED ,8I6/,7H COFGS ,8I6/,7H RENT  ,8I6/,7H WAGES ,8I6/,7H D
     3EPRC ,8I6/,7H OTHEX ,8I6/,7H JSDWG ,8I6/,7H TOTCR ,8I6)
C INPUT FORMAT TO IDENTIFY WEEK NUMBER
5     FORMAT(I2)
C OUTPUT FORMATS FOR TRIAL BALANCE,INCOME STATEMENT,BALANCE SHEET,ERROR
C MESSAGES,ACCOUNTS RECEIVABLE AND ACCOUNTS PAYABLE SCHEDULES
6    0FORMAT(1H1,7X,13HTRIAL BALANCE/,11X,5HWEEK ,I2/,10X,2HDR,5X,2HCR/,
     17H CASH  ,I6/,7H ACREC ,I6/,7H INVEN ,I6/,7H FIXED ,I6/,7H COFGS ,
     2I6/,7H RENT  ,I6/,7H WAGES ,I6/,7H DEPRC ,I6/,7H OTHEX ,I6/,7H JSD
     3WG ,I6/,7H ACDEP ,7X,I6/,7H ACPAY ,7X,I6/,7H JSCAP ,7X,I6/,7H JSCU
     4R ,7X,I6/,7H SALES ,7X,I6/,7H TOTAL ,I6,1X,I6)
7    0FORMAT(1H1,7X,16HINCOME STATEMENT/,11X,5HWEEK ,I2/,7H SALES ,8X,I6
     1/,7H COFGS ,8X,I6/,7H GROSP ,8X,I6/,7H RENT  ,2X,I6/,7H WAGES ,2X,
     2I6/,7H DEPRC ,2X,I6/,7H OTHEX ,2X,I6/,7H TOTEX ,8X,I6/,7H NETPR ,8
     3X,I6)
8    0FORMAT(1H ,9X,13HBALANCE SHEET/,11X,5HWEEK ,I2/,7H ASSETS/,7H CASH
     1  ,8X,I6/,7H ACREC ,8X,I6/,7H INVEN ,8X,I6/,7H FIXED ,2X,I6/,7H AC
     2DEP ,2X,I6/,7H NETFIX,8X,I6/,7H TOTASS,8X,I6/,7H ACPAY ,8X,I6/,7H
     3JSCAP ,2X,I6/,7H JSCUR ,2X,I6/,7H OWNEQ ,8X,I6)
9     FORMAT(1H1.19HTRIAL BALANCE ERROR)

10    FORMAT(1H1,19HBALANCE SHEET ERROR)
11   0FORMAT(1H1,17X,13HAC RECEIVABLE/,18X,5HWEEK ,I2/,11X,36HBEGBAL  DR
     1 1  DR 2  DR 3   CR ENDBAL/,11H C.CARTER  ,6I6/,11H J.JACKSON ,6I6
     2/,11H N.NORTON  ,6I6/,11H P.PARKER  ,6I6/,11H R.ROYCE   ,6I6/,11H
     3W.WILLIAMS,6I6/,11H TOTAL      ,6I6)
12   0FORMAT(1H1,16X,10HAC PAYABLE/,17X,5HWEEK ,I2/,11X,24HBEGBAL   DR
     1  CR ENDBAL/,11H B.BARKER  ,4I6/,11H G.GRACE   ,4I6/,11H H.HOWARD
     2 ,4I6/,11H M.MOSS    ,4I6/,11H S.STEWART ,4I6/,11H T.THOMPSON,4I6/
     3,11H TOTAL     ,4I6)
13    FORMAT(1H1,24HRECEIVABLE CONTROL ERROR)
14    FORMAT(1H1,21HPAYABLE CONTROL ERROR)
C CLEARING,POSTING AND ACCUMULATING INSTRUCTIONS
15    FORMAT(I1)
      DO50J=1,8
      DO50I=1,11
      KASYS(I,J)=0
50    CONTINUE
      DO51J=1,6
      DO51I=1,7
      IREC(I,J)=0
51    CONTINUE
      DO52J=1,4
      DO52I=1,7
      IPAY(I,J)=0
52    CONTINUE
      READ(5,15)N


      DO25K=1,N
      READ(5,1)((IASYS(I,J),J=1,8),I=1,11)
      DO26I=1,11
      DO26J=1,8
      KASYS(I,J)=KASYS(I,J)+IASYS(I,J)
26    CONTINUE
25    CONTINUE
      READ(5,2)((IREC(I,J),J=1,6),I=1,7)
      READ(5,3)((IPAY(I,J),J=1,4),I=1,7)
      READ(5,5)IDATE
      DO53J=1,7
      DO53I=1,10
      KASYS(I,8)=KASYS(I,8)+KASYS(I,J)
```

```
53    CONTINUE
      DO54J=1,8
      DO54I=1,10
      KASYS(11,J)=KASYS(11,J)+KASYS(I,J)
54    CONTINUE
C INSTRUCTIONS TO DEFINE TRIAL BALANCE AND SUBSIDIARY LEDGER VALUES
      KCASHD=KASYS(1,8)
      KACREC=KASYS(2,8)
      INVEND=KASYS(3,8)
      KFIXED=KASYS(4,8)
      KCOFGS=KASYS(5,8)
      KRENT=KASYS(6,8)
      KWAGES=KASYS(7,8)
      KDEPRC=KASYS(8,8)
      KOTHEX=KASYS(9,8)
      JSDWG =KASYS(10,8)
      KTOTDR=KASYS(11,8)
      KCASHC=KASYS(11,1)
      INVENC=KASYS(11,2)
      KACDEP=KASYS(11,3)
      KACPAY=KASYS(11,4)
      JSCAP =KASYS(11,5)
      JSCUR =KASYS(11,6)
      KSALES=KASYS(11,7)
      KCASHB=KCASHD-KCASHC
      INVENB=INVEND-INVENC
     0KTBTDR=KCASHB+KACREC+INVENB+KFIXED+KCOFGS+KRENT+KWAGES+KDEPRC+KOTH
     1EX+JSDWG
      KTBTCR=KACDEP+KACPAY+JSCAP+JSCUR+KSALES
      IF(KTBTDR.NE.KTBTCR)WRITE(6,9)
      DO55I=1,6
      DO55J=1,5
      IREC(I,6)=IREC(I,6)+IREC(I,J)
55    CONTINUE
      DO56J=1,6
      DO56I=1,6
      IREC(7,J)=IREC(7,J)+IREC(I,J)
56    CONTINUE
      DO57I=1,6
      DO57J=1,3
      IPAY(I,4)=IPAY(I,4)+IPAY(I,J)
57    CONTINUE
      DO58J=1,4
      DO58I=1,6
      IPAY(7,J)=IPAY(7,J)+IPAY(I,J)
58    CONTINUE
      IF(KACREC.NE.IREC(7,6))WRITE(6,13)
      IF(KACPAY.NE.IPAY(7,4))WRITE(6,14)
C OUTPUT INSTRUCTIONS FOR LEDGER MATRIX,TRIAL BALANCE AND SUBSIDIARY
C LEDGER SCHEDULES AT END OF EACH WEEK
      WRITE(6,4)((KASYS(I,J),J=1,8),I=1,11)
     0WRITE(6,6)IDATE,KCASHB,KACREC,INVENB,KFIXED,KCOFGS,KRENT,KWAGES,KD
     1EPRC,KOTHEX,JSDWG,KACDEP,KACPAY,JSCAP,JSCUR,KSALES,KTBTDR,KTBTCR
      WRITE(6,11)IDATE,((IREC(I,J),J=1,6),I=1,7)
      WRITE(6,12)IDATE,((IPAY(I,J),J=1,4),I=1,7)
      IF(N.NE.4)GO TO 59
C INSTRUCTIONS TO DEFINE INCOME STATEMENT AND BALANCE SHEET VALUES AND
C TO PRINT INCOME STATEMENT AND BALANCE SHEET AT END OF EACH 4-WEEK
C PERIOD
      KGROSP=KSALES-KCOFGS
      KTOTEX=KRENT+KWAGES+KDEPRC+KOTHEX
      KNETPR=KGROSP-KTOTEX
      JSCUB =JSCUR+KNETPR-JSDWG
      KTOTAS=KCASHB+KACREC+INVENB+KFIXED-KACDEP
      KOWNEQ=JSCAP+JSCUB
      NETFIX=KFIXED-KACDEP
      IF((KTOTAS-KACPAY).NE.KOWNEQ)WRITE(6,10)
     0WRITE(6,7)IDATE,KSALES,KCOFGS,KGROSP,KRENT,KWAGES,KDEPRC,KOTHEX,KT
     1OTEX,KNETPR
     0WRITE(6,8)IDATE,KCASHB,KACREC,INVENB,KFIXED,KACDEP,NETFIX,KTOTAS,K
     1ACPAY,JSCAP,JSCUB,KOWNEQ
59    STOP
      END
```