1984

General ledger system

Chung-Chieh Wu

The University of Montana

Follow this and additional works at: https://scholarworks.umt.edu/etd
Let us know how access to this document benefits you.

Recommended Citation
https://scholarworks.umt.edu/etd/2432

This Thesis is brought to you for free and open access by the Graduate School at ScholarWorks at University of Montana. It has been accepted for inclusion in Graduate Student Theses, Dissertations, & Professional Papers by an authorized administrator of ScholarWorks at University of Montana. For more information, please contact scholarworks@mso.umt.edu.
COPYRIGHT ACT OF 1976

This is an unpublished manuscript in which copyright subsists. Any further reprinting of its contents must be approved by the author.

Mansfield Library
University of Montana
Date: 1984
The general ledger system

by Chung-Chieh Wu

Presented in partial fulfillment of the requirements for
the degree of Master of Science
University of Montana
1984

Approved by:

[Signature]
Chairman

[Signature]
Dean, Graduate School

June 13, 1984
Date
The general ledger system is the heart of an accounting system. It is a record of the balance of each of the accounts of a business. At the end of an accounting period, the system is used to produce financial statements to show the results of the business activities of that period.

This general ledger system was developed by following all of the phases of development for a large software system. The system design was developed after the requirement and the analysis specification was approved by the thesis committee. The Problem Specification Language / Problem Statement Analysis (PSL/PSA) was used in the analysis phase to produce the specification. This particular system is designed by structural design methodology. The design documentation was produced through the use of Program Design Specification Language (PDSL).

The programs were written by following the design specification. It was decided to use COBOL to implement this project. The programs were tested during the implementation phase.

During the progress of this long and tedious process, many people have contributed toward my thesis. First of all, my thesis committee: Dr. John Barr for helping me utilize the techniques and tools in software development; Professor Gene Schiedermayer for helping me to know more about the general ledger system; Dr. George McRae for his comments on my thesis. Secondly, the following friends encouraged and helped me enormously; Raj, Rama, Philip, Scott, Trish. Finally, Olive, my wife's superior support is unforgettable and appreciated.
# Table of Contents

Abstract ................................................................. ii

List of Figures .......................................................... iv

Chapter I  Introduction ................................................ 1

Chapter II General Ledger System ................................. 5

Chapter III Analysis .................................................... 11

Chapter IV Design ....................................................... 18

Chapter V Implement .................................................... 25

Chapter VI Conclusion ................................................ 30

Bibliography ............................................................... 33

Appendix-A Specification ............................................. A-1

Appendix-B Design Documentation ................................. B-1

Appendix-C Programs .................................................... C-1

Appendix-D User's Manual ............................................. D-1
List of Figures

Figure 4-1 .................................................. 21
Figure 4-2 .................................................. 21
Data Flow Diagram Level-Context .......................... A-1
Data Flow Diagram Level-0 .................................. A-7
Data Flow Diagram Level-1 .................................. A-20
Data Flow Diagram Level-2 .................................. A-36
Data Flow Diagram Level-3 .................................. A-52
Data Flow Diagram Level-3.2 ............................... A-75
Design Structure Chart 1 ................................. B-15
Design Structure Chart 2 ................................. B-20
Design Structure Chart 3 ................................. B-28
Design Structure Chart 4 ................................. B-35
Design Structure Chart 5 ................................. B-41
Design Structure Chart 6 ................................. B-50
CHAPTER 1

INTRODUCTION

According to *Software Engineering Economics* by Barry W. Boehm, in the past thirty years the cost of hardware has decreased and the cost of software has increased. The increased requirement for efficient and complex software over the last three decades is mainly responsible for this situation. The progress in the advancement of software technology compared to requirements is very slow, the need for more efficient tools and methods of software technology is difficult to meet. The development and use of new software engineering tools and methods is a logical way of resolving this problem. Experience indicates that efficient software products can be developed by using these methods and tools.

This project is an exercise in software engineering covering the specification, design, and implementation of a model general ledger system. The software technology tools: Problem Specification Language / Problem Statement Analysis (PSL/PSA) and Problem Design Specification Language (PDSL); and methods: structured analysis and structured design; were utilized in this project.
Normally, the products of a software development process include specification, design and implementation. The specification is the product of problem formulation and problem analysis phases. The problem formulation phase describes the problem of the system without detail. Then, the problem definition is refined to supply essential detailed specifications during the problem analysis phase. A set of potential solutions to the problems can be identified through the study of the specifications. The potential solutions are evaluated and compared to the alternatives until the best solution is obtained. The design documentation describes the chosen solution in detail. Referencing the design documentation, the final product is constructed in the implementation phase.

This project will be approached by following the software engineering steps. Acquiring knowledge about accounting system to identify and define the problems in it is the first step. The accounting cycle and functions of general ledger system are known from this step. Interviews with the user help in defining the general ledger system and ensuring that his requirements are incorporated. This newly defined general ledger system will be represented in the data flow diagram. The specification of a model general ledger system will be produced after interviews with user. It is described by using Problem Specification Language / Problem Statement Analysis (PSL/PSA).
Referencing the data flow diagram and specification, the specified system is designed in a structured manner.

Structured design is a method to obtain maintainable, modifiable, and implementable system designs. The Program Design Specification Language (PDSL) will be used to describe the design. The design documentation is used in the implementation phase. The user manual will be written simultaneously when the system is implemented in COBOL. The testing will be done from the beginning of coding until the final integration of whole system.

The following chapter describes what a general ledger system is and what it should do according to user requirements. The analysis, design, and implementation phases will be discussed in succeeding chapters. The problems will be discussed in the last chapter. It also will describe solutions to those problems.

The specifications which are laid out in the Data Flow Diagram and formatted problem statement of the Problem Specification Language / Problem Statement Analysis (PSL/PSA) are in appendix A. The design documentation which contains the structured design chart and the formatted statement of Program Design Specification (PDSL) are in appendix B. Appendix C is the system implementation in COBOL code. The user manual is in appendix D.
The task of accounting is to track the financial events in the life of any entity (individual or organization) in a manner that makes it possible for that individual or organization to report on its financial position and activities to anyone who may be interested. Therefore, accounting deals with financial events which are very important information for an entity. Anytime during the life of a business, an accounting system must be prepared to provide many different types of information about the activities of that business.

The transactions of daily business events are entered in an accounting record or journal to provide up-to-date financial information about a business. Because all transactions are written in the journal, it can be treated as a book of original entries. There are several types of journals to accomplish the recording function of accounting process. Ideally, we want a "daily" that gives the chronological history of that business by recording the important accounting events. It also is required to record these transactions in a way that makes it easy to recognize the effect of events on assets, liabilities, owner's equity, revenue, and expenses of the business.
The journal records raw data for the system. It is only the beginning of the accounting cycle. In order to produce useful reports about the business at the end of any accounting period, an accounting system must be able to determine the total effect of all transactions in a period on the accounts of the business. During the second step in the accounting cycle the transactions are posted to a ledger.

The classification function is accomplished by the ledger. The ledger is used to separate components of each transaction and group those components that affect each individual asset, liability, owner's equity, revenue, or expense account. So, the ledger can show the cumulative effect of all transactions that affected a particular account.

Three important characteristics of ledgers are:

1. The ledger focuses on the individual accounts of a business. For example, there may be 50 or more transactions affecting cash, spread throughout the journals. The ledger enables the net change in an account to be determined by accounting for all transactions that affect that account.

2. Posting to the ledger usually takes place at the end of an accounting period. Because it is important to know
the net effect on all assets, liabilities, owner's equity, revenues, and expenses for a series of events. But it is not necessary to know the effect of each individual transaction as it occurs.

3. Posting to the ledger simply reproduces information already recorded in the journal. It does not produce new information. It only shows the cumulative effect of transactions on individual accounts.

The third step in the accounting cycle is called the trial balance. The value of an account shows in the balance. Businesses need to know the dollar value of each account at various times in order to be assured that the basic accounting equation (Assets = Liabilities + Owner's Equity) is still in balance. The informal internal statement, called a trial balance, is usually created when these balances are determined. The trial balance is simply a list of all accounts used by the business and the dollar amount in each account. It can give an overview of the accounts, their amounts, and make sure the basic accounting equation still holds.

The above steps may be performed continuously, frequently, or only once during an accounting period. The end-of-period activities occur only at end of each accounting period. Business events can be categorized into two parts: external events and internal events. External
events take place between a business and individuals. When the external events occur, they are usually documented and recorded by either or both parties. As mentioned above, steps one to three of the accounting cycle are concerned only with external events. In step one the external events are recorded and in step two they are posted. The trial balance in step three is produced from those external events.

The internal events are another class of transactions which occur in the life of a business. They are more difficult to grasp than external events. Because they may happen within the firm, and there may not be any individual or other business directly involved in the transaction. Also, these events occur constantly rather than at specific, identifiable points in time, and there is rarely a document to support the changes made through transactions. Because of this, a series of entries called adjusting entries must be recorded at the end of the period to accumulate the effect. Hence, before financial statements are prepared at the end of the accounting cycle, all the events, external and internal, should be properly recorded.
Financial statements are prepared in step four. Most users like to have two basic statements, a balance sheet and an income statement. The balance sheet lists the possessions of the business and the obligations the business faces. It reflects the equality condition of the basic accounting equation. It is a picture of the business depicting its financial position as of any one instance. For conveying information about the nature of individual assets, liabilities, and owner's equity, the balance sheet is prepared according to the above mentioned classifications.

The income statement is another primary statement which is produced by the accounting system. It shows the ultimate effect on the business of its profit-seeking activities during a specific period. The net income, net profit, or net earning is shown on the bottom line of this statement. The balance sheet and income statement are related to each other. The two statements interact together through retained earnings. Net income minus the amount of those assets withdrawn by the owners, represent the net amount by which retained earning is increased for the period.
The last step in accounting cycle is closing entries. Only the revenue and expense accounts and retained earnings account have to be closed. No other accounts are affected by this final step. The revenue and expense accounts are created temporarily to calculate the retained earning which resulted from the profit-seeking activities of a business. In the beginning of a new period, these revenue and expense accounts will have to be set to zero. In general, a computerized general ledger system includes most of the functions mentioned above, except the routine recording job (ie, the journal).
CHAPTER 3
ANALYSIS

The analysis is the most important step in the system development life cycle. A successful analysis can decrease the expenditure to obtain a desired result in the future. A system analyst, having knowledge about the system and the user needs, acts as a bridge between them. Thus he helps in the efficient operation of the system. Detailed documentation is developed during this phase, which subsequent phases can use to evaluate whether the target has been attained. It is the medium of communication between user and developer.

During the analysis phase, I spent most of the time in identifying the user's requirements. In the beginning, I created the physical data flow diagram of the system from general accounting reference [2]. This was presented to the user to make sure it was correct. Through discussions with the user, the requirements were understood by using the data flow diagram. Then, the logical data flow diagram was produced and shown to the user to assure that all his requirements were included in the diagram.
After interviewing the user, the system requirements were identified. According to the requirements of the user, the General Ledger System should collect the entries from the Account Receivable, Account Payable and Payroll accounting subsystems at the end of each period. The collected data will be entered to the system manually using a terminal. Entries will not be accepted if they have any error. The system accepts the valid entry and sends it to the entered account file. After finishing the entries, the user can request a Trial Balance Sheet which can be referenced to determine the adjustment entries. These adjustment entries will be entered to the system. A new Trial Balance sheet will be produced to make sure the accounts are in balance. If the entered accounts are balanced, then the entered entries are posted to update the general ledger account. The current period's Income Statement can be obtained before the revenues expenses accounts of the general ledger file are set to zero for the next period. After printing the Income Statement, the total income of this period will be moved to a capital account. The Balance Sheet can be obtained after the end of period process.
In addition to the above functions, the user would like to have a ledger sheet showing the flow in each account in a period of time. A chart of accounts showing the current accounts in the system is also desired by the user. He also wants to have an audit trail which shows the trail of entered accounts for a certain period. The user desires a list of entered accounts for the current period. This should be produced before the entered account file is posted to the previous entered account file and cleaned.

The system must allow the general ledger accounts to be maintained in terms of addition, deletion or modification. Maintenance must be double checked to make sure it is correct.

In the beginning, the control flow was sometimes shown on the diagrams, which was not correct. Thus, in the data flow diagrams I tried to show the flow of data, not the control. This helped to design a less coupled system in the next phase. It is a major function to produce the useful reports of the general ledger system. Thus, during identification of the data flows, it is important that the data to the system should be sufficient enough to produce the reports. The data of input and output were discussed in detail to make sure the information was available to perform the required functions. Because the detailed input can not be totally decided before knowing the required reports, the report formats were determined before trying
to identify the detailed input. After the interviews, the contents of the data flow were recognized. At the same time, the data flow diagram was refined many times to get the best version.

The Data Flow Diagram and changes made to match the requirements of user is shown in appendix A. There are four major components in a Data Flow Diagram. First, the Data Flow: It portrays some interface among components of the Data Flow Diagram. Most data flows move between processes. Second, the Process: It shows some amount of work performed on data. Third, the File: It is a temporary repository for data. It may be a tape, or an area of disk, or a card data set. Fourth, the Source or Sink: The source and sink show where the net input is coming from and the net output is going to. It may be a person or organization, lying outside the context of a system, acting as a originator or receiver of the system data.

The Data Flow Diagram is easy to read and conceptualize. Instead of showing a very wordy and complex system, the Data Flow Diagram partitions the user requirements down to components. It shows the major divisions of functions, and all the interfaces among the components. Communication with the user is made simple through the compartmentalization of the Data Flow Diagram.
The Problem Statement Language / Problem Statement Analyzer was used to describe the system according to the Data Flow Diagram. PSL/PSA was used after the above steps. The leveled data flow diagram was described from the top of the hierarchy to the bottom. Each time I added only one page's description of the data flow diagram to the PSL/PSA data base. Before integrating another page's description, all the errors caused by the current page's description were corrected. This method only corrected the new errors which were caused by the recently added part and it made the integration easier and more correct.

The PROCESS in the Problem Statement Language is defined by specifying the information upon which it operates and the information which it produces. In the view of Problem Statement Language, every bubble in the Data Flow Diagram is a PROCESS. The top bubble, which is the highest level can be regarded as the total target system. The Data Dictionary can then be described easily through the use of ELEMENT, ENTITY, GROUP, SET, INPUT and OUTPUT in the Problem Specification Language. Besides, in each defined PROCESS, the analyst can describe the procedure for each bottom level of the data flow diagram in structured English (minispec). After the described system is put into the Problem Statement Language data base, the analyst can obtain many analysis reports through the Problem Statement Analyzer. The Formatted Statement
command combined with the parameter TRACE-KEY in Name Selection can produce the partitioned target system Formated Statement reports. The Partitioned report is a representation of a leveled Data Flow Diagram.

The formats of the reports were described in the "layout" of the OUTPUT in the PSL. Because only seventy three columns were accepted by the "IPSL" command of the PSA, every description line of reports which are more than seventy three columns must be separated into two lines to meet the constraint. When trying to get the reports' layout, the width parameter for the "layout" command in PSA had to be more than seventy three columns. Then the layout reports would be created correctly. All the margins of this analysis specification had to be adjusted to match the constraint of this thesis. The consideration of margin was trifling, but it would reduce the trouble when changing it in the future, especially when changing the source code.

From the above brief description of the Problem Statement Language, we can find that it contains three features that help to describe the system. First, it is easy to get a partition documentation which corresponds to the Data Flow Diagram. Second, the minispec is a tool which is used to document the internal details of the Data Flow Diagram in a structured English fashion. Third, the data dictionary is clearly described through the use of Problem Statement Language.
The purpose of system design is to generate a model to reflect the primary functions which the user expects from the system. Usually, to begin the design, the designer reads the system specification which is approved by the analyst to make sure that the design will follow the requirements. But in the specification, only the system inputs, outputs and minispecs are available to make decisions for the system design. Most problems in the system must be identified in this phase, like how many subsystems and modules are needed to accomplish the user's requirements and how those subsystems, modules are interconnected.

The whole system's major characters will determined by this phase. How many subsystems in this system, how many modules are needed to reach the expected functions for each subsystems. After choosing the best allocation of all components for the system, the design documentation (which has precise specifications of individual modules) will be produced and passed to the programmer. Usually, it must include information about inputs, outputs, interfaces with other parts of the system as well as the algorithms.
The "structured design" guidelines are used to help solve the above problems. The collection of guidelines and techniques for structured design have proved that these lead to a better design. For each module, seven subordinate modules proved to be manageable. The smaller the size of each module, the less complicated the functions for each module. The coupling is a measure of the relations between modules. It may be measured by the number of connections, type of connection and type of information contained within those connections. The simpler and fewer the interrelationships between modules, the lower measure of coupling. The cohesion is a measure of the type of relationships that exist between elements in the same module. The stronger cohesion of a module, the more the module can be treated as a single unit. Thus, the higher the cohesion of the module the less the coupling of the module.

According to those guidelines, the less coupling the system, the better. It ideally means that one module can be studied without knowing very much about any other modules in the system. It should be possible to change one module without changing the other modules. Consideration of the type and complexity of interfaces between modules will help to produce a less coupled system. On the other hand, the lower the coupling of the system the stronger the cohesion of individual modules in the system. The Data
Flow Diagram, which is accepted from the analysis phase, is used to figure out the flow of the data through the system. Beginning from the top level, a trivial Data Flow Diagram is then refined by segmenting the whole system down to the bottom. However, this diagram does not show the loops, terminations or decisions.

Second, the trivial diagram is used to try to identify afferent and efferent data. The afferent data elements are those elements of data which are inputs to the system. The process starts by reading from the physical input into the system along the Data Flow Diagram. Beginning at the other end of the system, the outputs are identified. The reason for identifying the efferent data elements from the back, is that the efferent data elements are those farthest removed from the input. Then, the system is separated into three parts: afferent, efferent and transforms. A main module is specified which performs the entire task of the system by calling upon the three sub-ordinates. The main module is the overall control for the process. The controls and coordinates of the efferent, afferent and transform modules are handled by this module. Fig. 4-1 is a simple example of the sectioned data flow diagram. Fig. 4-2 is the resulting structure chart from Fig. 4-1. There may be different ways to separate the data flow diagram. Differently sectioned data flow diagrams will result in the different types of design.
structure chart.

FIGURE 4-1

FIGURE 4-2
Each subordinate module is factored separately. The structure chart will be used to present the factored system. It helps to define and identify subordinate modules. In this step, it is preferable to go too far in an initial factoring, and to have recognized processes that are too small, too fractional, and too specialized to constitute distinct modules. Those very tiny modules can be combined later with a fuller understanding of the design structure; hopefully an optimal choice can be made. If the design process does not go far enough, the opportunity for conscious decision may be lost.

After the final version of the system design is decided, the Program Design Specification Language (PDSL) can be used to describe the properly allocated modules of this system in detail. Through the description of PDSL, the design specification (please reference the appendix B) can be obtained. The design specification will be used for implementation in the next system development life cycle. Thus, it is better to pay more attention to this phase than to find errors in the next phase.

In the beginning, the afferent, transform, and efferent portions of system were decided. The functions to accept the entry, adjust the entry, and the general ledger file maintenance were all included in the afferent portion. The reason was that these process all accept input. Then, the efferent portion was decided. All the report process
were included in this partition. This left processes which update the general ledger file, clean the old entry, clean the revenues and expense accounts, and the end of year process. These all belong to the transaction portion. These processes transform input data which may be used by all the report generators. So, the whole system was separated into three parts.

Working with the afferent, transform and efferent separately, the whole system was factored into modules. The first version of the design chart did not go into enough detail. Therefore, a detailed design was produced by further factoring the first design. Because I am both system designer and programmer, the bottom level modules were designed in too much detail, so some were removed for the final design.

Trying to reduce the number of parameters between modules, especially the control parameters, was considered during the factoring. The number of subordinate modules of this design was manageable. The report generator has eight subordinates which still are not too complicated to handle. Each module's algorithm can be described within one page, which is a reasonable size.
When the design chart was described, the final version design was changed once to match the definition of the PDSL. It was caused by those global variables which defined in DATA-STRUCTURE. Those elements in DATA-STRUCTURE can not be used as parameters which must not be a global variables under the definition of PDSL. The modules which would be described as library routines in PDSL were considered too. The library routines' cohesion were not reduced by changing from a normal routine to a library routine.

There were several error handlings considered during the design. As any illegal data was detected, an error message was shown and reentry requested. When the accounts were not in balance, the update general ledger file module would show the message and the update would not be executed. The module cleaning the entered account file would not be processed until entries were used to update the general ledger file.

Since this was a big design, the design charts contained many pages. It might not be understood clearly by studying only one page. The off page symbol of design chart was used to connect to another page. But when studying the design charts the off page reference is an inconvenience. Thus, the use of the off page reference table, which contained the page numbers and the names of connected modules, reduced the necessity of turning pages
and increased the design chart's readability.
CHAPTER 5
IMPLEMENTATION

Our purpose is to produce a more reliable software for the final product. Thus, each time an error is detected and successfully rectified, the system reliability will be improved. In this phase, the methods of testing and integrating modules of the system will effect the achievement of this ambition.

Before the coding, one must decide which language is to be used. Among the languages, COBOL was chosen to implement this general ledger system. There are many reasons for choosing COBOL to implement this project. The ability to handle vast amounts of data is one of the reasons for choosing COBOL. It permits data to be described precisely and in detail, thus allowing data to be accessed and manipulated at different levels. One of its benefits is that it can produce a large number of reports because it easily formats reports and edits information very quickly. Even though COBOL was created before the invention of structured programming, it still possible to use structured programming techniques with COBOL. The general ledger system is a business system which is the reason for the creation of COBOL. The above reasons prove that it is advantageous to use COBOL for coding this project.
Before implementing this system, I tried to test COBOL to make sure it was suitable to implement in this system. First, the screen driver was written to prove that the use of table and string functions of COBOL work well. Second, the file handling ability was tested. The index file was tested to check that it would suit the requirements of this particular system, which needs sequential and random files. After closing the sequential file, one cannot append to the end of the file. This problem can be solved by using a temporary file. After the above testing, the use of COBOL was decided. During the design, the use of language for implementation was not considered. Thus, when using the COBOL to code, the control flags need to be considered for the PERFORM loops' condition.

As soon as the coding is begun the testing will start and will continue throughout the phase. The individual module tests can be processed in following categories: computational test and data handling test. The computational test verifies the quantitative accuracy of the result of the operation of the software. The data handling test ensures that the input data is properly ingested, the output data is stored in the proper location and format, and data conversions have been properly performed. The bad data is to be handled properly and the data discriminately discarded.
After the module-level testing proves successful, the modules will be connected to determine whether all the required functions are correct. Beside the detail tests on an individual module, testing for the interaction between the modules is necessary. Some errors may never occur before the interconnection of the modules and will be exposed in this test. The assembly and testing of modules is done in several ways. The method used for combining software components is described below.

For combining the modules, the incremental method is used.

1. code and test one module by itself;
2. add another module;
3. test and debug the combination;
4. repeat 2 and 3 steps until the process is completed.

The advantage of the incremental approach is that it focuses on the new errors. Those errors may be caused either by a defect in the most recently added component or by the new interaction between component and the rest of the system. Before adding new components to the system, those errors must be detected and rectified. This increases the reliability of the system.
The approach chosen to integrate these components is top-down. After referencing the design specification and design structure chart, coding was begun from the top of the hierarchy and the incremental method was used to test and integrate the components down to the lower levels. During the implementation of this approach, "stubs" are required. A 'stub' simulates the functions of a component subordinate to the component being tested. After the whole system is integrated, testing will continue to assure the system's reliability.

The module testing and integration were began with the main module. Then, the afferent modules were established. Thus, the input data can be accepted and used for the following added modules. The functions to accept the input were tested entirely. Only correct data can be accepted; the system will ask for reentry of the error data. The transform modules were integrated next. These modules handle and transform the inputed data. The general ledger file will not be processed unless the input data is in balance. The entered account file can not be cleaned unless it is used to update the general ledger file. The efferent modules were integrated last. The report generators were tested one by one. If the beginning and ending date for the audit trail is not correct, the audit trail will not be produced. So, the user will be asked to reenter the dates for the incorrect dates.
During testing, three categories of values were covered. They were normal value, boundary value, and illegal value. The normal value was used by normal procedures to simulate the realistic processes, the boundary value was used to test the limits of the data range, and the illegal value which had to be detected was handled properly. All the possible conditions were tested in order to find the possible bugs.
CHAPTER 6
CONCLUSION

This project was a very good opportunity for me to utilize the knowledge which I learned from the graduate level courses. The gathered experience will help me to know more about system development methodology. Following is my own opinion about problems which I encountered in this project.

In the analysis phase, the use of the Data Flow Diagram communicates with the user more easily. Experience tells that even those people who do not have any knowledge about computers, that with the use of Data Flow Diagram, the user can more easily understand. This is very important because most users do not have much computer background. Serving as a bridge between analyst and users, the Data Flow Diagram proves to be a good tool. It can help the analyst to understand the user's requirement explicitly.

During the analysis phase, the use of PSL/PSA to describe the system by referencing Data Flow Diagram is handy. The manuals of PSL/PSA are not easy to study and understand throughly. The best way to learn the PSL/PSA is through using it. Anytime there is a question in using it, the manuals can be referenced to find out the desired object's definition. The definition of PSL/PSA objects can be matched to the Data Flow Diagram very well.
In the design phase, the structure charts help to reach a better design. The reference table of interface is used to combine with the structure chart for easy reference. Also, it helps to determine the coupling between modules, which is one of the important criteria of design. It is a good idea to have the reference table added to the structure chart. This table allows the user to understand the whole page's modules without switching to another page for reference. Because I am both system designer and programmer the design did not have to go into extreme detail. If the design is given to another programmer, it may be good to have a very detailed design down to the bottom level. The definition of PDSL was not easy to match to the design structure chart. When a DATA STRUCTURE is used, the elements of this structure must be defined as global variables. Those elements can not be used as a interface, because the interface must not be a global variable. Thus, PDSL, for structure charts, is not as good as the PSL/PSA for the Data Flow Diagram.

In this project, it took two and one-half months for analysis, one and one-half months in design and another one and one-half months in coding and writing user's manual. It proved that putting more effort in the beginning of the system development will reduce the effort for the following phases. The most important benefit of putting more effort in the analysis is that the developed system will follow
the user's requirement better.

The software engineering methodology gives us the guidelines, which is the result of people's experience in developing large scale software system. Those guidelines lead to less errors for the system development. The major purpose of this project is to learn the use of those guidelines. Changing those guidelines to be one's own experience is the most important aspect.


UNIVERSITY OF MONTANA PSA/PSL

Name Selection

Parameters:  DB=FINAL.DBF PRINT PUNCH=PSANAM.TMP
SELECTION='TRACE-KEY="level-c"' ORDER=BYTYPE

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>account-entries</td>
<td>INPUT</td>
</tr>
<tr>
<td>2</td>
<td>request-account-report</td>
<td>INPUT</td>
</tr>
<tr>
<td>3</td>
<td>request-financial-report</td>
<td>INPUT</td>
</tr>
<tr>
<td>4</td>
<td>update-request</td>
<td>INPUT</td>
</tr>
<tr>
<td>5</td>
<td>book-keeper</td>
<td>INTERFACE</td>
</tr>
<tr>
<td>6</td>
<td>chief-accountant</td>
<td>INTERFACE</td>
</tr>
<tr>
<td>7</td>
<td>account-report</td>
<td>OUTPUT</td>
</tr>
<tr>
<td>8</td>
<td>balance-error</td>
<td>OUTPUT</td>
</tr>
<tr>
<td>9</td>
<td>financial-report</td>
<td>OUTPUT</td>
</tr>
<tr>
<td>10</td>
<td>general-ledger-processing</td>
<td>PROCESS</td>
</tr>
</tbody>
</table>
1 DEFINE INPUT account-entries;
2     /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
3     SYNONYMS ARE: acc-es;
4     DESCRIPTION;
5     It contains the account-entry and adjust-account-entry
6     which are collected from other accounting systems or
7     the adjustment;
8     TRACE-KEY IS: 'level-c';
9     GENERATED: BY book-keeper;
10     RECEIVED: BY general-ledger-processing;
11     SUBPARTS ARE: account-information,
12     account-entry,
13     adjust-account-entry;

15 DEFINE INPUT request-account-report;
16     /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
17     SYNONYMS ARE: r-a-r;
18     DESCRIPTION;
19     This indicates the account reports requested.;
20     TRACE-KEY IS: 'level-c',
21     'level-0',
22     'level-3';
23     GENERATED: BY book-keeper;
24     RECEIVED: BY general-ledger-processing,
25     BY report-generator,
26     BY print-chart-of-account,
27     BY print-list-of-account-entries,
28     BY print-audit-trail;
29     CONSISTS OF:
30     request-audit-trail,
31     request-chart-of-acc,
32     request-list-of-acc-entries;
34
35 DEFINE INPUT request-financial-report;
36     /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
37     SYNONYMS ARE: r-f-r;
38     DESCRIPTION;
39     This data indicates the requests on
40     the financial reports.;
41     TRACE-KEY IS: 'level-c',
42     'level-0',
43     'level-3',
UNIVERSITY OF MONTANA PSA/PSL

Formatted Problem Statement

'level-3.2';
GENERATED: BY chief-accountant;
RECEIVED: BY general-ledger-processing,
BY report-generator,
BY print-financial-report,
BY print-ledger-sheet,
BY produce-trial-balance,
BY print-balance-sheet,
BY print-income-statement;
CONSISTS OF:
request-balance-sheet,
request-ledger-sheet,
request-income-statement,
request-trial-balance;

DEFINE INPUT update-request;

DESCRIPTION;
This request indicates the decision to make an update
to the general-ledger-file, close accounts at the end
of the year and clean the revenue and expense accounts
at end of period.;
TRACE-KEY IS: 'level-0',
'level-c',
'level-2';
GENERATED: BY book-keeper;
RECEIVED: BY general-ledger-processing,
BY account-update;
CONSISTS OF:
request-general-ledger-update,
request-end-of-year-close,
request-revenues-expense-clean;

DEFINE INTERFACE book-keeper;
DESCRIPTION;
The book-keeper collects accounts from payroll,
account receivable and account payable. He
inputs the accounts and adjusts the accounts.
He receives the ledger-sheet and
list-of-account-entries;
TRACE-KEY IS: 'level-c';
GENERATES:
update-request,
account-entries,
request-account-report,
account-entry,
account-information,
adjust-account-entry;
RECEIVES:
  balance-error,
  account-report,
  audit-trail,
  chart-of-account;

DEFINE INTERFACE chief-accountant;

SYNONYMS ARE: c-ac;

DESCRIPTION;
He decides the kind of financial-reports
107 to be made. He also checks the Trial-Balance
108 for possible adjustments.;
TRACE-KEY IS: 'level-c';

GENERATES:
  request-financial-report;

RECEIVES:
  financial-report,
  balance-sheet;

DEFINE OUTPUT account-report;

SYNONYMS ARE: acc-r;

DESCRIPTION;
The reports are for reference. It will be produced
121 according to the book keeper's request.;
TRACE-KEY IS: 'level-c',
'level-0';

GENERATED:    BY general-ledger-processing,
125    BY report-generator;
126 RECEIVED:    BY book-keeper;
SUBPARTS ARE:  chart-of-account,
129    list-of-account-entries,
    audit-trail;
DERIVED BY:    report-generator
USING:          general-ledger-file,
132    entered-account-file,
    previous-entered-account-file;

DEFINE OUTPUT balance-error;

SYNONYMS ARE: b-err;

DESCRIPTION;
The output indicates the error message about
the imbalance of the accounts.;
TRACE-KEY IS: 'level-c',
'level-0',
'level-2';
GENERATED:    BY general-ledger-processing,
145    BY account-update,
    BY update-general-ledger;
DEFINE OUTPUT financial-report;

DESCRIPTION;

These reports show the financial situation of a business.

TRACE-KEY IS: 'level-c',
'lvel-3';

GENERATED: BY general-ledger-processing,
BY report-generator,
BY print-financial-report;

RECEIVED: BY chief-accountant;

SUBPARTS ARE: balance-sheet,
income-statement,
trial-balance,
ledger-sheet;

DERIVED BY: report-generator
USING: general-ledger-file;

DEFINE PROCESS
general-ledger-processing;

DESCRIPTION;

This process represents the complete
general-ledger-system.

TRACE-KEY IS: 'level-c';

GENERATES:
account-report,
financial-report,
balance-error;

RECEIVES:
update-request,
account-entries,
request-account-report,
request-financial-report;

SUBPARTS ARE: account-maintain,
account-update,
report-generator;

EOF EOF EOF EOF EOF
Name Selection

Parameters: DB=FINAL.DBF PRINT PUNCH=PSANAM.TMP
SELECTION='TRACE-KEY="level-0"' ORDER=BYTYPE

1  entered-account  ENTITY
2  general-ledger-account  ENTITY
3  previous-entered-account  ENTITY
4  account-entry  INPUT
5  account-information  INPUT
6  adjust-account-entry  INPUT
7  request-account-report  INPUT
8  request-financial-report  INPUT
9  update-request  INPUT
10 account-report  OUTPUT
11 balance-error  OUTPUT
12 financial-report  OUTPUT
13 account-maintain  PROCESS
14 account-update  PROCESS
15 report-generator  PROCESS
16 entered-account-file  SET
17 general-ledger-file  SET
18 previous-entered-account-file  SET
1 DEFINE ENTITY entered-account;
2 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
3 SYNONYMS ARE: e-acc;
4 DESCRIPTION;
5 This is the entity of an entered-account-file
6 which contains the whole entered accounts and
7 adjusted accounts. It keeps the whole trail
8 for a period of the business;
9 TRACE-KEY IS: 'level-0',
10 'level-1',
11 'level-2',
12 'level-3',
13 'level-3.2';
14 COLLECTED IN: entered-account-file;
15 CONSISTS OF:
16   journal-entry-number,
17   source-code,
18   account-number,
19   date-of-entry,
20   account-name,
21   value-of-account,
22   update-code;
23 ADDED TO:
24   entered-account-file
25   BY account-maintain;
26 ADDED TO:
27   entered-account-file
28   BY enter-account-entries;
29 ADDED TO:
30   entered-account-file
31   BY adjust-account-entries;
32 REFERENCED IN:
33   entered-account-file
34   BY adjust-account-entries;
35 REMOVED FROM:
36   entered-account-file
37   BY account-update;
38 REMOVED FROM:
39   entered-account-file
40   BY clean-old-entries;
41
42 DEFINE ENTITY
general-ledger-account;
43 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: g-l-acc;
DESCRIPTION:
This is an entity of general-ledger-file.
It keeps the all information about general ledger account.;
TRACE-KEY IS: 'level-0',
    'level-1',
    'level-2',
    'level-3',
    'level-3.2';
COLLECTED IN: general-ledger-file;
CONSISTS OF:
    account-number,
    account-name,
    account-type,
    current-total,
    year-to-now-total,
    previous-year-total;
IDENTIFIED BY: account-number;
ADDED TO:
    general-ledger-file
    BY account-maintain;
ADDED TO:
    general-ledger-file
    BY maintain-account-information;
MODIFIED IN:
    general-ledger-file
    BY account-maintain;
MODIFIED IN:
    general-ledger-file
    BY maintain-account-information;
REFERENCED IN:
    general-ledger-file
    BY account-maintain;
REFERENCED IN:
    general-ledger-file
    BY enter-account-entries;
REMOVED FROM:
    general-ledger-file
    BY account-maintain;
REMOVED FROM:
    general-ledger-file
    BY maintain-account-information;

DEFINE ENTITY
previous-entered-account;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: p-e-a;
DESCRIPTION:
This is an entity of previous-entered-account-file
which keeps the trail for a fiscal year of business.
It can be used to produce the audit-trail report.;
TRACE-KEY IS: 'level-0',
'level-2',
'level-3';
COLLECTED IN: previous-entered-account-file;
CONSISTS OF:
  journal-entry-number,
  account-number,
  date-of-entry,
  value-of-account;
ADDED TO:
  previous-entered-account-file
  BY account-update;
ADDED TO:
  previous-entered-account-file
  BY clean-old-entries;
DEFINE INPUT account-entry;
SYNONYMS ARE: acc-e;
DESCRIPTION;
This data are collected from other accounting system
which is to be the input to the system.;
TRACE-KEY IS: 'level-0',
'level-1';
GENERATED: BY book-keeper;
RECEIVED: BY account-maintain,
  BY enter-account-entries;
PART OF: account-entries;
CONSISTS OF:
  account-number,
  account-name,
  source-code,
  date-of-entry,
  value-of-account;
DEFINE INPUT account-information;
SYNONYMS ARE: a-i;
DESCRIPTION;
This data shows the added, deleted and modified
information on the general ledger account.;
TRACE-KEY IS: 'level-0',
'level-1';
GENERATED: BY book-keeper;
RECEIVED: BY account-maintain,
  BY maintain-account-information;
PART OF: account-entries;
CONSISTS OF:
  account-number,
  account-name,
account-type,  
year-to-now-total,  
current-total,  
previous-year-total,  
information-indicator;

USED BY:
account-maintain
TO MAINTAIN general-ledger-file;

USED BY:
maintain-account-information
TO MAINTAIN general-ledger-file;

DEFINE INPUT
adjust-account-entry;

DESCRIPTION;
This data makes adjustments on accounts.;

TRACE-KEY IS: 'level-0',
'level-1';

GENERATED: BY book-keeper;
RECEIVED: BY account-maintain,
BY adjust-account-entries;

PART OF: account-entries;

CONSISTS OF:
value-of-account,
source-code,
account-name,
account-number,
date-of-entry;

DEFINE INPUT
request-account-report;

DESCRIPTION;
This indicates the account reports requested.;

TRACE-KEY IS: 'level-c',
'level-0',
'level-3';

GENERATED: BY book-keeper;
RECEIVED: BY general-ledger-processing,
BY report-generator,
BY print-chart-of-account,
BY print-list-of-account-entries,
BY print-audit-trail;

CONSISTS OF:
request-audit-trail,
request-chart-of-acc,
request-list-of-acc-entries;
UNIVERSITY OF MONTANA PSA/PSL
Formatted Problem Statement

198 DEFINE INPUT
199 request-financial-report;
200 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
201 SYNONYMS ARE: r-f-r;
202 DESCRIPTION;
203 This data indicates the requests on
204 the financial reports.;
205 TRACE-KEY IS: 'level-c',
206 'level-0',
207 'level-3',
208 'level-3.2';
209 GENERATED: BY chief-accountant;
210 RECEIVED: BY general-ledger-processing,
211 BY report-generator,
212 BY print-financial-report,
213 BY print-ledger-sheet,
214 BY produce-trial-balance,
215 BY print-balance-sheet,
216 BY print-income-statement;
217 CONSISTS OF:
218 request-balance-sheet,
219 request-ledger-sheet,
220 request-income-statement,
221 request-trial-balance;
222
223 DEFINE INPUT
224 update-request;
225 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
226 SYNONYMS ARE: u-r;
227 DESCRIPTION;
228 This request indicates the decision to make an update
229 to the general-ledger-file, close accounts at the end
230 of the year and clean the revenue and expense accounts
231 at end of period.;
232 TRACE-KEY IS: 'level-0',
233 'level-c',
234 'level-2';
235 GENERATED: BY book-keeper;
236 RECEIVED: BY general-ledger-processing,
237 BY account-update;
238 CONSISTS OF:
239 request-general-ledger-update,
240 request-end-of-year-close,
241 request-revenues-expense-clean;
242
243 DEFINE OUTPUT
244 account-report;
245 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
246 SYNONYMS ARE: acc-r;
247 DESCRIPTION;
248 The reports are for reference. It will be produced
249 according to the book keeper's request.;
250 TRACE-KEY IS: 'level-c',
249 DEFINE OUTPUT balance-error;
250 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
251 SYNONYMS ARE: b-err;
252 DESCRIPTION;
253 The output indicates the error message about
254 the inbalance of the accounts.;
255 TRACE-KEY IS: 'level-c',
256 'level-0',
257 'level-2';
258 GENERATED: BY general-ledger-processing,
259 BY account-update,
260 BY update-general-ledger;
261 RECEIVED: BY book-keeper;
262
263 DEFINE OUTPUT financial-report;
264 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
265 SYNONYMS ARE: f-r;
266 DESCRIPTION;
267 These reports show the financial situation of a business.;
268 TRACE-KEY IS: 'level-c',
269 'level-0',
270 'level-3';
271 GENERATED: BY general-ledger-processing,
272 BY report-generator,
273 BY print-financial-report;
274 RECEIVED: BY chief-accountant;
275 SUBPARTS ARE: balance-sheet,
276 income-statement,
277 trial-balance,
278 ledger-sheet;
279 DERIVED BY: report-generator
280 USING: general-ledger-file;
281
282 DEFINE PROCESS account-maintain;
283 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
284 SYNONYMS ARE: a-m;
285 DESCRIPTION;
286 This process handles those things that relate
287 to the account information, account entry and
TRACE-KEY IS: 'level-0';

RECEIVES:
account-information,
adjust-account-entry,

SUBPARTS ARE: maintain-account-information,
enter-account-entries,
adjust-account-entries;

PART OF: general-ledger-processing;

ADDS: general-ledger-account TO
general-ledger-account;

ADDS: entered-account TO entered-account-file;

MAINTAINS: general-ledger-account
USING: account-information;

MODIFIES: general-ledger-account IN
general-ledger-file;

REFERENCES: general-ledger-account IN
general-ledger-file;

REMOVES: general-ledger-account FROM

general-ledger-file;

DEFINE PROCESS account-update;

SYNONYMS ARE: a-u;

DESCRIPTION;

This process uses the entered-account-file
to update the general-ledger-file at the end
of the period. It cleans the entered-account-file
and also appends to the previous-entered-account-
file. It also cleans the revenues and expense
accounts at the end of a period. At the end of
a year, it will close the accounts.;

TRACE-KEY IS: 'level-0';

GENERATES:
balance-error;

RECEIVES:
update-request;

SUBPARTS ARE: update-general-ledger,
clean-old-entries,
end-of-year-update,
clean-revenues-expense;

PART OF: general-ledger-processing;

ADDS: previous-entered-account TO
previous-entered-account-file;

REMOVES: entered-account FROM
entered-account-file;

UPDATES: general-ledger-file
USING: entered-account-file;

DEFINE PROCESS report-generator;
UNIVERSITY OF MONTANA PSA/PSL
Formatted Problem Statement

SYNONYMS ARE: r-g;
DESCRIPTION;
This process generates the financial-report and account-report, when requested.;
TRACE-KEY IS: 'level-0';

GENERATES:
  financial-report,
  account-report,
  audit-trail,
  chart-of-account,
  list-of-account-entries;
RECEIVES:
  request-financial-report,
  request-account-report;
SUBPARTS ARE: print-chart-of-account,
  print-financial-report,
  print-list-of-account-entries,
  print-audit-trail;
PART OF: general-ledger-processing;
DERIVES: financial-report
  USING: general-ledger-file;
DERIVES: account-report
  USING: general-ledger-file,
  entered-account-file,
  previous-entered-account-file;

DEFINE SET
entered-account-file;
SYNONYMS ARE: e-t-f;
DESCRIPTION;
This file contains the whole entered accounts and adjusted accounts. It keeps the whole trail for a period of the business. It will be used to update the general-ledger-file and append to the previous-entered-account-file.;
TRACE-KEY IS: 'level-0',
  'level-1',
  'level-2',
  'level-3',
  'level-3.2';
COLLECTION OF:
  entered-account;
HAS: entered-account
  ADDED BY account-maintain;
HAS: entered-account
  ADDED BY enter-account-entries;
HAS: entered-account
  ADDED BY adjust-account-entries;
USED BY:
DEFINE SET general-ledger-file;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: g-l-f;
DESCRIPTION;
This file contains ledger, total, title and heading accounts. It keeps the all information about general ledger account. So, it always is in balance. It is used to produced the financial reports.;
TRACE-KEY IS: 'level-0',
'level-1',
'level-2',
'level-3',
'level-3.2';
COLLECTION OF:
general-ledger-account;
ORDERED BY: account-number;
HAS: general-ledger-account
ADDED BY account-maintain;
HAS: general-ledger-account
ADDED BY maintain-account-information;
USED BY:
report-generator
TO DERIVE financial-report;
USED BY:
report-generator
TO DERIVE account-report;
FORMATTED PROBLEM STATEMENT

453 
454 USED BY: 
455 print-chart-of-account 
456 TO DERIVE chart-of-account; 
457 
458 USED BY: 
459 print-ledger-sheet 
460 TO DERIVE ledger-sheet; 
461 
462 USED BY: 
463 produce-trial-balance 
464 TO DERIVE trial-balance; 
465 
466 USED BY: 
467 print-balance-sheet 
468 TO DERIVE balance-sheet; 
469 
470 MAINTAINED BY: account-maintain 
471 USING : account-information; 
472 MAINTAINED BY: maintain-account-information 
473 USING : account-information; 
474 HAS: general-ledger-account 
475 MODIFIED BY account-maintain; 
476 HAS: general-ledger-account 
477 MODIFIED BY maintain-account-information; 
478 HAS: general-ledger-account 
479 REFERENCED BY account-maintain; 
480 HAS: general-ledger-account 
481 REFERENCED BY enter-account-entries; 
482 HAS: general-ledger-account 
483 REMOVED BY account-maintain; 
484 UPDATED BY: account-update 
485 USING: entered-account-file; 
486 UPDATED BY: update-general-ledger 
487 USING: entered-account-file; 
488 UPDATED BY: clean-revenues-expense; 
489 UPDATED BY: end-of-year-update; 
490 
491 DEFINE SET 
492 previous-entered-account-file; 
493 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */ 
494 SYNONYMS ARE: p-e-a-f; 
495 DESCRIPTION; 
496 This file keeps the trail for a fiscal year 
497 of business. It can be used to produce the 
498 audit-trail report.; 
499 TRACE-KEY IS: 'level-0', 
500 'level-2', 
501 'level-3'; 
502 COLLECTION OF: 
503 previous-entered-account;
504 HAS: previous-entered-account
505 ADDED BY account-update;
506 HAS: previous-entered-account
507 ADDED BY clean-old-entries;
508 USED BY:
509 report-generator
510 TO DERIVE account-report;
511 USED BY:
512 print-audit-trail
513 TO DERIVE audit-trail;
514
515 EOF EOF EOF EOF EOF
Name Selection

Parameters:  DB=FINAL.DBF  PRINT  PUNCH=PSANAM.TMP
            SELECTION='TRACE-KEY="level-l"'  ORDER=BYTYPE

1  account-name  ELEMENT
2  account-type  ELEMENT
3  category-number  ELEMENT
4  current-total  ELEMENT
5  day  ELEMENT
6  information-indicater  ELEMENT
7  journal-entry-number  ELEMENT
8  month  ELEMENT
9  previous-year-total  ELEMENT
10  source-code  ELEMENT
11  subcategory-number  ELEMENT
12  update-code  ELEMENT
13  value-of-account  ELEMENT
14  year-to-now-total  ELEMENT
15  entered-account  ENTITY
16  general-ledger-account  ENTITY
17  account-number  GROUP
18  date-of-entry  GROUP
19  account-entry  INPUT
20  account-information  INPUT
21  adjust-account-entry  INPUT
22  adjust-account-entries  PROCESS
23  enter-account-entries  PROCESS
24  maintain-account-information  PROCESS
25  entered-account-file  SET
26  general-ledger-file  SET
DEFINE ELEMENT account-name;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: acc-n;
DESCRIPTION;
It is a description of each of the
general-ledger-account.
It will not more than 30 characters.;
TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';
CONTAINED IN: entered-account,
general-ledger-account,
account-entry,
account-information,
adjust-account-entry;

DEFINE ELEMENT account-type;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: acc-ty;
DESCRIPTION;
This element distinguishes the types of
general-ledger-accounts. The four different
types of accounts are 0)regular 1)title
2)total 3)heading.;
TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';
CONTAINED IN: general-ledger-account,
account-information;
VALUES ARE:
0 THRU 3;

DEFINE ELEMENT category-number;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: ca-nb;
DESCRIPTION;
The category number identifies groups,
1)assets 2)liabilities 3)income 4) expense,
of a general-ledger-file.;
TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
DEFINE ELEMENT current-total;
SYNONYMS ARE: cu-to;
DESCRIPTION;
It indicates the current amount of account.;
TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';
CONTAINED IN: general-ledger-account,
account-information;
VALUES ARE:
-100000000.000000
THRU 100000000.000000;

DEFINE ELEMENT day;
SYNONYMS ARE: d;
DESCRIPTION;
It indicates the day in a month.;
TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';
CONTAINED IN: date-of-entry;
VALUES ARE:
1 THRU 31;

DEFINE ELEMENT information-indicater;
SYNONYMS ARE: in-id;
DESCRIPTION;
It indicates the types of information.
1) add account 2) delete account 3) modify account.;
TRACE-KEY IS: 'level-1';
CONTAINED IN: account-information;
VALUES ARE:
1 THRU 3;

DEFINE ELEMENT journal-entry-number;
SYNONYMS ARE: j-e-n;
DESCRIPTION;
It keeps a sequence number to identify the
96 account entered for a fiscal year.;
97 TRACE-KEY IS: 'level-3',
98 'level-3.2',
99 'level-2',
100 'level-1';
101 CONTAINED IN: entered-account,
102 previous-entered-account;
103 VALUES ARE:
104 1 THRU 999999;
105
106 DEFINE ELEMENT month;
107 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
108 SYNONYMS ARE: m;
109 DESCRIPTION;
110 It indicates the month in a year.;
111 TRACE-KEY IS: 'level-3',
112 'level-3.2',
113 'level-2',
114 'level-1';
115 CONTAINED IN: date-of-entry;
116 VALUES ARE:
117 1 THRU 12;
118
119 DEFINE ELEMENT previous-year-total;
120 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
121 SYNONYMS ARE: p-y-to;
122 DESCRIPTION;
123 It keeps the previous year account amount.;
124 TRACE-KEY IS: 'level-3',
125 'level-3.2',
126 'level-2',
127 'level-1';
128 CONTAINED IN: general-ledger-account,
129 account-information;
130 VALUES ARE:
131 -100000000.000000 THRU 100000000.000000;
132
133 DEFINE ELEMENT source-code;
134 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
135 SYNONYMS ARE: s-c;
136 DESCRIPTION;
137 It indicates which system the accounts come from:
138 (1) the account receivable system
139 (2) the account payable system
140 (3) the payroll system
141 (4) the adjustment entry.;
142 TRACE-KEY IS: 'level-1',
143 'level-2',
144 'level-3',
145 'level-3.2';
147 CONTAINED IN: entered-account, account-entry, adjust-account-entry;
150 VALUES ARE:
151 1 THRU 4;
152
153 DEFINE ELEMENT subcategory-number;
154 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
155 SYNONYMS ARE: su-nb;
156 DESCRIPTION;
157 It is an extension to the category number
158 which is a part of account number.;
159 TRACE-KEY IS: 'level-3',
160 'level-3.2',
161 'level-2',
162 'level-1';
163 CONTAINED IN: account-number;
164 VALUES ARE:
165 0 THRU 99999;
166
167 DEFINE ELEMENT update-code;
168 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
169 SYNONYMS ARE: up-c;
170 DESCRIPTION;
171 It indicates the entered-account-record is used
172 to update the general-ledger-file or not.
173 1) when the entered-account-record is not used to
174 update the general-ledger-file yet.
175 2) when the entered-account-record
176 was used to update the general-ledger-file.;
177 TRACE-KEY IS: 'level-3',
178 'level-3.2',
179 'level-2',
180 'level-1';
181 CONTAINED IN: entered-account;
182
183 DEFINE ELEMENT value-of-account;
184 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
185 SYNONYMS ARE: v-o-acc;
186 DESCRIPTION;
187 This element represents the amount in each account.;
188 TRACE-KEY IS: 'level-3',
189 'level-3.2',
190 'level-2',
191 'level-1';
192 CONTAINED IN: entered-account,
193 previous-entered-account,
194 account-entry,
195 adjust-account-entry;
196 VALUES ARE:
197 -100000000.000000
UNIVERSITY OF MONTANA PSA/PSL
Formatted Problem Statement

198 THRU 100000000.000000;

200 DEFINE ELEMENT year-to-now-total;
201 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
202 SYNONYMS ARE: y-t-n-t;
203 DESCRIPTION;
204 This element is part of the general-ledger account.
205 It is used to put the year to the now value for
206 the revenues and the expense accounts.
207 TRACE-KEY IS: 'level-3',
208 'level-3.2',
209 'level-2',
210 'level-1';
211 CONTAINED IN: general-ledger-account,
212 account-information;
213 VALUES ARE:
214 -100000000.000000
215 THRU 100000000.000000;

217 DEFINE ENTITY entered-account;
218 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
219 SYNONYMS ARE: e-acc;
220 DESCRIPTION;
221 This is the entity of an entered-account-file
222 which contains the whole entered accounts and
223 adjusted accounts. It keeps the whole trail
224 for a period of the business.
225 TRACE-KEY IS: 'level-0',
226 'level-1',
227 'level-2',
228 'level-3',
229 'level-3.2';
230 COLLECTED IN: entered-account-file;
231 CONSISTS OF:
232 journal-entry-number,
233 source-code,
234 account-number,
235 date-of-entry,
236 account-name,
237 value-of-account,
238 update-code;
239 ADDED TO:
240 entered-account-file
241 BY account-maintain;
242 ADDED TO:
243 entered-account-file
244 BY enter-account-entries;
245 ADDED TO:
246 entered-account-file
247 BY adjust-account-entries;
248 REFERENCED IN:
UNIVERSITY OF MONTANA PSA/PSL
Formatted Problem Statement

249 entered-account-file
BY adjust-account-entries;

251 REMOVED FROM:
252 entered-account-file
BY account-update;

254 REMOVED FROM:
255 entered-account-file
BY clean-old-entries;

257

258 DEFINE ENTITY
259 general-ledger-account;
260 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
261 SYNONYMS ARE: g-l-acc;
262 DESCRIPTION;
This is an entity of general-ledger-file.
It keeps the all information about general ledger account.;
264
265 TRACE-KEY IS: 'level-0',
266 'level-1',
268 'level-2',
269 'level-3',
270 'level-3.2';
271 COLLECTED IN: general-ledger-file;
272 CONSISTS OF:
account-number,
account-name,
account-type,
current-total,
year-to-now-total,
previous-year-total;
279 IDENTIFIED BY: account-number;
280 ADDED TO:
281 general-ledger-file
282 BY account-maintain;
283 ADDED TO:
284 general-ledger-file
285 BY maintain-account-information;
286 MODIFIED IN:
287 general-ledger-file
288 BY account-maintain;
289 MODIFIED IN:
290 general-ledger-file
291 BY maintain-account-information;
292 REFERENCED IN:
293 general-ledger-file
294 BY account-maintain;
295 REFERENCED IN:
296 general-ledger-file
297 BY enter-account-entries;
298 REMOVED FROM:
299 general-ledger-file
DEFINE GROUP account-number;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: acc-nb;

DESCRIPTION;
It identifies a particular account in the
general-ledger-file. It contains the category
and sub-category numbers.

TRACE KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';

CONSISTS OF:
category-number,
subcategory-number;

CONTAINED IN: entered-account,
general-ledger-account,
previous-entered-account,
account-entry,
account-information,
adjust-account-entry;

IDENTIFIES: general-ledger-account;
ORDERS: general-ledger-file;

DEFINE GROUP date-of-entry;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: d-of-e;

DESCRIPTION;
This element represents the date of making an entry.

TRACE KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';

CONSISTS OF:
day,
month;

CONTAINED IN: entered-account,
previous-entered-account,
account-entry,
adjust-account-entry;

DEFINE INPUT account-entry;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: acc-e;

DESCRIPTION;
This data are collected from other accounting system
which is to be the input to the system;
TRACE-KEY IS: 'level-0',
'level-1';
GENERATED: BY book-keeper;
RECEIVED: BY account-maintain,
BY enter-account-entries;
PART OF: account-entries;
CONSISTS OF:
  account-number,
  account-name,
  source-code,
  date-of-entry,
  value-of-account;

DEFINE INPUT
account-information;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: a-i;
DESCRIPTION;
This data shows the added, deleted and modified
information on the general ledger account.;
TRACE-KEY IS: 'level-0',
'level-1';
GENERATED: BY book-keeper;
RECEIVED: BY account-maintain,
BY maintain-account-information;
PART OF: account-entries;
CONSISTS OF:
  account-number,
  account-name,
  account-type,
  year-to-now-total,
  current-total,
  previous-year-total,
  information-indicater;
USED BY:
  account-maintain
TO MAINTAIN general-ledger-file;
USED BY:
  maintain-account-information
TO MAINTAIN general-ledger-file;

DEFINE INPUT
adjust-account-entry;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: ad-acc-e;
DESCRIPTION;
This data makes adjustments on accounts.;
TRACE-KEY IS: 'level-0',
'level-1';
GENERATED: BY book-keeper;
RECEIVED: BY account-maintain,
BY adjust-account-entries;
402 PART OF: account-entries;
403 CONSISTS OF:
404 value-of-account,
405 source-code,
406 account-name,
407 account-number,
408 date-of-entry;
409
410 DEFINE PROCESS
411 adjust-account-entries;
412 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
413 SYNONYMS ARE: a-a-e;
414 DESCRIPTION;
415 This process receives the adjust-account-entries
416 to update or add to the entered-account-file.;
417 TRACE-KEY IS: 'level-1';
418 RECEIVES:
419 adjust-account-entry;
420 PART OF: account-maintain;
421 ADDS: entered-account TO entered-account-file;
422 REFERENCES: entered-account IN entered-account-file;
423 PROCEDURE;
424 ****************************
425 TAKE the next journal-entry-number value.
426 REPEAT the following:
427 take the NEXT adjust-account-entry.
428 IF adjust-account-entry is valid,
429 THEN,
430 MOVE the corresponding fields from
431 adjust-account-entry to entered-account.
432 MOVE the journal-entry-number to
433 entered-account.
434 WRITE the entered-account to the end of
435 the entered-account-file.
436 INCREASE the journal-entry-number by 1.
437 OTHERWISE,
438 display 'input invalid' message.
439 UNTIL there are no more adjust-account-entry.
440 ****************************
441 ;
442
443 DEFINE PROCESS
444 enter-account-entries;
445 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
446 SYNONYMS ARE: e-a-e;
447 DESCRIPTION;
448 This process receives the account-entries and
449 creates the entered-account-file by using the
450 valid account-entry.;
451 TRACE-KEY IS: 'level-1';
452 RECEIVES:
UNIVERSITY OF MONTANA PSA/PSL
Formatted Problem Statement

account-entry;

PART OF: account-maintain;

ADDS: entered-account TO entered-account-file;

REFERENCES: general-ledger-account IN
general-ledger-file;

PROCEDURE;

********************************************************************************

TAKE the next journal-entry-number value.

REPEAT the following:

TAKE the next account-entry.

IF account-entry is valid,

THEN,

SEARCH general-ledger-file for matching
account-number and account-type='regular',

IF found,

THEN,

MOVE corresponding fields from
account-entry to entered-account.

MOVE the journal-entry-number to
entered-account.

WRITE account-entry to end of
entered-account.

INCREASE the journal-entry-number by 1.

OTHERWISE,

DISPLAY 'acc-no. not exist' or 'acc-type
is not regular'.

OTHERWISE,

DISPLAY 'acc-entry invalid'.

UNTIL there are no more acc-entry.

********************************************************************************

DEFINE PROCESS
maintain-account-information;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */

SYNONYMS ARE: m-a-i;

DESCRIPTION;

This process maintains the general-ledger-file.

It adds, deletes and modifies the account in the
general-ledger-file;

TRACE-KEY IS: 'level-1';

RECEIVES:

account-information;

PART OF: account-maintain;

ADDS: general-ledger-account TO
general-ledger-file;

MAINTAINS: general-ledger-file

USING : account-information;

MODIFIES: general-ledger-account IN
general-ledger-file;

REMOVES: general-ledger-account FROM
504  general-ledger-file;
505  PROCEDURE;
506
507  REPEAT the following:
508  TAKE the next account-information.
509  IF account-information is valid,
510  THEN,
511     SELECT the case which applies:
512     CASE1: (information-indicator is 'add')
513        SEARCH general-ledger-file for matching
514        account-number.
515        IF found,
516           DISPLAY 'account exist already'.
517        OTHERWISE,
518           MOVE corresponding fields from
519           account-information
520           to general-ledger-account.
521           ADD general-ledger-account to
522           general-ledger-file.
523     CASE2: (information-indicator is 'delete')
524        SEARCH general-ledger-file for matching
525        account-number.
526        IF found,
527           THEN,
528           IF amount in general-ledger-account
529           of that account-number equal zero,
530           DELETE the general-ledger-account
531           from general-ledger-file.
532           OTHERWISE,
533           DISPLAY 'the account value is not
534           zero, can't be delete '.
535        OTHERWISE,
536           DISPLAY 'account not found'.
537     CASE3: (information-indicator is 'modify')
538        SEARCH general-ledger-file for matching
539        account-number.
540        IF found,
541           THEN,
542           UPDATE the corresponding field of
543           general-ledger-account USING
544           account-information.
545        OTHERWISE,
546           DISPLAY 'account not found'.
547     OTHERWISE,
548           DISPLAY 'account-information invalid'.
549  UNTIL there are no more account-information.
550
551  DEFINE SET
552  entered-account-file;
UNIVERSITY OF MONTANA PSA/PSL
Formatted Problem Statement

SYNONYMS ARE: e-t-f;
DESCRIPTION;
This file contains the whole entered accounts and adjusted accounts. It keeps the whole trail for a period of the business. It will be used to update the general-ledger-file and append to the previous-entered-account-file.;
TRACE-KEY IS: 'level-0',
'level-1',
'level-2',
'level-3',
'level-3.2';

COLLECTION OF:
entered-account;
HAS: entered-account
ADDED BY account-maintain;
entered-account
ADDED BY enter-account-entries;
entered-account
ADDED BY adjust-account-entries;

USED BY:
report-generator
TO DERIVE account-report;
print-list-of-account-entries
TO DERIVE list-of-account-entries;
print-ledger-sheet
TO DERIVE ledger-sheet;
produce-trial-balance
TO DERIVE trial-balance;
entered-account
REFERENCED BY adjust-account-entries;
entered-account
REMOVED BY account-update;
entered-account
REMOVED BY clean-old-entries;
account-update
TO UPDATE general-ledger-file;
update-general-ledger
TO UPDATE general-ledger-file;

EMPLOYED BY: clean-old-entries;

DEFINE SET general-ledger-file;
SYNONYMS ARE: g-l-f;
DESCRIPTION;
606 This file contains ledger, total, title and heading
607 accounts. It keeps the all information about
608 general ledger account. So, it always is in balance.
609 It is used to produced the financial reports.

610 TRACE-KEY IS: 'level-0',
611 'level-1',
612 'level-2',
613 'level-3',
614 'level-3.2';

615 COLLECTION OF:
616  general-ledger-account;
617 ORDERED BY: account-number;
618 HAS: general-ledger-account
619  ADDED BY account-maintain;
620 HAS: general-ledger-account
621  ADDED BY maintain-account-information;
622 USED BY:
623  report-generator
624  TO DERIVE financial-report;
625 USED BY:
626  report-generator
627  TO DERIVE account-report;
628 USED BY:
629  print-chart-of-account
630  TO DERIVE chart-of-account;
631 USED BY:
632  print-ledger-sheet
633  TO DERIVE ledger-sheet;
634 USED BY:
635  produce-trial-balance
636  TO DERIVE trial-balance;
637 USED BY:
638  print-balance-sheet
639  TO DERIVE balance-sheet;
640 USED BY:
641  print-income-statement
642  TO DERIVE income-statement;
643 MAINTAINED BY: account-maintain
644 USING: account-information;
645 MAINTAINED BY: maintain-account-information
646 USING: account-information;
647 HAS: general-ledger-account
648 MODIFIED BY account-maintain;
649 HAS: general-ledger-account
650 MODIFIED BY maintain-account-information;
651 HAS: general-ledger-account
652 REFERENCED BY account-maintain;
653 HAS: general-ledger-account
654 REFERENCED BY enter-account-entries;
655 HAS: general-ledger-account
656 REMOVED BY account-maintain;
HAS: general-ledger-account
REMOVED BY maintain-account-information;
UPDATED BY: account-update
USING: entered-account-file;
UPDATED BY: update-general-ledger
USING: entered-account-file;
UPDATED BY: clean-revenues-expense;
UPDATED BY: end-of-year-update;

EOF EOF EOF EOF EOF
Name Selection

Parameters: DB=FINAL.DBF PRINT PUNCH=PSANAM.TMP
SELECTION='TRACE-KEY="level-2"' ORDER=BYTYPE

1. account-name ELEMENT
2. account-type ELEMENT
3. category-number ELEMENT
4. current-total ELEMENT
5. day ELEMENT
6. journal-entry-number ELEMENT
7. month ELEMENT
8. previous-year-total ELEMENT
9. request-end-of-year-close ELEMENT
10. request-general-ledger-update ELEMENT
11. request-revenues-expense-clean ELEMENT
12. source-code ELEMENT
13. subcategory-number ELEMENT
14. update-code ELEMENT
15. value-of-account ELEMENT
16. year-to-now-total ELEMENT
17. entered-account ENTITY
18. general-ledger-account ENTITY
19. previous-entered-account ENTITY
20. account-number GROUP
21. date-of-entry GROUP
22. update-request INPUT
23. balance-error OUTPUT
24. clean-old-entries PROCESS
25. clean-revenues-expense PROCESS
26. end-of-year-update PROCESS
27. update-general-ledger PROCESS
28. entered-account-file SET
29. general-ledger-file SET
30. previous-entered-account-file SET
1 DEFINE ELEMENT account-name ;
2   /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
3   SYNONYMS ARE: acc-n ;
4   DESCRIPTION ;
5   It is a description of each of the
6   general-ledger-account.
7   It will not more than 30 characters . ;
8   TRACE-KEY IS: 'level-3',
9      'level-3.2',
10     'level-2',
11    'level-1';
12  CONTAINED IN: entered-account,
13      general-ledger-account,
14     account-entry,
15      account-information,
16     adjust-account-entry;
17
18 DEFINE ELEMENT account-type ;
19   /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
20   SYNONYMS ARE: acc-ty ;
21   DESCRIPTION ;
22   This element distinguishes the types of
23   general-ledger-accounts. The four different
24   types of accounts are 0)regular 1)title
25      2)total 3)heading.;
26   TRACE-KEY IS: 'level-3',
27      'level-3.2',
28     'level-2',
29    'level-1';
30  CONTAINED IN: general-ledger-account,
31      account-information;
32  VALUES ARE: 0 THRU 3 ;
33
34 DEFINE ELEMENT category-number ;
35   /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
36   SYNONYMS ARE: ca-nb ;
37   DESCRIPTION ;
38   The category number identifies groups,
39  1)assets 2)liabilities 3)income 4) expense,
40   of a general-ledger-file.;
41   TRACE-KEY IS: 'level-3',
42      'level-3.2',
43     'level-2',
44    'level-1',
'level-1';
CONTAINED IN: account-number;
VALUES ARE:
1 THRU 4;

DEFINE ELEMENT current-total;

SYNONYMS ARE: cu-to;
DESCRIPTION:
It indicates the current amount of account.;
TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';
CONTAINED IN: general-ledger-account,
account-information;
VALUES ARE:
-100000000.000000
THRU 100000000.000000;

DEFINE ELEMENT day;

SYNONYMS ARE: d;
DESCRIPTION:
It indicates the day in a month.;
TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';
CONTAINED IN: date-of-entry;
VALUES ARE:
1 THRU 31;

DEFINE ELEMENT journal-entry-number;

SYNONYMS ARE: j-e-n;
DESCRIPTION:
It keeps a sequence number to identify the account entered for a fiscal year.;
TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';
CONTAINED IN: entered-account,
previous-entered-account;
VALUES ARE:
1 THRU 999999;

DEFINE ELEMENT month;

SYNONYMS ARE: m;
DESCRIPTION:
It indicates the month.;
TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';
CONTAINED IN: date-of-entry;
VALUES ARE:
1 THRU 12;
SYNONYMS ARE: m;
DESCRIPTION;
It indicates the month in a year.;
TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';
CONTAINED IN: date-of-entry;
VALUES ARE:
1 THRU 12;

DEFINE ELEMENT previous-year-total;

SYNONYMS ARE: p-y-to;
DESCRIPTION;
It keeps the previous year account amount.;
TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';
CONTAINED IN: general-ledger-account,
account-information;
VALUES ARE:
-100000000.000000
THRU 100000000.000000;

DEFINE ELEMENT request-end-of-year-close;

SYNONYMS ARE: e-o-y-c;
DESCRIPTION;
At the end of the year, all the revenues and expense
accounts are cleaned to zero and all the
previous-year-totals of the assets and liability
accounts should be changed to the value of the
current-total.;
TRACE-KEY IS: 'level-2';
CONTAINED IN: update-request;

DEFINE ELEMENT request-general-ledger-update;

SYNONYMS ARE: g-l-u;
DESCRIPTION;
This element indicates the request for posting the
entered-account to the general-ledger-file.;
TRACE-KEY IS: 'level-2';
CONTAINED IN: update-request;

DEFINE ELEMENT request-revenues-expense-clean;
SYNONYMS ARE: r-e-c;

DESCRIPTION;

It indicates the request for the clean of all the revenues and expense accounts at the end of the period;

TRACE-KEY IS: 'level-2';

CONTAINED IN: update-request;

DEFINE ELEMENT source-code;

SYNONYMS ARE: S-c;

DESCRIPTION;

It indicates which system the accounts come from:

(1) the account receivable system
(2) the account payable system
(3) the payroll system
(4) the adjustment entry;

TRACE-KEY IS: 'level-1', 'level-2', 'level-3', 'level-3.2';

CONTAINED IN: entered-account, account-entry, adjust-account-entry;

VALUES ARE:
1 THRU 4;

DEFINE ELEMENT subcategory-number;

SYNONYMS ARE: su-nb;

DESCRIPTION;

It is an extension to the category number which is a part of account number;

TRACE-KEY IS: 'level-3', 'level-3.2', 'level-2', 'level-1';

CONTAINED IN: account-number;

VALUES ARE:
0 THRU 99999;

DEFINE ELEMENT update-code;

SYNONYMS ARE: up-c;

DESCRIPTION;

It indicates the entered-account-record is used to update the general-ledger-file or not.

1) when the entered-account-record is not used to update the general-ledger-file yet.

2) when the entered-account-record
PSA Version A5.1R5  May 26, 1984 16:22:56  Page A-42

UNIVERSITY OF MONTANA PSA/PSL

Formatted Problem Statement

198 was used to update the general-ledger-file.;
199 TRACE-KEY IS: 'level-3',
200 'level-3.2',
201 'level-2',
202 'level-1';
203 CONTAINED IN: entered-account;

205 DEFINE ELEMENT value-of-account;
206 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
207 SYNONYMS ARE: v-o-acc;
208 DESCRIPTION;
209 This element represents the amount in each account.;
210 TRACE-KEY IS: 'level-3',
211 'level-3.2',
212 'level-2',
213 'level-1';
214 CONTAINED IN: entered-account,
215 previous-entered-account,
216 account-entry,
217 adjust-account-entry;
218 VALUES ARE:
219 -100000000.000000
220 THRU 100000000.000000;

222 DEFINE ELEMENT year-to-now-total;
223 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
224 SYNONYMS ARE: y-t-n-t;
225 DESCRIPTION;
226 This element is part of the general-ledger account.
227 It is used to put the year to the now value for
228 the revenues and the expense accounts.;
229 TRACE-KEY IS: 'level-3',
230 'level-3.2',
231 'level-2',
232 'level-1';
233 CONTAINED IN: general-ledger-account,
234 account-information;
235 VALUES ARE:
236 -100000000.000000
237 THRU 100000000.000000;

239 DEFINE ENTITY entered-account;
240 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
241 SYNONYMS ARE: e-acc;
242 DESCRIPTION;
243 This is the entity of an entered-account-file
244 which contains the whole entered accounts and
245 adjusted accounts. It keeps the whole trail
246 for a period of the business.;
247 TRACE-KEY IS: 'level-0',
248 'level-1',
UNIVERSITY OF MONTANA PSA/PSL
Formatted Problem Statement

`level-2`,

'level-3',

'level-3.2';

COLLECTED IN: entered-account-file;

CONSISTS OF:
- journal-entry-number,
- source-code,
- account-number,
- date-of-entry,
- account-name,
- value-of-account,
- update-code;

ADDED TO:
- entered-account-file
  BY account-maintain;

ADDED TO:
- entered-account-file
  BY enter-account-entries;

ADDED TO:
- entered-account-file
  BY adjust-account-entries;

REFERENCED IN:
- entered-account-file
  BY adjust-account-entries;

REMOVED FROM:
- entered-account-file
  BY account-update;

REMOVED FROM:
- entered-account-file
  BY clean-old-entries;

DEFINE ENTITY
general-ledger-account;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: g-l-acc;

DESCRIPTION;
This is an entity of general-ledger-file.
It keeps the all information about general
ledger account. ;

TRACE-KEY IS: 'level-0',

'level-1',

'level-2',

'level-3',

'level-3.2';

COLLECTED IN: general-ledger-file;

CONSISTS OF:
- account-number,
- account-name,
- account-type,
- current-total,
- year-to-now-total,
DEFINE ENTITY
previous-entered-account;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: p-e-a;
DESCRIPTION;
This is an entity of previous-entered-account-file
which keeps the trail for a fiscal year of business.
It can be used to produce the audit-trail report.;
TRACE-KEY IS: 'level-0',
'level-2',
'level-3';
COLLECTED IN: previous-entered-account-file;
CONSISTS OF:
journal-entry-number,
account-number,
date-of-entry,
value-of-account;
ADDED TO:
previous-entered-account-file
BY account-update;
ADDED TO:
previous-entered-account-file
BY clean-old-entries;
DEFINE GROUP account-number;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: acc-nb;
DESCRIPTION;

It identifies a particular account in the
general-ledger-file. It contains the category
and sub-category numbers.;

TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';

CONSISTS OF:
category-number,
subcategory-number;

CONTAINED IN: entered-account,
general-ledger-account,
previous-entered-account,
account-entry,
account-information,
adjust-account-entry;

IDENTIFIES: general-ledger-account;
ORDERS: general-ledger-file;

DEFINE GROUP date-of-entry;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: d-of-e;
DESCRIPTION;

This element represents the date of making an entry.;

TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';

CONSISTS OF:
day,
month;

CONTAINED IN: entered-account,
previous-entered-account,
account-entry,
adjust-account-entry;

DEFINE INPUT update-request;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: u-r;
DESCRIPTION;

This request indicates the decision to make an update
to the general-ledger-file, close accounts at the end
of the year and clean the revenue and expense accounts
at end of period.;

TRACE-KEY IS: 'level-0',
'level-c',
'level-2';
DEFINE OUTPUT balance-error;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: b-err;
DESCRIPTION;
The output indicates the error message about the inbalance of the accounts.;
TRACE-KEY IS: 'level-c',
'level-0',
'level-2';

DEFINE PROCESS clean-old-entries;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: c-o-e;
DESCRIPTION;
This process cleans the entered-account-file after the entered-accounts are posted to the general-ledger-file. It appends the entered-account to the previous-entered-account-file to keep the audit trial.;
TRACE-KEY IS: 'level-2';
PART OF: account-update;
ADDS: previous-entered-account TO previous-entered-account-file;
REMOVES: entered-account FROM entered-account-file;
EMPLOYs: entered-account-file;
PROCEDURE;
****************************************************
REPEAT the following;
READ entered-account from entered-account-file.
IF update-code is '2' THEN
MOVE corresponding entered-account field to previous-entered-account.
WRITE previous-entered-account to end of previous-entered-account-file.
UNTIL end of entered-account-file.
CLEAN the entered-account-file.
****************************************************
;
DEFINE PROCESS clean-revenues-expense;

SYNONYMS ARE: c-r-e;

DESCRIPTION;
This process cleans all the revenues and expense accounts in the general-ledger-file at the end of the period according to the request.

TRACE-KEY IS: 'level-2';
PART OF: account-update;
UPDATES: general-ledger-file;

***********************************************************************
SET the revenues and expense to 0.
REPEAT the following:
READ the general-ledger-account from general-ledger-file.
IF category-number is 3 or 4,
THEN ,
SELECT the case which applies:
CASE1:
(category-number is 3 'revenues')
ADD current-total to revenues.
ADD current-total to year-to-now-total.
SET current-total to 0.
CASE2:
(category-number is 4 'expense')
ADD current-total to expense.
ADD current-total to year-to-now-total.
SET current-total to 0.
UNTIL end of general-ledger-file.
ADD (revenues-expense) to the retainal-earins account.
***********************************************************************

DEFINE PROCESS end-of-year-update;

SYNONYMS ARE: e-o-y-u;

DESCRIPTION;
This process closes all the assets and liability accounts in the general-ledger-file at the end of the year.

TRACE-KEY IS: 'level-2';
PART OF: account-update;
UPDATES: general-ledger-file;

***********************************************************************
REPEAT the following:
READ the general-ledger-account from general-ledger-file.
UNIVERSITY OF MONTANA PSA/PSL

Formatted Problem Statement

504 IF category-number is 1 or 2,
505 THEN ,
506 MOVE current-total to previous-year-total.
507 UNTIL category-number greater than 2.
508 *******************************************************

511 DEFINE PROCESS
512 update-general-ledger;
513 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
514 SYNONYMS ARE: u-g-1;
515 DESCRIPTION;
516 This process maintains the general-ledger-file .
517 It references the entered-account-file to update the
518 general-ledger-file in the end of the period of the
519 business.;
520 TRACE-KEY IS: 'level-2';
521 GENERATES:
522 balance-error;
523 PART OF: account-update;
524 UPDATES: general-ledger-file
525 USING: entered-account-file;
526 PROCEDURE;
527 *******************************************************
528 INITIAL the debit and credit values to 0.
529 REPEAT the following:
530 READ entered-account from entered-account-file.
531 SELECT the case which applies:
532 CASE1: ( value-of-account is positive )
533 ADD value-of-account to credit.
534 CASE2: ( value-of-account is negative )
535 ADD value-of-account to debit.
536 UNTIL end of entered-account-file.
537 IF credit equal debit (the entered accounts are balance)
538 THEN ,
539 RESET entered-account-file.
540 REPEAT the following:
541 READ entered-account from entered-account-file.
542 SEARCH general-ledger-file for matching
543 account-number.
544 IF found,
545 ADD value-of-account to current-total.
546 MOVE '2' to update-code.
547 OTHERWISE,
548 DISPLAY 'error error !! why
549 account not found?'.
550 UNTIL end of file entered-account-file.
551 OTHERWISE,
552 DISPLAY 'account not balance. can't update.'.
553 *******************************************************

554 ;
DEFINE SET entered-account-file;

SYNONYMS ARE: e-t-f;

DESCRIPTION;

This file contains the whole entered accounts and adjusted accounts. It keeps the whole trail for a period of the business. It will be used to update the general-ledger-file and append to the previous-entered-account-file.

TRACE-KEY IS: 'level-0', 'level-1', 'level-2', 'level-3', 'level-3.2';

COLLECTION OF:

entered-account;

HAS: entered-account
ADD BY account-maintain;

HAS: entered-account
ADD BY enter-account-entries;

HAS: entered-account
ADD BY adjust-account-entries;

USED BY:

report-generator
TO DERIVE account-report;

USED BY:

print-list-of-account-entries
TO DERIVE list-of-account-entries;

USED BY:

print-ledger-sheet
TO DERIVE ledger-sheet;

USED BY:

produce-trial-balance
TO DERIVE trial-balance;

HAS: entered-account
REFERENCED BY adjust-account-entries;

HAS: entered-account
REMOVED BY account-update;

HAS: entered-account
REMOVED BY clean-old-entries;

USED BY:

account-update
TO UPDATE general-ledger-file;

USED BY:

update-general-ledger
TO UPDATE general-ledger-file;

EMPLOYED BY: clean-old-entries;

DEFINE SET general-ledger-file;
SYNONYMS ARE: g-l-f;

DESCRIPTION;

This file contains ledger, total, title and heading accounts. It keeps the all information about general ledger account. So, it always is in balance. It is used to produce the financial reports.

TRACE-KEY IS: 'level-0',
'level-1',
'level-2',
'level-3',
'level-3.2';

COLLECTION OF:
general-ledger-account;

ORDERED BY: account-number;

HAS: general-ledger-account
ADDED BY account-maintain;

HAS: general-ledger-account
ADDED BY maintain-account-information;

USED BY:
report-generator
TO DERIVE financial-report;

USED BY:
report-generator
TO DERIVE account-report;

USED BY:
print-chart-of-account
TO DERIVE chart-of-account;

USED BY:
print-ledger-sheet
TO DERIVE ledger-sheet;

USED BY:
produce-trial-balance
TO DERIVE trial-balance;

USED BY:
print-balance-sheet
TO DERIVE balance-sheet;

USED BY:
print-income-statement
TO DERIVE income-statement;

MAINTAINED BY: account-maintain
USING: account-information;

MAINTAINED BY: maintain-account-information
USING: account-information;

HAS: general-ledger-account
MODIFIED BY account-maintain;

HAS: general-ledger-account
MODIFIED BY maintain-account-information;

HAS: general-ledger-account
REFERENCED BY account-maintain;

HAS: general-ledger-account
657 REFERENCED BY enter-account-entries;
658 HAS: general-ledger-account
659 REMOVED BY account-maintain;
660 HAS: general-ledger-account
661 REMOVED BY maintain-account-information;
662 UPDATED BY: account-update
663 USING: entered-account-file;
664 UPDATED BY: update-general-ledger
665 USING: entered-account-file;
666 UPDATED BY: clean-revenues-expense;
667 UPDATED BY: end-of-year-update;
668
669 DEFINE SET
670 previous-entered-account-file;
671 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
672 SYNONYMS ARE: p-e-a-f;
673 DESCRIPTION;
674 This file keeps the trail for a fiscal year
675 of business. It can be used to produce the
676 audit-trail report.;
677 TRACE-KEY IS: 'level-0',
678    'level-2',
679    'level-3';
680 COLLECTION OF:
681    previous-entered-account;
682 HAS: previous-entered-account
683 ADDED BY account-update;
684 HAS: previous-entered-account
685 ADDED BY clean-old-entries;
686 USED BY:
687    report-generator
688 TO DERIVE account-report;
689 USED BY:
690    print-audit-trail
691 TO DERIVE audit-trail;
692
693 EOF EOF EOF EOF EOF
REQUEST FINANCIAL REPORT → PRINT FINANCIAL REPORT

PRINT FINANCIAL REPORT → FINANCIAL REPORT

CHART OF ACCOUNT REQUEST

PRINT CHART OF ACCOUNT → GENERAL LEDGER FILE

ENTERED ACCOUNT FILE

LIST OF ACCOUNT ENTRIES REQUEST

PRINT LIST OF ACCOUNT ENTRIES

PRINT AUDIT TRAIL

AUDIT TRAIL REQUEST

PREVIOUS ENTERED ACCOUNT FILE
UNIVERSITY OF MONTANA PSA/PSL

Name Selection

Parameters:  DB=FINAL.DBF  PRINT  PUNCH=PSANAM.TMP

SELECTION='TRACE-KEY="level-3"'  ORDER=BYTYPE

1  account-name              ELEMENT  
2  account-type               ELEMENT  
3  category-number            ELEMENT  
4  current-total              ELEMENT  
5  day                        ELEMENT  
6  journal-entry-number       ELEMENT  
7  month                      ELEMENT  
8  previous-year-total        ELEMENT  
9  request-audit-trail        ELEMENT  
10 request-chart-of-acc        ELEMENT  
11 request-list-of-acc-entries ELEMENT  
12 source-code                ELEMENT  
13 subcategory-number         ELEMENT  
14 update-code                ELEMENT  
15 value-of-account           ELEMENT  
16 year-to-now-total          ELEMENT  
17 entered-account            ENTITY  
18 general-ledger-account     ENTITY  
19 previous-entered-account   ENTITY  
20 account-number             GROUP  
21 date-of-entry              GROUP  
22 request-account-report     INPUT  
23 request-financial-report   INPUT  
24 audit-trail                OUTPUT  
25 chart-of-account           OUTPUT  
26 financial-report           OUTPUT  
27 list-of-account-entries    OUTPUT  
28 print-audit-trail          PROCESS  
29 print-chart-of-account     PROCESS  
30 print-financial-report     PROCESS  
31 print-list-of-account-entries PROCESS  
32 entered-account-file       SET  
33 general-ledger-file        SET  
34 previous-entered-account-file SET  

DEFINE ELEMENT account-name;
  /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
  SYNONYMS ARE: acc-n;
  DESCRIPTION;
  It is a description of each of the general-ledger-account.
  It will not more than 30 characters.;
  TRACE-KEY IS: 'level-3',
  'level-3,2',
  'level-2',
  'level-1';
  CONTAINED IN: entered-account,
  general-ledger-account,
  account-entry,
  account-information,
  adjust-account-entry;

DEFINE ELEMENT account-type;
  /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
  SYNONYMS ARE: acc-ty;
  DESCRIPTION;
  This element distinguishes the types of general-ledger-accounts. The four different types of accounts are 0) regular 1) title 2) total 3) heading.;
  TRACE-KEY IS: 'level-3',
  'level-3,2',
  'level-2',
  'level-1';
  CONTAINED IN: general-ledger-account,
  account-information;
  VALUES ARE:
  0 THRU 3;

DEFINE ELEMENT category-number;
  /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
  SYNONYMS ARE: ca-nb;
  DESCRIPTION;
  The category number identifies groups, 1) assets 2) liabilities 3) income 4) expense, of a general-ledger-file.;
  TRACE-KEY IS: 'level-3',
  'level-3,2',
  'level-2',
  'level-1',
DEFINE ELEMENT current-total;
   /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
   SYNONYMS ARE: cu-to;
   DESCRIPTION;
   It indicates the current amount of account.
   TRACE-KEY IS: 'level-3',
   'level-3.2',
   'level-2',
   'level-1';
   CONTAINED IN: general-ledger-account,
   account-information;
   VALUES ARE:
   -100000000.000000
   THRU 100000000.000000;

DEFINE ELEMENT day;
   /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
   SYNONYMS ARE: d;
   DESCRIPTION;
   It indicates the day in a month.
   TRACE-KEY IS: 'level-3',
   'level-3.2',
   'level-2',
   'level-1';
   CONTAINED IN: date-of-entry;
   VALUES ARE:
   1 THRU 31;

DEFINE ELEMENT journal-entry-number;
   /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
   SYNONYMS ARE: j-e-n;
   DESCRIPTION;
   It keeps a sequence number to identify the
   account entered for a fiscal year.
   TRACE-KEY IS: 'level-3',
   'level-3.2',
   'level-2',
   'level-1';
   CONTAINED IN: entered-account,
   previous-entered-account;
   VALUES ARE:
   1 THRU 999999;

DEFINE ELEMENT month;
   /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: m;

DESCRIPTION;

It indicates the month in a year.;

TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';

CONTAINED IN: date-of-entry;

VALUES ARE:
1 THRU 12;

DEFINE ELEMENT previous-year-total;

SYNONYMS ARE: p-y-to;

DESCRIPTION;

It keeps the previous year account amount.;

TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';

CONTAINED IN: general-ledger-account,
account-information;

VALUES ARE:
-100000000.000000
THRU 100000000.000000;

DEFINE ELEMENT request-audit-trail;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */

SYNONYMS ARE: a-t-re;

DESCRIPTION;

It indicates the request for audit trail.;

TRACE-KEY IS: 'level-3';

CONTAINED IN: request-account-report;

USED BY:

print-audit-trail
TO DERIVE audit-trail;

DEFINE ELEMENT request-chart-of-acc;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */

SYNONYMS ARE: c-o-a-re;

DESCRIPTION;

It indicates the request for the chart of account.;

TRACE-KEY IS: 'level-3';

CONTAINED IN: request-account-report;

USED BY:

print-chart-of-account
TO DERIVE chart-of-account;

DEFINE ELEMENT request-list-of-acc-entries;
147 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
148 SYNONYMS ARE: 1-o-a-e-re;
149 DESCRIPTION;
150 It indicates the request for report of the
151 list-of-account-entries.;
152 TRACE-KEY IS: 'level-3';
153 CONTAINED IN: request-account-report;
154
155 DEFINE ELEMENT source-code;
156 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
157 SYNONYMS ARE: s-c;
158 DESCRIPTION;
159 It indicates which system the accounts come from:
160 (1) the account receivable system
161 (2) the account payable system
162 (3) the payroll system
163 (4) the adjustment entry.;
164 TRACE-KEY IS: 'level-1',
165 'level-2',
166 'level-3',
167 'level-3.2';
168 CONTAINED IN: entered-account,
169 account-entry,
170 adjust-account-entry;
171 VALUES ARE:
172 1 THRU 4;
173
174 DEFINE ELEMENT subcategory-number;
175 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
176 SYNONYMS ARE: su-nb;
177 DESCRIPTION;
178 It is an extension to the category number
179 which is a part of account number.;
180 TRACE-KEY IS: 'level-3',
181 'level-3.2',
182 'level-2',
183 'level-1';
184 CONTAINED IN: account-number;
185 VALUES ARE:
186 0 THRU 99999;
187
188 DEFINE ELEMENT update-code;
189 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
190 SYNONYMS ARE: up-c;
191 DESCRIPTION;
192 It indicates the entered-account-record is used
193 to update the general-ledger-file or not.
194 1) when the entered-account-record is not used to
195 update the general-ledger-file yet.
196 2) when the entered-account-record
197 was used to update the general-ledger-file.;
TRACE-KEY IS: 'level-3',
   'level-3.2',
   'level-2',
   'level-1';
CONTAINED IN: entered-account;

DEFINE ELEMENT value-of-account;

DESCRIPTION;
This element represents the amount in each account.;

TRACE-KEY IS: 'level-3',
   'level-3.2',
   'level-2',
   'level-1';
CONTAINED IN: entered-account,
   previous-entered-account,
   account-entry,
   adjust-account-entry;
VALUES ARE:
-100000000.000000
   THRU 100000000.000000;

DEFINE ELEMENT year-to-now-total;

DESCRIPTION;
This element is part of the general-ledger account.
It is used to put the year to the now value for
the revenues and the expense accounts.;

TRACE-KEY IS: 'level-3',
   'level-3.2',
   'level-2',
   'level-1';
CONTAINED IN: general-ledger-account,
   account-information;
VALUES ARE:
-100000000.000000
   THRU 100000000.000000;

DEFINE ENTITY entered-account;

DESCRIPTION;
This is the entity of an entered-account-file
which contains the whole entered accounts and
adjusted accounts. It keeps the whole trail
for a period of the business.;
TRACE-KEY IS: 'level-0',
   'level-1',
   'level-2',
UNIVERSITY OF MONTANA PSA/PSL
Formatted Problem Statement

249 'level-3',
250 'level-3.2';
251 COLLECTED IN: entered-account-file;
252 CONSISTS OF:
253 journal-entry-number,
254 source-code,
255 account-number,
256 date-of-entry,
257 account-name,
258 value-of-account,
259 update-code;
260 ADDED TO:
261 entered-account-file
262 BY account-maintain;
263 ADDED TO:
264 entered-account-file
265 BY enter-account-entries;
266 ADDED TO:
267 entered-account-file
268 BY adjust-account-entries;
269 REFERENCED IN:
270 entered-account-file
271 BY adjust-account-entries;
272 REMOVED FROM:
273 entered-account-file
274 BY account-update;
275 REMOVED FROM:
276 entered-account-file
277 BY clean-old-entries;
278
279 DEFINE ENTITY
general-ledger-account;
280 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
281 SYNONYMS ARE: g-l-acc;
282 DESCRIPTION;
283 This is an entity of general-ledger-file.
284 It keeps the all information about general
285 ledger account.;
286 TRACE-KEY IS: 'level-0',
287 'level-1',
288 'level-2',
289 'level-3',
290 'level-3.2';
291 COLLECTED IN: general-ledger-file;
292 CONSISTS OF:
293 account-number,
294 account-name,
295 account-type,
296 current-total,
297 year-to-now-total,
DEFINE ENTITY
previous-entered-account;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: p-e-a;
DESCRIPTION;
This is an entity of previous-entered-account-file
which keeps the trail for a fiscal year of business.
It can be used to produce the audit-trail report.;
TRACE-KEY IS: 'level-0',
'level-2',
'level-3';
COLLECTED IN: previous-entered-account-file;
CONSISTS OF:
    journal-entry-number,
    account-number,
    date-of-entry,
    value-of-account;
ADDED TO:
    previous-entered-account-file
    BY account-update;
ADDED TO:
    previous-entered-account-file
    BY clean-old-entries;
DEFINE GROUP account-number;
SYNONYMS ARE: acc-nb;

DESCRIPTION;

It identifies a particular account in the general-ledger-file. It contains the category and sub-category numbers.

TRACE-KEY IS: 'level-3',
'sub-category-number';

CONSISTS OF:

category-number,
subcategory-number;

CONTAINED IN: entered-account,
general-ledger-account,
previous-entered-account,
account-entry,
account-information,
adjust-account-entry;

IDENTIFIES: general-ledger-account;
ORDERS: general-ledger-file;

DEFINE GROUP date-of-entry;

SYNONYMS ARE: d-of-e;

DESCRIPTION;

This element represents the date of making an entry.

TRACE-KEY IS: 'level-3',
'sub-category-number';

CONSISTS OF:

day,
month;

CONTAINED IN: entered-account,
previous-entered-account,
account-entry,
account-information,
adjust-account-entry;

DEFINE INPUT request-account-report;

SYNONYMS ARE: r-a-r;

DESCRIPTION;

This indicates the account reports requested.

TRACE-KEY IS: 'level-c',
'sub-category-number';

GENERATED: BY book-keeper;
RECEIVED: BY general-ledger-processing,
BY report-generator,
UNIVERSITY OF MONTANA
PSA/PSL
Formatted Problem Statement

402 BY print-chart-of-account,
403 BY print-list-of-account-entries,
404 BY print-audit-trail;

CONSISTS OF:
request-audit-trail,
request-chart-of-acc,
request-list-of-acc-entries;

409

410 DEFINE INPUT
411 request-financial-report;
412 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
413 SYNONYMS ARE: r-f-r;
414 DESCRIPTION;
415 This data indicates the requests on
416 the financial reports.;
417 TRACE-KEY IS: 'level-c',
418 'level-0',
419 'level-3',
420 'level-3.2';
421 GENERATED: BY chief-accountant;
422 RECEIVED: BY general-ledger-processing,
423 BY report-generator,
424 BY print-financial-report,
425 BY print-ledger-sheet,
426 BY produce-trial-balance,
427 BY print-balance-sheet,
428 BY print-income-statement;

CONSISTS OF:
request-balance-sheet,
request-ledger-sheet,
request-income-statement,
request-trial-balance;

434

435 DEFINE OUTPUT audit-trail;
436 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
437 SYNONYMS ARE: a-t;
438 DESCRIPTION;
439 This report shows the trail of entered accounts
440 for a certain of period.;
441 TRACE-KEY IS: 'level-3';
442 GENERATED: BY print-audit-trail,
443 BY report-generator;
444 RECEIVED: BY book-keeper;
445 PART OF: account-report;
446 DERIVED BY: print-audit-trail
447 USING: previous-entered-account-file,
448 request-audit-trail;
449 LAYOUT;
### JOURNAL ENTRY ACCOUNT SOURCE

<table>
<thead>
<tr>
<th>AMOUNT</th>
<th>ENTRY NO.</th>
<th>DATE</th>
<th>NO.</th>
<th>CODE</th>
</tr>
</thead>
</table>

#### NAME DR CR

<table>
<thead>
<tr>
<th>DR</th>
<th>CR</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Entry</th>
<th>Date</th>
<th>No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>A/P</td>
<td>XX</td>
<td>XXX</td>
</tr>
<tr>
<td>A/R</td>
<td>XX</td>
<td>XXX</td>
</tr>
<tr>
<td>A/P</td>
<td>XX</td>
<td>XXX</td>
</tr>
<tr>
<td>A/R</td>
<td>XX</td>
<td>XXX</td>
</tr>
<tr>
<td>A/P</td>
<td>XX</td>
<td>XXX</td>
</tr>
<tr>
<td>A/R</td>
<td>XX</td>
<td>XXX</td>
</tr>
<tr>
<td>A/P</td>
<td>XX</td>
<td>XXX</td>
</tr>
<tr>
<td>A/R</td>
<td>XX</td>
<td>XXX</td>
</tr>
</tbody>
</table>

Page x of x
DEFINE OUTPUT chart-of-account;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: c-o-acc;
DESCRIPTION;
This report shows the current accounts in the general-ledger-file.;
TRACE-KEY IS: 'level-3';
GENERATED: BY print-chart-of-account, report-generator;
RECEIVED: BY book-keeper;
PART OF: account-report;
DERIVED BY: print-chart-of-account
USING: general-ledger-file,
request-chart-of-acc;
LAYOUT;

<table>
<thead>
<tr>
<th>COMPANY</th>
</tr>
</thead>
<tbody>
<tr>
<td>X OF X</td>
</tr>
</tbody>
</table>

*** BALANCE SHEET ***

<table>
<thead>
<tr>
<th>ACCT NO</th>
<th>ACCT NAME</th>
<th>TYP...</th>
</tr>
</thead>
<tbody>
<tr>
<td>X,XXX.XXCR</td>
<td>XXXXXX</td>
<td></td>
</tr>
<tr>
<td>X,XXX.XXCR</td>
<td>XXXXXX</td>
<td>REG...</td>
</tr>
<tr>
<td>X,XXX.XXCR</td>
<td>XXXXXX</td>
<td>REG...</td>
</tr>
<tr>
<td>X,XXX.XXCR</td>
<td>XXXXXX</td>
<td>REG...</td>
</tr>
</tbody>
</table>
*** INCOME STATEMENT ***

This report shows the financial situation of a business. The reports are generated by general-ledger-processing and report-generator, and the trace-key is 'level-c', 'level-0', and 'level-3'. The reports are derived by the chief-accountant and used to generate financial-report.

This report shows the entered accounts during a certain period. The report is generated by print-list-of-account-entries, and the trace-key is 'level-3'. The report is derived by print-list-of-account-entries.
USING: entered-account-file;

LAYOUT;

JOURNAL ENTRY LIST ...

JOURNAL ENTRY ACCOUNT AMOUNT
ENTRY NO. DATE NO. CODE ...

AME DR CR

PAGE x OF x

DEFINE PROCESS

print-audit-trail;
SYNONYMS ARE: p-a-t;

DESCRIPTION;
This process produces the audit-trail report for a
certain range of period.;
TRACE-KEY IS: 'level-3';

DERIVES:

      audit-trail;

RECEIVES:

      request-account-report;

PART OF:  report-generator;

USING:  previous-entered-account-file,

request-audit-trail;

PROCEDURE;

*************************************************************
(*Please reference the layout report
for the audit-trail.*)

TAKE the start-date and end-date.

SET page-number and line-number equal 1.

(* print the title lines *)

WRITE the "XXXXXXXXXXXX COMPANY" line.

WRITE the start-date "TO" end-date lines.

INCREASE line-number by 8.

(* end of print the title lines *)

REPEAT the following:

(* print heading lines *)

WRITE the "AUDIT...TRAIL...PAGE x OF x" line.

WRITE the "JOURNAL ENTRY ACCOUNT SOURCE AMOUNT".

INCREASE line-number by 6 lines.

(* end of print the heading lines *)

REPEAT the following:

READ previous-entered-account from

previous-entered-file.

IF (day-of-entry GREATER OR EQUAL start-day)

OR

(day-of-entry SMALLER OR EQUAL end-day),

THEN

WRITE journal-entry-number, date-of-entry,

account-name.

SELECT the case which applies:

CASE1: (source-code is 1)

SOURCE = "A/R".

CASE2: (source-code is 2)

SOURCE = "A/P".

CASE3: (source-code is 3)

SOURCE = "P/R".
WRITE SOURCE, account-name.

SELECT the case which applies:

CASE 1: (value-of-account is positive)
WRITE the value-of-account in
the CR column.

CASE 2: (value-of-account is negative)
WRITE the abs(value-of-account)
in the DR column.

INCREASE line-number by 1.
UNTIL line-number equal 50 OR end of
previous-entered-account-file.
INCREASE page-number by 1.
SET line-number to 1.
UNTIL end of previous-entered-account-file.

****************************************************
;
DEFINE PROCESS
print-chart-of-account;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: p-c-o-a;
DESCRIPTION;
This process produces the chart of accounts
according to the request.
TRACE-KEY IS: 'level-3';
GENERATES:
chart-of-account;
RECEIVES:
request-account-report;
PART OF: report-generator;
DERIVES: chart-of-account
USING: general-ledger-file,
request-chart-of-acc;
PROCEDURE;
****************************************************
(*Please reference the layout report
for the chart-of-account.*)
SET page-number and line-number equal 1.
TAKE the date.

(* print the title *)
WRITE the "XXXXXXXXXXXXXXX COMPANY" line.
WRITE the "CHART OF ACCOUNT AS OF xx/xx/xx....".
WRITE the "*** BALANCE SHEET ***" line.
INCREASE the line number by 10.
(* end of the printing the title *)

REPEAT the following:

(* print the heading *)
IF page-number greater than 1,
THEN,
  WRITE the "CHART OF ACCOUNT AS OF....".
  INCREASE line-number by 1. AS OF ...".
  WRITE the "ACCT NO ACCT NAME....".
  ADVANCE 2 lines.
  INCREASE the line number by 4.
  (* end of printing the heading *)
REPEAT the following:
  READ general-ledger-account from general-ledger-file.
  IF account-name is 'INCOME STATEMENT',
  THEN,
  WRITE the "*** INCOME STATEMENT ***".
  INCREASE line-number by 1.
  WRITE the account-number,account-name.
SELECT the case which applies:
  CASE1: (account-type is 0)
    TYPE = "regular".
  CASE2: (account-type is 1)
    TYPE = "title ".
  CASE3: (account-type is 2)
    TYPE = "total ".
  CASE4: (account-type is 3)
    TYPE = "heading".
WRITE the TYPE.
IF account-type is 0 'regular',
  THEN ,
  SELECT the case which applies:
    CASE1 : (current-total = > 0)
    WRITE the [current-total+"CR"].
    CASE2 : (current-total < 0)
    WRITE the [abs(current-total)+"DR"].
INCREASE line-number by 1.
UNTIL line-number equal 50 or end of general-ledger-file.
INCREASE line-number equal 50 or end of general-ledger-file.
SET line-number equal to 1.
UNTIL end of general-ledger-file.
*********************************************************************
;
DEFINE PROCESS
print-financial-report;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: p-f-r;
DESCRIPTION;
This process produces the financial reports.;
TRACE-KEY IS: 'level-3';
GENERATES:
financial-report;
RECEIVES:
  request-financial-report;
SUBPARTS ARE:  produce-trial-balance,
               print-ledger-sheet,
               print-income-statement,
               print-balance-sheet;
PART OF:  report-generator;

DEFINE PROCESS
print-list-of-account-entries;
/*  DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE:  p-l-o-a-e;
DESCRIPTION;
This process produces a list of entered accounts
in a period of a business.;
TRACE-KEY IS:  'level-3';
GENERATES:
  list-of-account-entries;
RECEIVES:
  request-account-report;
PART OF:  report-generator;
DERIVES:  list-of-account-entries
USING:  entered-account-file;
PROCEDURE;
***********************************************************************
(*Please reference the layout report for the
list-of-account-entries*)
SET page-number and line-number equal 1.
TAKE date.

(* print the title lines *)
WRITE the "XXXXXXXXXXXXX COMPANY ... page x "line.
WRITE the "JOURNAL ENTRY LIST AS OF xx/xx/xx".
INCREASE line-number by 8.
(* end of printing the title lines *)

REPEAT the following:

(* print the heading lines *)
IF page-number greater than 1 ,
THEN,
  WRITE the "JOURNAL ENTRY LIST .....xx/xx/xx".
  WRITE the "PAGE x of x".
  INCREASE line-number by 1.
WRITE the "JOURNAL ENTRY ACCOUNT SOURCE ...".
INCREASE line-number by 6.
(* end of printing the heading lines *)

REPEAT the following:
  READ entered-account from
  entered-account-file.
UNIVERSITY OF MONTANA PSA/PSL
Formatted Problem Statement

```plaintext
861 WRITE journal-entry-number,date
862 ,account-number.
863 SELECT the case which applies:
864 CASE1: (source-code is 1)
865     SOURCE = "A/R".
866 CASE2: (source-code is 2)
867     SOURCE = "A/P".
868 CASE3: (source-code is 3)
869     SOURCE = "P/R".
870 WRITE SOURCE,account-name.
871 SELECT the case which applies:
872 CASE1: (value-of-account is postive)
873     WRITE the value-of-account
874     in CR column.
875 CASE2: (value-of-account is negative)
876     WRITE the [abs(value-of-account)]
877     in DR column.
878     INCREASE line-number by 1.
879 UNTIL line-number equal 50 OR the end of
880 entered-account-file.
881     INCREASE page-number by 1.
882 SET line-number to 1.
883 UNTIL the end of entered-account-file.
884 *****************************************
885 ;
886
887 DEFINE SET
888 entered-account-file;
889 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
890 SYNONYMS ARE: e-t-f;
891 DESCRIPTION;
892 This file contains the whole entered accounts and
893 adjusted accounts. It keeps the whole trail for a
894 period of the business. It will be used to update
895 the general-ledger-file and append to the
896 previous-entered-account-file.;
897 TRACE-KEY IS: 'level-0',
898     'level-1',
899     'level-2',
900     'level-3',
901     'level-3.2';
902 COLLECTION OF:
903     entered-account;
904 HAS: entered-account
905     ADDED BY account-maintain;
906 HAS: entered-account
907     ADDED BY enter-account-entries;
908 HAS: entered-account
909     ADDED BY adjust-account-entries;
910 USED BY:
911     report-generator
```
UNIVERSITY OF MONTANA PSA/PSL
Formatted Problem Statement

912 TO DERIVE account-report;
913 USED BY:
914 print-list-of-account-entries
915 TO DERIVE list-of-account-entries;
916 USED BY:
917 print-ledger-sheet
918 TO DERIVE ledger-sheet;
919 USED BY:
920 produce-trial-balance
921 TO DERIVE trial-balance;
922 HAS: entered-account
923 REFERENCED BY adjust-account-entries;
924 HAS: entered-account
925 REMOVED BY account-update;
926 HAS: entered-account
927 REMOVED BY clean-old-entries;
928 USED BY:
929 account-update
930 TO UPDATE general-ledger-file;
931 USED BY:
932 update-general-ledger
933 TO UPDATE general-ledger-file;
934 EMPLOYED BY: clean-old-entries;
935
936 DEFINE SET general-ledger-file;
937 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
938 SYNONYMS ARE: g-l-f;
939 DESCRIPTION;
940 This file contains ledger, total, title and heading
941 accounts. It keeps the all information about
942 general ledger account. So, it always is in balance.
943 It is used to produced the financial reports;
944 TRACE-KEY IS: 'level-0',
945 'level-1',
946 'level-2',
947 'level-3',
948 'level-3.2';
949 COLLECTION OF:
950 general-ledger-account;
951 ORDERED BY: account-number;
952 HAS: general-ledger-account
953 ADDED BY account-maintain;
954 HAS: general-ledger-account
955 ADDED BY maintain-account-information;
956 USED BY:
957 report-generator
958 TO DERIVE financial-report;
959 USED BY:
960 report-generator
961 TO DERIVE account-report;
962 USED BY:
963 print-chart-of-account
964 TO DERIVE chart-of-account;
965 USED BY;
966 print-ledger-sheet
967 TO DERIVE ledger-sheet;
968 USED BY;
969 produce-trial-balance
970 TO DERIVE trial-balance;
971 USED BY;
972 print-balance-sheet
973 TO DERIVE balance-sheet;
974 USED BY:
975 print-income-statement
976 TO DERIVE income-statement;
977 MAINTAINED BY: account-maintain
978 USING: account-information;
979 MAINTAINED BY: maintain-account-information
980 USING: account-information;
981 HAS: general-ledger-account
982 MODIFIED BY account-maintain;
983 HAS: general-ledger-account
984 MODIFIED BY maintain-account-information;
985 HAS: general-ledger-account
986 REFERENCED BY account-maintain;
987 HAS: general-ledger-account
988 REFERENCED BY enter-account-entries;
989 HAS: general-ledger-account
990 REMOVED BY account-maintain;
991 HAS: general-ledger-account
992 REMOVED BY maintain-account-information;
993 UPDATED BY: account-update
994 USING: entered-account-file;
995 UPDATED BY: update-general-ledger
996 USING: entered-account-file;
997 UPDATED BY: clean-revenues-expense;
998 UPDATED BY: end-of-year-update;
999
1000 DEFINE SET
1001 previous-entered-account-file;
1002 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
1003 SYNONYMS ARE: p-e-a-f;
1004 DESCRIPTION;
1005 This file keeps the trail for a fiscal year
1006 of business. It can be used to produce the
1007 audit-trail report.;
1008 TRACE-KEY IS: 'level-0',
1009 'level-2',
1010 'level-3';
1011 COLLECTION OF:
1012 previous-entered-account;
1013 HAS: previous-entered-account
ADDED BY account-update;
HAS: previous-entered-account
ADDED BY clean-old-entries;
USED BY:
report-generator
TO DERIVE account-report;
USED BY:
print-audit-trail
TO DERIVE audit-trail;

EOF EOF EOF EOF EOF
UNIVERSITY OF MONTANA PSA/PSL

Name Selection

Parameters:  DB=FINAL.DBF  PRINT PUNCH=PSANAM.TMP
SELECTION='TRACE-KEY="level-3.2"'  ORDER=BYTYPE

1  account-name  ELEMENT
2  account-type  ELEMENT
3  category-number  ELEMENT
4  current-total  ELEMENT
5  day  ELEMENT
6  journal-entry-number  ELEMENT
7  month  ELEMENT
8  previous-year-total  ELEMENT
9  request-balance-sheet  ELEMENT
10  request-income-statement  ELEMENT
11  request-ledger-sheet  ELEMENT
12  request-trial-balance  ELEMENT
13  source-code  ELEMENT
14  subcategory-number  ELEMENT
15  update-code  ELEMENT
16  value-of-account  ELEMENT
17  year-to-now-total  ELEMENT
18  entered-account  ENTITY
19  general-ledger-account  ENTITY
20  account-number  GROUP
21  date-of-entry  GROUP
22  request-financial-report  INPUT
23  balance-sheet  OUTPUT
24  income-statement  OUTPUT
25  ledger-sheet  OUTPUT
26  trial-balance  OUTPUT
27  print-balance-sheet  PROCESS
28  print-income-statement  PROCESS
29  print-ledger-sheet  PROCESS
30  produce-trial-balance  PROCESS
31  entered-account-file  SET
32  general-ledger-file  SET
1 DEFINE ELEMENT account-name;
   /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
   SYNONYMS ARE: acc-n;
   DESCRIPTION;
   It is a description of each of the general-ledger-account.
   It will not more than 30 characters.;
   TRACE-KEY IS: 'level-3',
   'level-3.2',
   'level-2',
   'level-1';
   CONTAINED IN: entered-account,
   general-ledger-account,
   account-entry,
   account-information,
   adjust-account-entry;

18 DEFINE ELEMENT account-type;
   /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
   SYNONYMS ARE: acc-ty;
   DESCRIPTION;
   This element distinguishes the types of general-ledger-accounts. The four different types of accounts are 0)regular 1)title 2)total 3)heading.;
   TRACE-KEY IS: 'level-3',
   'level-3.2',
   'level-2',
   'level-1';
   CONTAINED IN: general-ledger-account,
   account-information;
   VALUES ARE:
   0 THRU 3;

35 DEFINE ELEMENT category-number;
   /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
   SYNONYMS ARE: ca-nb;
   DESCRIPTION;
   The category number identifies groups, 1)assets 2)liabilities 3)income 4) expense, of a general-ledger-file.;
   TRACE-KEY IS: 'level-3',
   'level-3.2',
   'level-2',
DEFINE ELEMENT current-total;
  /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
  SYNONYMS ARE: cu-to;
  DESCRIPTION;
  It indicates the current amount of account.;
  TRACE-KEY IS: 'level-3',
                'level-3.2',
                'level-2',
                'level-1';
  CONTAINED IN: general-ledger-account,
                 account-information;
  VALUES ARE:
          -100000000.000000     THRU     100000000.000000;

DEFINE ELEMENT day;
  /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
  SYNONYMS ARE: d;
  DESCRIPTION;
  It indicates the day in a month.;
  TRACE-KEY IS: 'level-3',
                'level-3.2',
                'level-2',
                'level-1';
  CONTAINED IN: date-of-entry;
  VALUES ARE:
          1 THRU     31;

DEFINE ELEMENT journal-entry-number;
  /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
  SYNONYMS ARE: j-e-n;
  DESCRIPTION;
  It keeps a sequence number to identify the account entered for a fiscal year.;
  TRACE-KEY IS: 'level-3',
                'level-3.2',
                'level-2',
                'level-1';
  CONTAINED IN: entered-account,
                 previous-entered-account;
  VALUES ARE:
          1 THRU     999999;

DEFINE ELEMENT month;
  /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: m;
DESCRIPTIO;
It indicates the month in a year.;
TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';
CONTAINED IN: date-of-entry;
VALUES ARE:
1 THRU 12;

DEFINE ELEMENT previous-year-total;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: p-y-to;
DESCRIPTIO;
It keeps the previous year account amount.;
TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';
CONTAINED IN: general-ledger-account,
account-information;
VALUES ARE:
-100000000.00000
THRU 100000000.00000;

DEFINE ELEMENT request-balance-sheet;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: b-s-re;
DESCRIPTIO;
It indicates the request for balance sheet.;
TRACE-KEY IS: 'level-3.2';
CONTAINED IN: request-financial-report;
USED BY:
print-ledger-sheet
TO DERIVE balance-sheet;

DEFINE ELEMENT request-income-statement;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: i-s-re;
DESCRIPTIO;
It indicates the request for income statement.;
TRACE-KEY IS: 'level-3.2';
CONTAINED IN: request-financial-report;
USED BY:
print-income-statement
TO DERIVE income-statement;

DEFINE ELEMENT
UNIVERSITY OF MONTANA PSA/PSL

Formatted Problem Statement

147 request-ledger-sheet;
148 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
149 SYNONYMS ARE: 1-s-re;
150 DESCRIPTION;
151 It show the list of account changes.;
152 TRACE-KEY IS: 'level-3.2';
153 CONTAINED IN: request-financial-report;
154 USED BY:
155    print-ledger-sheet
156    TO DERIVE    ledger-sheet;
157
158 DEFINE ELEMENT
159 request-trial-balance;
160 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
161 SYNONYMS ARE: r-t-b;
162 DESCRIPTION;
163 It indicates the request for trail-balance.;
164 TRACE-KEY IS: 'level-3.2';
165 CONTAINED IN: request-financial-report;
166
167 DEFINE ELEMENT    source-code;
168 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
169 SYNONYMS ARE: s-c;
170 DESCRIPTION;
171 It indicates which system the accounts come from:
172 (1) the account receivable system
173 (2) the account payable system
174 (3) the payroll system
175 (4) the adjustment entry.;
176 TRACE-KEY IS: 'level-1',
177 'level-2',
178 'level-3',
179 'level-3.2';
180 CONTAINED IN: entered-account,
181 account-entry,
182 adjust-account-entry;
183 VALUES ARE:
184 1 THRU 4;
185
186 DEFINE ELEMENT    subcategory-number;
187 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
188 SYNONYMS ARE: su-nb;
189 DESCRIPTION;
190 It is an extension to the category number
191 which is a part of account number.;
192 TRACE-KEY IS: 'level-3',
193 'level-3.2',
194 'level-2',
195 'level-1';
196 CONTAINED IN: account-number;
197 VALUES ARE:
DEFINE ELEMENT update-code;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: up-c;

DESCRIPTION;
It indicates the entered-account-record is used to update the general-ledger-file or not.
1) when the entered-account-record is not used to update the general-ledger-file yet.
2) when the entered-account-record was used to update the general-ledger-file.;

TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';

CONTAINED IN: entered-account;

DEFINE ELEMENT value-of-account;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: v-o-acc;

DESCRIPTION;
This element represents the amount in each account.;

TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';

CONTAINED IN: entered-account,
previous-entered-account,
account-entry,
adjust-account-entry;

VALUES ARE:
-100000000.000000 THRU 100000000.000000;

DEFINE ELEMENT year-to-now-total;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: y-t-n-t;

DESCRIPTION;
This element is part of the general-ledger account. It is used to put the year to the now value for the revenues and the expense accounts.;

TRACE-KEY IS: 'level-3',
'level-3.2',
'level-2',
'level-1';

CONTAINED IN: general-ledger-account,
account-information;

VALUES ARE:
-100000000.000000 THRU 100000000.000000;
DEFINE ENTITY entered-account;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: e-acc;
DESCRIPTION;
This is the entity of an entered-account-file which contains the whole entered accounts and adjusted accounts. It keeps the whole trail for a period of the business;
TRACE-KEY IS: 'level-0', 'level-1', 'level-2', 'level-3', 'level-3.2';

COLLECTED IN: entered-account-file;
CONSISTS OF: journal-entry-number,
source-code,
account-number,
date-of-entry,
account-name,
value-of-account,
update-code;

ADDED TO:
entered-account-file
BY account-maintain;

ADDED TO:
entered-account-file
BY enter-account-entries;

ADDED TO:
entered-account-file
BY adjust-account-entries;

REFERENCED IN:
entered-account-file
BY adjust-account-entries;

REMOVED FROM:
entered-account-file
BY account-update;

REMOVED FROM:
entered-account-file
BY clean-old-entries;

DEFINE ENTITY general-ledger-account;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: g-l-acc;
DESCRIPTION;
This is an entity of general-ledger-file. It keeps the all information about general ledger account;
TRACE-KEY IS: 'level-0',
UNIVERSITY OF MONTANA PSA/PSL
Formatted Problem Statement

300 'level-1',
301 'level-2',
302 'level-3',
303 'level-3.2';
304 COLLECTED IN: general-ledger-file;
305 CONSISTS OF:
306 account-number,
307 account-name,
308 account-type,
309 current-total,
310 year-to-now-total,
311 previous-year-total;
312 IDENTIFIED BY: account-number;
313 ADDED TO:
314 general-ledger-file
315 BY account-maintain;
316 ADDED TO:
317 general-ledger-file
318 BY maintain-account-information;
319 MODIFIED IN:
320 general-ledger-file
321 BY account-maintain;
322 MODIFIED IN:
323 general-ledger-file
324 BY maintain-account-information;
325 REFERENCED IN:
326 general-ledger-file
327 BY account-maintain;
328 REFERENCED IN:
329 general-ledger-file
330 BY enter-account-entries;
331 REMOVED FROM:
332 general-ledger-file
333 BY account-maintain;
334 REMOVED FROM:
335 general-ledger-file
336 BY maintain-account-information;
337
338 DEFINE GROUP account-number;
339 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
340 SYNONYMS ARE: acc-nb;
341 DESCRIPTION;
342 It identifies a particular account in the
343 general-ledger-file. It contains the category
344 and sub-category numbers;
345 TRACE-KEY IS: 'level-3',
346 'level-3.2',
347 'level-2',
348 'level-1';
349 CONSISTS OF:
350 category-number,
PSA Version A5.1R5  May 26, 1984 16:50:37  Page A-84

UNIVERSITY OF MONTANA PSA/PSL
Formatted Problem Statement

351 subcategory-number;
352 CONTAINED IN: entered-account,
353 general-ledger-account,
354 previous-entered-account,
355 account-entry,
356 account-information,
357 adjust-account-entry;
358 IDENTIFIES: general-ledger-account;
359 ORDERS: general-ledger-account;

360 DEFINE GROUP date-of-entry;
361 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
362 SYNONYMS ARE: d-of-e;
363 DESCRIPTION;
364 This element represents the date of making an entry;
365 TRACE-KEY IS: 'level-3',
366 'level-3.2',
367 'level-2',
368 'level-1';
369 CONSISTS OF:
370 day,
371 month;
372 CONTAINED IN: entered-account,
373 previous-entered-account,
374 account-entry,
375 adjust-account-entry;

378 DEFINE INPUT
379 request-financial-report;
380 /* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
381 SYNONYMS ARE: r-f-r;
382 DESCRIPTION;
383 This data indicates the requests on
384 the financial reports;
385 TRACE-KEY IS: 'level-c',
386 'level-0',
387 'level-3',
388 'level-3.2';
389 GENERATED: BY chief-accountant;
390 RECEIVED: BY general-ledger-processing,
391 BY report-generator,
392 BY print-financial-report,
393 BY print-ledger-sheet,
394 BY produce-trial-balance,
395 BY print-balance-sheet,
396 BY print-income-statement;
397 CONSISTS OF:
398 request-balance-sheet,
399 request-ledger-sheet,
400 request-income-statement,
401 request-trial-balance;
DEFINE OUTPUT balance-sheet;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: b-s;
DESCRIPTION;
This report shows the assets and liability condition
of a business.;
TRACE-KEY IS: 'level-3.2';
GENERATED: BY print-balance-sheet;
RECEIVED: BY chief-accountant;
PART OF: financial-report;
DERIVED BY: print-balance-sheet
USING: general-ledger-file;
DERIVED BY: print-ledger-sheet
USING: request-balance-sheet;
LAYOUT;

XXXXXXXXXXXXX COMPANY

PAGE x OF x

BALANCE SHEET

AS OF XX/XX/XX

ASSETS

CURRENT ASSETS

CASH

REVENUE BANK $ xx,xxx,xxx.xx
PETTY CASH <xx,xxx,xxx...

x,xxx.xx
TOTAL ASSETS
X,XXX.XX

LIABILITIES & 
STOCKHOLDER' EQUITY

CURRENT LIABILITY

SALES TAX PAYABLE $ XX,XXX,XXX.XX
INCOME TAX PAYABLE
FEDERAL XX,XXX,XXX.XX
STATE <XX,XXX,XXX...

TOTAL LIAB & STOCKHOLDER'S EQ ...
This report shows the net profit for a certain period of time.

TRACE-KEY IS: 'level-3.2';
GENERATED: BY print-income-statement;
PART OF: financial-report;
DERIVED BY: print-income-statement
USING: general-ledger-file,
request-income-statement;

LAYOUT;

XXXXXXXXXXXXXXXXX COM...

INCOME STATEMENT
PAGE x OF x

FOR THE YEAR (PERIOD) ENDT...

REVENUES

SALE OF GOODS

FINISHED GOODS

SALES RETURNS & ALLOWANCES

TOTAL

$ XX,XXX,XXX.XX

...
EXPENSES

COST OF INCOME

COST OF SALE OF GOODS ... 

COST OF CONSULTING ... 

ROYALTY PAYMENTS ... 

VARIANCE EXPENSES ... 

TOTAL EXPENSES $ XX,XXX,XXX.XX

DEFINE OUTPUT ledger-sheet;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: f-f;
DESCRIPTION;
This report shows the flow in each account in a period of time;
TRACE-KEY IS: 'level-3.2';
GENERATED: BY print-ledger-sheet;
PART OF: financial-report;
DERIVED BY: print-ledger-sheet
USING: entered-account-file,
    general-ledger-file,
    request-ledger-sheet;
LAYOUT;

$ XX,XXX,XXX.XX
<table>
<thead>
<tr>
<th>ACCOUNT NAME</th>
<th>XX,XXX,XXX.XX</th>
<th>OPENING BALANCE</th>
<th>XX,XXX,XXX.XX</th>
<th>DR (OR CR)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACCOUNT NO.</td>
<td>XX,XXX,XXX.XX</td>
<td>COLSING BALANCE</td>
<td>XX,XXX,XXX.XX</td>
<td>DR (OR CR)</td>
</tr>
</tbody>
</table>
P.S. FOR EVERY REGULAR ACCOUNT

```plaintext
DEFINE OUTPUT trial-balance;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: t-b;
DESCRIPTION;
This reports show the accounts balance.
It is received by chief accountant to
decide what account needs an adjustment.;
TRACE-KEY IS: 'level-3.2';
GENERATED: BY produce-trial-balance;
PART OF: financial-report;
DERIVED BY: produce-trial-balance
USING: entered-account-file,
general-ledger-file;
```

```
LAYOUT;
```

```
<table>
<thead>
<tr>
<th>ACCOUNT NO.</th>
<th>ACCOUNT NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>xxxxxxxx</td>
<td>REVENUE BANK(regular)</td>
</tr>
<tr>
<td>XXXXXXXX</td>
<td>PETTYCASH(regular)</td>
</tr>
</tbody>
</table>
```

Page x of x
DEFINE PROCESS
print-balance-sheet;

/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: p-b-s;
DESCRIPTION;
This process produces the balance sheet to show the
business's financial situation .;
TRACE-KEY IS: 'level-3.2';
GENERATES:
balance-sheet;
RECEIVES:
request-financial-report;
PART OF: print-financial-report;
DERIVES: balance-sheet
USING: general-ledger-file;
PROCEDURE;
(*Please reference the layout report
for the balance sheet*)
SET the accumulators to 0
(acc[0],acc[1],acc[2].......acc[7],acc[8]).
SET index of accumulator to 8 (acc[index]).
SET page-number and line-number to 1.
TAKE date.
UNIVERSITY OF MONTANA PSA/PSL
Formatted Problem Statement

(* print the title *)
WRITE the "XXXXXXXXXXXXX COMPANY" line.
INCREASE line-number by 4.
(* end of printing the title *)

REPEAT the following:

(* print the heading *)
WRITE the "BALANCE SHEET ... PAGE x OF x".
INCREASE line-number by 5.
(* end of printing the heading *)

REPEAT the following:
READ the general-ledger-account from general-ledger-file.
IF category-number is 1 or 2,
THEN,
SELECT the case which allpies:
CASE1: (account-type is 'hading')
DECREASE index by l.
WRITE account-name.
CASE2: (account-type is 'regular')
WRITE account-name.
IF current-total is positive,
THEN,
WRITE current-total in the value column.
OTHERWISE,
WRITE "<abs(current-total)>" in the value column.
ADD current-total to acc[index] through acc[8].
CASE3: (account-type is 'total')
WRITE account-name.
IF acc[index] is positive,
THEN,
WRITE acc[index] in the total column.
OTHERWISE,
WRITE "<abs(acc[index])>" in the total column.
CLEAN acc[0] through acc[index] to 0.
INCREASE the index by 1.
INCREASE line-number by 1.
OTHERWISE,
DISPLAY"This account is not ASSETS OR LIE".
UNTIL line-number equal 50 OR the category-number greater than 2.
INCREASE page-number by 1.
SET line-number to 1.
UNTIL the category-number greater than 2.
DEFINE PROCESS
print-income-statement;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: p-in-s;
DESCRIPTION;
This process produces the income statement to
show the loss and earnings of the business in
a period of time.;
TRACE-KEY IS: 'level-3.2';
GENERATES:
income-statement;
RECEIVES:
request-financial-report;
PART OF: print-financial-report;
DERIVES: income-statement
USING: general-ledger-file,
request-income-statement;
PROCEDURE;
(*Please reference the layout report
for the income statement *)
SET the accumulators to 0
(acc[0], acc[1], acc[2]........acc[7], acc[8]).
SET index of accumulator to 8 (acc[index]).
SET page-number and line-number to 1.
TAKE date.

(* print the title *)
WRITE the "XXXXXXXXXXXXX COMPANY" line.
INCREASE line-number by 4.
(* end of printing the title *)

REPEAT the following :
(* print the heading *)
WRITE the "INCOME STATEMENT ... PAGE x OF x".
INCREASE line-number by 5.
(* end of printing the heading *)

REPEAT the following:
READ the general-ledger-account from
genral-ledger-file.
IF category-number is 3 or 4,
THEN,
SELECT the case which allpies:
CASE1: (account-type is 'heading')
DECREASE index by 1.
WRITE account-name.
CASE2: (account-type is 'regular')
WRITE account-name.
IF current-total is postive,
THEN,
WRITE current-total in the value column.
OTHERWISE,
WRITE "<abs(current-total)>",
in the value column.
ADD current-total to acc[index] through acc[8].
CASE3: (account-type is 'total')
WRITE account-name.
IF acc[index] is postive,
THEN,
WRITE acc[index] in the total column.
OTHERWISE,
WRITE "<abs(acc[index])>",
in the total column.
CLEAN acc[0] through acc[index] to 0.
INCREASE the index by 1.
INCREASE line-number by 1.
OTHERWISE,
DISPLAY "This account not for this report"
UNTIL line-number equal 50 OR end
of the general-ledger-file.
INCREASE page-number by 1.
SET line-number to 1.
UNTIL the end of the general-ledger-file.
****************************************************

DEFINE PROCESS print-ledger-sheet;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: p-f-f;
DESCRIPTION;
This process produces the ledger sheet
according to the request.;
TRACE-KEY IS: 'level-3.2';
GENERATES:
ledger-sheet;
RECEIVES:
request-financial-report;
PART OF: print-financial-report;
DERIVES: ledger-sheet
USING: entered-account-file,
general-ledger-file,
request-ledger-sheet;
DERIVES: balance-sheet
USING: request-balance-sheet;
PROCEDURE;

(*Please reference the layout report
for the ledger sheet*)

TAKE date.

SET page-number and line-number to 1.

(* print the title *)
WRITE the "XXXXXXXXXXXXX COMPANY ....xx/xx/xx".
WRITE the "LEDGER SHEET ... PAGE x OF x" line.
INCREASE line-number by 6.

(* end of printing title *)

REPEAT the following:

SET net-change to 0.
READ general-ledger-account from the
general-ledger-file.
IF account-type is 0 (regular),
THEN,
REPEAT the following :

(* print heading *)
WRITE the "ACCOUNT SOURCE JOURNAL
AMOUNT" lines.
INCREASE the line-number by 3.
(* end of printing heading *)

READ the entered-account from the
tened-account-file.

IF the account-number of the
entered-account and the
general-ledger-account are equal.
THEN ,
WRITE account-number.
SELECT the case which applies:
CASE1: (source-code is 1)
SOURCE = "A/R".
CASE2: (source0code is 2)
SOURCE = "A/P".
CASE3: (source-code is 3)
SOURCE = "P/R".
WRITE SOURCE,
journal-entry-number ,date.
SELECT the case which applies:
CASE1 :
(value-of-account is positive)
WRITE the value-of-account
in CR column.
CASE2 :
UNIVERSITY OF MONTANA PSA/PSL
Formatted Problem Statement

(value-of-account is negative)
WRITE abs(value-of-account)
ADD the value-of-account to net-change.
UNTIL end of file entered-account-file or line-number>50.

SET line-number to 1.
INCREASE page-number by 1.
UNTIL end of file entered-account-file.
WRITE "NET CHANGE ",abs(net-change).
IF net-change > 0 THEN WRITELN "CR",
OTHERWISE WRITELN "DR".
WRITE the "ACCOUNT NAME ..." line.
WRITE the account-name line.
WRITE "OPEN BALANCE",abs(current-total) following by "CR" or "DR".
IF current-total > 0 THEN WRITE "CR",
OTHERWISE WRITE the "DR".
WRITE "CLOSE BALANCE",abs(net-change+current-total) following by "CR" or "DR".
IF net-change+current-total > 0 THEN WRITE "CR",
OTHERWISE WRITELN "DR".
UNTIL the end of general-ledger-file.
********************************************************

DEFINE PROCESS
produce-trial-balance;
/* DATE OF LAST CHANGE - May 25, 1984, 00:01:07 */
SYNONYMS ARE: p-t-b;
DESCRIPTION;
This process produces the trial balance to make the accounts balance and provide the information for making an adjustment.;
TRACE-KEY IS: 'level-3.2';
GENERATES:
  trial-balance;
RECEIVES:
  request-financial-report;
  PART OF: print-financial-report;
DERIVES: trial-balance
  USING: entered-account-file,
general-ledger-file;
PROCEDURE;
********************************************************
(*Please reference the layout report for the trial balance*)
TAKE date.
SET page-number and line-number to 1.
SET amount,credit and debit are 0.
Formatted Problem Statement

(* print the title *)
WRITE the "XXXXXXXXXXXX COMPANY ... xx/xx/xx".
INCREASE line-number by 4,
(* end of printing title *)

REPEAT the following:

(* print the heading *)
WRITE the "TRIAL BALANCE SHEET ... PAGE x of x".
WRITE the "ACCOUNT ACCOUNT AMOUNT ......" lines.
INCREASE line-number by 8.
(* end of printing the heading *)

REPEAT the following:

READ the general-ledger-account from
general-ledger-file.
IF account-type in general-ledger-account
is 0 (regular),
THEN,
REPEAT the following:
READ entered-account from
entered-account-file.
IF account-number entered-account and
general-ledger-account are equal.
THEN,
ADD value-of-account to amount.
IF value-of-account is postive.
THEN,
ADD value-of-account to credit.
OTHERWISE,
ADD value-of-account to debit.
UNTIL end of entered-account-file.
WRITE account-number,account-name.
IF amount is postive,
THEN,
WRITE the amount in the CR colmun.
OTHERWISE,
WRITE the abs(amount) in the DR column.
UNTIL line-number equal 50 OR end of
general-ledger-file.
SET line-number to 1.
INCREASE page-number by 1.
UNTIL the end of general-ledger-file.
WRITE the "TOTAL ",debit," = ",credit line.
**********************************************************

DEFINE SET
entered-account-file;
This file contains the whole entered accounts and adjusted accounts. It keeps the whole trail for a period of the business. It will be used to update the general-ledger-file and append to the previous-entered-account-file.

TRACE-KEY IS: 'level-0', 'level-1', 'level-2', 'level-3', 'level-3.2'.

COLLECTION OF:
entered-account;
HAS: entered-account
ADDED BY account-maintain;
HAS: entered-account
ADDED BY enter-account-entries;
HAS: entered-account
ADDED BY adjust-account-entries;
USED BY:
report-generator
TO DERIVE account-report;
USED BY:
print-list-of-account-entries
TO DERIVE list-of-account-entries;
USED BY:
print-ledger-sheet
TO DERIVE ledger-sheet;
USED BY:
produce-trial-balance
TO DERIVE trial-balance;
HAS: entered-account
REFERENCED BY adjust-account-entries;
HAS: entered-account
REMOVED BY account-update;
HAS: entered-account
REMOVED BY clean-old-entries;
USED BY:
account-update
TO UPDATE general-ledger-file;
USED BY:
update-general-ledger
TO UPDATE general-ledger-file;
EMPLOYED BY: clean-old-entries;

DEFINE SET general-ledger-file;
This file contains ledger, total, title and heading accounts. It keeps all information about general ledger account. So, it always is in balance. It is used to produce the financial reports.

TRACE-KEY IS: 'level-0', 'level-1', 'level-2', 'level-3', 'level-3.2';

COLLECTION OF:
    general-ledger-account;
ORDERED BY: account-number;
HAS: general-ledger-account
ADDED BY account-maintain;
HAS: general-ledger-account
ADDED BY maintain-account-information;
USED BY:
    report-generator
    TO DERIVE financial-report;
USED BY:
    report-generator
    TO DERIVE account-report;
USED BY:
    print-chart-of-account
    TO DERIVE chart-of-account;
USED BY:
    print-ledger-sheet
    TO DERIVE ledger-sheet;
USED BY:
    produce-trial-balance
    TO DERIVE trial-balance;
USED BY:
    print-balance-sheet
    TO DERIVE balance-sheet;
USED BY:
    print-income-statement
    TO DERIVE income-statement;
MAINTAINED BY: account-maintain
USING: account-information;
MAINTAINED BY: maintain-account-information
USING: account-information;
HAS: general-ledger-account
MODIFIED BY account-maintain;
HAS: general-ledger-account
MODIFIED BY maintain-account-information;
HAS: general-ledger-account
REFERENCED BY account-maintain;
HAS: general-ledger-account
REFERENCED BY enter-account-entries;
HAS: general-ledger-account
REMOVED BY account-maintain;
1167 HAS: general-ledger-account
1168 REMOVED BY maintain-account-information;
1169 UPDATED BY: account-update
1170 USING: entered-account-file;
1171 UPDATED BY: update-general-ledger
1172 USING: entered-account-file;
1173 UPDATED BY: clean-revenues-expense;
1174 UPDATED BY: end-of-year-update;
1175
1176 EOF EOF EOF EOF EOF
UNIVERSITY OF MONTANA PSA/PSL
Name Selection

Parameters:  DB=ccw.dbf  PRINT  PUNCH=PSANAM.TMP  EMPTY
            SELECTION='LAYOUT'  ORDER=BYTYPE

1  audit-trail  output
2  balance-sheet  output
3  chart-of-account  output
4  income-statement  output
5  ledger-sheet  output
6  list-of-account-entries  output
7  trial-balance  output
### Layout Report

**Parameters:** DB=CCU.DBF FILE=PSAANM.TMP PURCH=PSALO.TMP WIDTH=132

#### Audit Trail

<table>
<thead>
<tr>
<th>Entry No.</th>
<th>Date</th>
<th>Account</th>
<th>Source Code</th>
<th>Account Name</th>
<th>DR</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>XX/XX/XX</td>
<td>XXXX</td>
<td>A/P</td>
<td>XXXXXXXXXX</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>XX/XX/XX</td>
<td>XXXX</td>
<td>A/P</td>
<td>XXXXXXXXXX</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>XX/XX/XX</td>
<td>XXXX</td>
<td>A/R</td>
<td>XXXXXXXXXX</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>XX/XX/XX</td>
<td>XXXX</td>
<td>A/R</td>
<td>XXXXXXXXXX</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>XX/XX/XX</td>
<td>XXXX</td>
<td>A/R</td>
<td>XXXXXXXXXX</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>XX/XX/XX</td>
<td>XXXX</td>
<td>A/R</td>
<td>XXXXXXXXXX</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Notes:**
- PSA352: No more lines left in LAYOUT. The rest are blanked.
### Balance Sheet

**Company Name:** XXXXXXXXXXXX

**Date:** XX/XX/XX

<table>
<thead>
<tr>
<th><strong>ASSETS</strong></th>
<th><strong>CURRENT ASSETS</strong></th>
<th><strong>LIABILITIES &amp; STOCKHOLDER'S EQUITY</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CASH</strong></td>
<td>( \text{rev. bank} ) &amp; ( \text{petty cash} )</td>
<td>( \text{sales tax payable} ) &amp; ( \text{income tax payable} )</td>
</tr>
<tr>
<td>( \text{XX,XXX,XXX.XX} )</td>
<td>( \text{XX,XXX,XXX.XX} )</td>
<td>( \text{XX,XXX,XXX.XX} ) &amp; ( \text{XX,XXX,XXX.XX} )</td>
</tr>
</tbody>
</table>

**TOTAL ASSETS** \( \text{XX,XXX,XXX.XX} \)

**TOTAL LIAB & STOCKHOLDER'S EQ** \( \text{XX,XXX,XXX.XX} \)

*PSA352: No more lines left in LAYOUT. The rest are blanked.*
### Chart of Account

<table>
<thead>
<tr>
<th>Acct No</th>
<th>Acct Name</th>
<th>Type</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>XXXXXX</td>
<td>XXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXX</td>
<td>TITLE</td>
<td>$</td>
</tr>
<tr>
<td>XXXXXX</td>
<td>XXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXX</td>
<td>HEADING</td>
<td></td>
</tr>
<tr>
<td>XXXXXX</td>
<td>XXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXX</td>
<td>REGULAR</td>
<td>XX,XXX,XXX,XXCR</td>
</tr>
<tr>
<td>XXXXXX</td>
<td>XXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXX</td>
<td>REGULAR</td>
<td>XX,XXX,XXX,XXCR</td>
</tr>
<tr>
<td>XXXXXX</td>
<td>XXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXX</td>
<td>TOTAL</td>
<td></td>
</tr>
</tbody>
</table>

### Income Statement

<table>
<thead>
<tr>
<th>Acct No</th>
<th>Acct Name</th>
<th>Type</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>XXXXXX</td>
<td>XXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXX</td>
<td>TITLE</td>
<td>$</td>
</tr>
<tr>
<td>XXXXXX</td>
<td>XXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXX</td>
<td>HEADING</td>
<td></td>
</tr>
<tr>
<td>XXXXXX</td>
<td>XXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXX</td>
<td>REGULAR</td>
<td>XX,XXX,XXX,XXCR</td>
</tr>
<tr>
<td>XXXXXX</td>
<td>XXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXX</td>
<td>REGULAR</td>
<td>XX,XXX,XXX,XXCR</td>
</tr>
<tr>
<td>XXXXXX</td>
<td>XXXXXXXXXX XXXXXXXXXXXXXXXXXXXXXXXXXXX</td>
<td>TOTAL</td>
<td></td>
</tr>
</tbody>
</table>

**Notes:**
- No more lines left in LAYOUT. The rest are blanked.
# Income Statement

XXX Company

**Income Statement**

For the year (period) ending **XX/XX/XX**

<table>
<thead>
<tr>
<th>Item</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Revenues</td>
<td>$ XX,XXX,XXX,XX</td>
</tr>
<tr>
<td>Cost of Income</td>
<td></td>
</tr>
<tr>
<td>Cost of Sale of Goods</td>
<td>$ XX,XXX,XXX,XX</td>
</tr>
<tr>
<td>Cost of Consulting</td>
<td>XX,XXX,XXX,XX</td>
</tr>
<tr>
<td>Royalty Payments</td>
<td>XX,XXX,XXX,XX</td>
</tr>
<tr>
<td>Variance Expenses</td>
<td>$ XX,XXX,XXX,XX</td>
</tr>
<tr>
<td>Total Expenses</td>
<td>$ XX,XXX,XXX,XX</td>
</tr>
</tbody>
</table>

**Note:** No more lines left in layout, the rest are blanked.
### Layout Report

<table>
<thead>
<tr>
<th>ACCOUNT NO.</th>
<th>SOURCE CODE</th>
<th>JOURNAL</th>
<th>LEGER SHEET</th>
<th>DATE</th>
<th>DR</th>
<th>CR</th>
<th>ACCOUNT</th>
<th>Source</th>
<th>Journal</th>
<th>Ledger</th>
<th>Sheet</th>
<th>Page</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ACCOUNT NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

### Opening Balance

XX.XX.XX.XX.XX.XX.XX

### Closing Balance

XX.XX.XX.XX.XX.XX.XX

### Notes

- For every regular account, no more lines left in LAYOUT. The rest are blanked.

---

**P.S.**
### JOURNAL ENTRY LIST AS OF XX/XX/XX

<table>
<thead>
<tr>
<th>ENTRY NO.</th>
<th>DATE</th>
<th>ACCOUNT NO.</th>
<th>SOURCE</th>
<th>AMOUNT</th>
<th>ACCOUNT NAME</th>
<th>DR</th>
<th>CR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**List of Account Entries**

PSA352: No more lines left in LAYOUT. The rest are blanked.
7. trial-balance

**ACCOUNT NAME**

<table>
<thead>
<tr>
<th>ACCOUNT NO.</th>
<th>ACCOUNT NAME</th>
<th>DR</th>
<th>AMOUNT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>REVENUE BANK (regular)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PETTY CASH (regular)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SALES TAX PAYABLE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FEDERAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>STATE</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**TOTAL** 

\[ XX,XXX,XXX.XX = XX,XXX,XXX.XX \]

*No more lines left in Layout. The rest are blanked.*
1
2 DEFINE DATA-STRUCTURE
3 entered-acc-record;
4 # Last changed - May 26, 1984 15:42:46
5 SYNONYMS ARE e-l-r;
6 DESCRIPTION;
7 This data-structure describe the file of the entered
8 account file.;
9 CONSTRUCTED OF E-acc-number,
10 E-acc-name,
11 E-acc-value,
12 E-source-code,
13 E-update-indicator,
14 E-date-of-entry,
15 E-entry-number;
16 KEYWORD 'record';
17
18 DEFINE DATA-STRUCTURE
19 general-ledger-record;
20 # Last changed - May 26, 1984 15:42:46
21 SYNONYMS ARE g-l-r;
22 DESCRIPTION;
23 This data-structure describe the file of the general
24 ledger file.;
25 CONSTRUCTED OF G-acc-number,
26 G-acc-name,
27 G-acc-current-total,
28 G-acc-pre-total,
29 G-acc-type;
30 KEYWORD 'record';
31
32 DEFINE DATA-STRUCTURE
33 pre-entered-acc-record;
34 # Last changed - May 26, 1984 15:45:08
35 SYNONYMS ARE p-e-l-r;
36 DESCRIPTION;
37 This data-structure describe the file of the previous
38 entered account file.;
39 CONSTRUCTED OF P-acc-number,
40 P-acc-name,
41 P-acc-value,
42 P-source-code,
43 P-update-indicator,
44 P-entry-number,
45 P-date-of-entry;
46 KEYWORD 'record';
47
48 DEFINE DEVICE entered-account-file;
49 # Last changed - May 26, 1984 15:47:36
50 SYNONYMS ARE e-a-f;
51 DESCRIPTION;
This file contains the whole entered entries and adjusted entries. It keeps the whole trail for a period of the business.

ACCESSSED SEQUENTIAL;
FORMAT IS RECORD;
DEVICE USAGE input/output;
KEYWORD 'files';

DEFINE DEVICE gennreal-ledger-file;
# Last changed - May 26, 1984 15:47:36
SYNONYMS ARE g-l-f;
DESCRIPTION;
This file keeps the all information about general ledger account. It always is in balance.
ACCESSSED DIRECT;
FORMAT IS RECORD;
DEVICE USAGE input/output;
KEYWORD 'files';

DEFINE DEVICE previous-entered-account-file;
# Last changed - May 26, 1984 15:47:36
SYNONYMS ARE p-e-a-f;
DESCRIPTION;
This file keeps the trail for a fiscal year of business.
ACCESSSED SEQUENTIAL;
FORMAT IS RECORD;
DEVICE USAGE input/output;
KEYWORD 'files';

DEFINE GLOBAL-VARIABLE E-acc-name;
# Last changed - May 26, 1984 15:50:32
SYNONYMS ARE e-a-nm;
DESCRIPTION;
This is a field in entered-account-file refering to account name.
KEYWORD 'maintain-routine';
USED TO CONSTRUCT entered-acc-record;
KNOWN-BY print-ledger-sheet-line;

DEFINE GLOBAL-VARIABLE E-acc-number;
# Last changed - May 26, 1984 15:50:32
SYNONYMS ARE e-a-n;
DESCRIPTION;
This is a field in entered-account-file refering to account number.
KEYWORD 'maintain-routine';
USED TO CONSTRUCT entered-acc-record;
KNOWN-BY update-G-L-F, print-ledger-sheet-line;

DEFINE GLOBAL-VARIABLE E-acc-value;
SYNONYMS ARE e-a-v;
DESCRIPTION;
This is a field in entered-account-file referring
to account values.;
KEYWORD 'maintain-routine';
USED TO CONSTRUCT entered-acc-record;
KNOWN-BY update-G-L-F,
print-ledger-sheet-line;

DEFINE GLOBAL-VARIABLE
E-date-of-entry;
SYNONYMS ARE e-d-o-e;
DESCRIPTION;
This is a field in entered-account-file referring
to date of entry.;
KEYWORD 'maintain-routine';
USED TO CONSTRUCT entered-acc-record;
KNOWN-BY print-ledger-sheet-line;

DEFINE GLOBAL-VARIABLE
E-entry-number;
SYNONYMS ARE e-e-n;
DESCRIPTION;
This is a field in entered-account-file referring to
count number.;
KEYWORD 'maintain-routine';
USED TO CONSTRUCT entered-acc-record;
KNOWN-BY get-current-entry-number,
print-ledger-sheet-line;

DEFINE GLOBAL-VARIABLE
E-source-code;
SYNONYMS ARE e-s-c;
DESCRIPTION;
This is a field in entered-account-file referring
to source code.;
KEYWORD 'maintain-routine';
USED TO CONSTRUCT entered-acc-record;
KNOWN-BY print-ledger-sheet-line;

DEFINE GLOBAL-VARIABLE
E-update-indicator;
SYNONYMS ARE e-u-i;
DESCRIPTION;
This is a field in entered-account-file referring
to update indicator.;
KEYWORD 'maintain-routine';
USED TO CONSTRUCT entered-acc-record;
KNOWN-BY clean-old-entries,
end-of-year-update.
162
163
164 DEFINE GLOBAL-VARIABLE
165 G-acc-current-total;
166 # Last changed - May 26, 1984 15:50:32
167 SYNONYMS ARE g-a-c-t;
168 DESCRIPTION;
169 This is a field in general-ledger-file referring to
170 account current total.;
171 KEYWORD 'maintain-routine';
172 USED TO CONSTRUCT general-ledger-record;
173 KNOWN-BY clean-revenues-expense,
174 end-of-year-update,
175 update-G-L-F,
176 print-income-statement-line,
177 print-chart-of-acc-line;
178
179 DEFINE GLOBAL-VARIABLE
180 G-acc-name;
181 # Last changed - May 26, 1984 15:50:32
182 SYNONYMS ARE g-a-nm;
183 DESCRIPTION;
184 This is a field in general-ledger-file referring to
185 account name.;
186 KEYWORD 'maintain-routine';
187 USED TO CONSTRUCT general-ledger-record;
188 KNOWN-BY add-g-l-record,
189 print-income-statement-line,
190 print-chart-of-acc-line;
191
192 DEFINE GLOBAL-VARIABLE
193 G-acc-number;
194 # Last changed - May 26, 1984 15:50:32
195 SYNONYMS ARE g-a-n;
196 DESCRIPTION;
197 This is a field in general-ledger-file referring to
198 account number.;
199 KEYWORD 'maintain-routine';
200 USED TO CONSTRUCT general-ledger-record;
201 KNOWN-BY add-g-l-record,
202 clean-revenues-expense,
203 end-of-year-update,
204 update-G-L-F,
205 print-chart-of-acc-line;
206
207 DEFINE GLOBAL-VARIABLE
208 G-acc-pre-total;
209 # Last changed - May 26, 1984 15:48:21
210 SYNONYMS ARE g-a-p-t;
211 DESCRIPTION;
212 This is a field in general-ledger-file referring
213 to account previous year total.;
214 KEYWORD 'maintain-routine';
215 USED TO CONSTRUCT general-ledger-record;
216 KNOWN-BY end-of-year-update;
DEFINE GLOBAL-VARIABLE G-acc-type;
# Last changed - May 26, 1984 15:50:32
SYNONYMS ARE g-a-t;
DESCRIPTION;
This is a field in general-ledger-file referring to account type.;
KEYWORD 'maintain-routine';
USED TO CONSTRUCT general-ledger-record;
KNOWN-BY add-g-l-record,
print-income-statement-line,
print-chart-of-acc-line;

DEFINE GLOBAL-VARIABLE P-acc-name;
# Last changed - May 26, 1984 15:50:32
SYNONYMS ARE p-a-nm;
DESCRIPTION;
This is a field in pre-entered-account-file referring to account name.;
KEYWORD 'maintain-routine';
USED TO CONSTRUCT pre-entered-acc-record;
KNOWN-BY print-audit-trail-line;

DEFINE GLOBAL-VARIABLE P-acc-number;
# Last changed - May 26, 1984 15:50:32
SYNONYMS ARE p-a-n;
DESCRIPTION;
This is a field in pre-entered-account-file referring to account number.;
KEYWORD 'maintain-routine';
USED TO CONSTRUCT pre-entered-acc-record;
KNOWN-BY print-audit-trail-line;

DEFINE GLOBAL-VARIABLE P-acc-value;
# Last changed - May 26, 1984 15:50:32
SYNONYMS ARE p-a-v;
DESCRIPTION;
This is a field in pre-entered-account-file referring to account value.;
KEYWORD 'maintain-routine';
USED TO CONSTRUCT pre-entered-acc-record;
KNOWN-BY print-audit-trail-line;

DEFINE GLOBAL-VARIABLE P-date-of-entry;
# Last changed - May 26, 1984 15:50:32
SYNONYMS ARE p-d-o-e;
DESCRIPTION;
This is a field in pre-entered-acc-file referring to date of entry.;
KEYWORD 'maintain-routine';
DEFINE GLOBAL-VARIABLE
P-entry-number;
SYNONYMS ARE p-e-n;
DESCRIPTION;
This is a field in pre-entered-account-file referring to entry number.;
KEYWORD 'maintain-routine';
USED TO CONSTRUCT pre-entered-acc-record;
KNOWN-BY get-current-entry-number,
print-audit-trail-line;

DEFINE GLOBAL-VARIABLE
P-source-code;
SYNONYMS ARE p-s-c;
DESCRIPTION;
This is a field in pre-entered-account-file referring to source code.;
KEYWORD 'maintain-routine';
USED TO CONSTRUCT pre-entered-acc-record;
KNOWN-BY print-audit-trail-line;

DEFINE GLOBAL-VARIABLE
P-update-indicator;
SYNONYMS ARE p-u-i;
DESCRIPTION;
This is a field in pre-entered-account-file referring to update indicator.;
KEYWORD 'maintain-routine';
USED TO CONSTRUCT pre-entered-acc-record;

DEFINE GLOBAL-VARIABLE
sys-date;
SYNONYMS ARE s-d;
DESCRIPTION;
It is the date of data entered.;
KEYWORD 'routine-maintain';
KNOWN-BY print-list-of-account-entered,
print-balance-sheet-heading;

DEFINE VARIABLE acc-name;
SYNONYMS ARE a-nm;
DESCRIPTION;
This is a local variable being tested for the validity of the account name.;
KEYWORD 'maintain-routine';
LOCAL-DATA FOR get-valid-acc-record;
LOCAL-DATA FOR add-g-l-record;
LOCAL-DATA FOR modify-g-l-record;

DEFINE VARIABLE acc-number;
SYNONYMS ARE a-n;
DESCRIPTION;
This is a local variable being tested for the validity of the account number.

KEYWORD 'maintain-routine';
LOCAL-DATA FOR get-valid-acc-number;
LOCAL-DATA FOR add-g-l-record;

DEFINE VARIABLE acc-number-error;
SYNONYMS ARE a-n-e;
DESCRIPTION;
It indicates whether an account name is valid or not.

KEYWORD 'maintain-routine';
LOCAL-DATA FOR get-valid-acc-number;
LOCAL-DATA FOR add-g-l-record;

DEFINE VARIABLE acc-type;
SYNONYMS ARE a-t;
DESCRIPTION;
This is a local variable being tested for the validity of the account type.

KEYWORD 'maintain-routine';
LOCAL-DATA FOR add-g-l-record;
LOCAL-DATA FOR modify-g-l-record;

DEFINE VARIABLE acc-type-error;
SYNONYMS ARE a-t-e;
DESCRIPTION;
It indicates whether an account name is valid or not.

KEYWORD 'maintain-routine';
LOCAL-DATA FOR add-g-l-record;
LOCAL-DATA FOR modify-g-l-record;

DEFINE VARIABLE acc-update-type;
SYNONYMS ARE a-v;
DESCRIPTION;
This is a local variable being tested for the validity of the account value.

KEYWORD 'maintain-routine';
LOCAL-DATA FOR get-valid-acc-record;
LOCAL-DATA FOR get-valid-acc-value;

DEFINE VARIABLE acc-value-error;

SYNONYMS ARE a-v-e;

DESCRIPTION;

It indicates whether an account value is valid or not;

KEYWORD 'maintain-routine';

LOCAL-DATA FOR get-valid-acc-value;

DEFINE VARIABLE account-maintain-type;

SYNONYMS ARE a-m-t;

DESCRIPTION;

It indicates the type of account-maintenance;

KEYWORD 'maintain-routine';

LOCAL-DATA FOR account-maintain-process;

PARAMETER FOR take-type-of-acc-maintain

PASSED-BY result;

DEFINE VARIABLE adjust-or-not;

SYNONYMS ARE a-o-n;

DESCRIPTION;

It indicates whether the entry is an adjusted entry or not;

KEYWORD 'maintain-routine';

LOCAL-DATA FOR account-maintain-process;

PARAMETER FOR append-to-entered-acc-file

PASSED-BY result;

PARAMETER FOR get-valid-acc-record

PASSED-BY result;

DEFINE VARIABLE amount-value;

LOCAL-DATA FOR print-trial-balance;

PARAMETER FOR print-trial-balance-line

PASSED-BY value;

PARAMETER FOR calculate-current-acc-value

PASSED-BY result;

DEFINE VARIABLE begin-date;

SYNONYMS ARE b-d;

DESCRIPTION;

Is is the initial date for the audit trail report.

; KEYWORD 'report-routine';

LOCAL-DATA FOR print-audit-trail;

PARAMETER FOR print-audit-trail-title

PASSED-BY value;

DEFINE VARIABLE clean-old-entry-error;

SYNONYMS ARE c-o-e;

DESCRIPTION;

It indicates any error in the process of
437 clean-old-entry.;
438 LOCAL-DATA FOR clean-old-entries;
439
440 DEFINE VARIABLE current-entry-number;
441 # Last changed - May 26, 1984 15:45:08
442 SYNONYMS ARE c-e-n;
443 DESCRIPTION;
444 This entry number keeps track of the entered entries.;
445 KEYWORD 'maintain-routine';
446 LOCAL-DATA FOR get-valid-acc-record;
447
448 DEFINE VARIABLE delete-confirm;
449 # Last changed - May 26, 1984 15:45:08
450 SYNONYMS ARE d-c;
451 DESCRIPTION;
452 This confirm is used for double checking the deletion
453 of general ledger account.;
454 KEYWORD 'maintain-routine';
455 LOCAL-DATA FOR delete-g-l-record;
456
457 DEFINE VARIABLE end-date;
458 # Last changed - May 26, 1984 15:53:46
459 SYNONYMS ARE e-d;
460 DESCRIPTION;
461 Is is the ending date for the audit trail report.
462 ;
463 KEYWORD 'report-routine';
464 LOCAL-DATA FOR print-audit-trail;
465 PARAMETER FOR print-audit-trail-title
466 PASSED-BY value;
467
468 DEFINE VARIABLE end-of-year-update-error;
469 # Last changed - May 26, 1984 15:49:44
470 DESCRIPTION;
471 It indicates any error in the process of
472 end-of-year-update.;
473 LOCAL-DATA FOR end-of-year-update;
474
475 DEFINE VARIABLE line-number;
476 # Last changed - May 26, 1984 15:53:46
477 SYNONYMS ARE l-n;
478 DESCRIPTION;
479 It is a counter for line number.
480 ;
481 KEYWORD 'report-routine';
482 LOCAL-DATA FOR print-audit-trail;
483 LOCAL-DATA FOR print-balance-sheet;
484 LOCAL-DATA FOR print-chart-of-account;
485 LOCAL-DATA FOR print-income-statement;
486 LOCAL-DATA FOR print-ledger-sheet;
487 LOCAL-DATA FOR print-list-of-account-entered;
488 LOCAL-DATA FOR print-trial-balance;
489
490 DEFINE VARIABLE maintain-acc-info-type;
491 # Last changed - May 26, 1984 15:45:08
SYNONYMS ARE m-a-i-t;
DESCRIPTION;
It indicates the type of account-informance-maintenance.;
KEYWORD 'maintain-routine';
LOCAL-DATA FOR maintain-acc-info;
PARAMETER FOR select-acc-info-maintain
PASSED-BY result;

DEFINE VARIABLE maintain-more;
SYNONYMS ARE m-m;
DESCRIPTION;
It indicates whether more account information is to be maintained or not.;
KEYWORD 'maintain-routine';
LOCAL-DATA FOR maintain-acc-info;

DEFINE VARIABLE modify-type;
SYNONYMS ARE m-t;
DESCRIPTION;
It indicates which field of the general ledger account is to be modified.;
KEYWORD 'maintain-routine';
LOCAL-DATA FOR modify-g-l-record;

DEFINE VARIABLE net-change;
SYNONYMS ARE e-c;
DESCRIPTION;
Is is used to calculate the net change for the ledger sheet.

; KEYWORD 'report-routine';
LOCAL-DATA FOR print-ledger-sheet;

DEFINE VARIABLE page-number;
SYNONYMS ARE p-n;
DESCRIPTION;
It indicates the page number of reports.

; KEYWORD 'report-routine';
LOCAL-DATA FOR print-audit-trail;
LOCAL-DATA FOR print-balance-sheet;
LOCAL-DATA FOR print-chart-of-account;
LOCAL-DATA FOR print-income-statement;
LOCAL-DATA FOR print-ledger-sheet;
LOCAL-DATA FOR print-list-of-account-entered;
LOCAL-DATA FOR print-trial-balance;
PARAMETER FOR print-trial-balance-title
PASSED-BY value;
PARAMETER FOR print-trial-balance-heading
PASSED-BY value;
PARAMETER FOR print-balance-sheet-heading
PASSED-BY

PARAMETER FOR

print-income-statement-heading
value;

PARAMETER FOR

print-chart-of-acc-heading
value;

PARAMETER FOR

print-audit-trail-heading
value;

PARAMETER FOR

print-list-acc-entered-heading
value;

PARAMETER FOR

print-ledger-sheet-heading
value;

DEFINE VARIABLE

repeat-counter;
SYNONYMS ARE r-c;
DESCRIPTION;
It is an indicator for counter the total page number.

KEYWORD
'report-routine';
LOCAL-DATA FOR
print-audit-trail;
LOCAL-DATA FOR
print-ledger-sheet;
LOCAL-DATA FOR
print-list-of-account-entered;
LOCAL-DATA FOR
print-trial-balance;

DEFINE VARIABLE
report-type;
SYNONYMS ARE S-C;
DESCRIPTION;
This is a local variable being tested for the validity
of the source code.;
KEYWORD
'maintain-routine';
LOCAL-DATA FOR
get-valid-acc-record;
LOCAL-DATA FOR
get-valid-source-code;

DEFINE VARIABLE
source-code-error;
SYNONYMS ARE S-C-E;
DESCRIPTION;
It indicates whether an source code is valid or not.;
KEYWORD
'maintain-routine';
LOCAL-DATA FOR
get-valid-source-code;

DEFINE VARIABLE
stop-enter;
SYNONYMS ARE S-E;
DESCRIPTION;
It indicates whether more entries to be made.;
KEYWORD
'maintain-routine';
LOCAL-DATA FOR
append-to-entered-acc-file;
DEFINE VARIABLE total-CR;
SYNONYMS ARE t-c;
DESCRIPTION;
Is is used to calculate the total credit for the trial balance sheet.

KEYWORD 'report-routine';
LOCAL-DATA FOR print-trial-balance;

DEFINE VARIABLE total-DR;
SYNONYMS ARE t-d;
DESCRIPTION;
Is is used to calculate the total debit for the trial balance sheet.

KEYWORD 'report-routine';
LOCAL-DATA FOR print-trial-balance;

DEFINE VARIABLE total-income;
SYNONYMS ARE t-p-n;
DESCRIPTION;
This represents the balance of revenues and expenses for a particular period.

LOCAL-DATA FOR clean-revenues-expense;

DEFINE VARIABLE total-page-number;
SYNONYMS ARE t-p-n;
DESCRIPTION;
It indicates the total page of a report.

KEYWORD 'report-routine';
LOCAL-DATA FOR print-audit-trail;
LOCAL-DATA FOR print-ledger-sheet;
LOCAL-DATA FOR print-list-of-account-entered;
LOCAL-DATA FOR print-trial-balance;
PARAMETER FOR print-trial-balance-title
PASSED-BY value;
PARAMETER FOR print-trial-balance-heading
PASSED-BY value;
PARAMETER FOR print-balance-sheet-heading
PASSED-BY value;
PARAMETER FOR print-income-statement-heading
PASSED-BY value;
PARAMETER FOR print-chart-of-acc-heading
PASSED-BY value;
PARAMETER FOR print-audit-trail-heading
PASSED-BY value;
PARAMETER FOR print-list-acc-entered-heading
PASSED-BY value;
PARAMETER FOR print-ledger-sheet-heading
PASSED-BY value;
657 DEFINE VARIABLE try-again;
658 # Last changed - May 26, 1984 15:45:08
659 SYNONYMS ARE t-a;
660 DESCRIPTION;
661 If the entered account number is not found the system
662 prompts for the stop-enter value, it can either be
663 for re-enter or quit.;
664 KEYWORD 'maintain-routine';
665 LOCAL-DATA FOR modify-g-l-record;
666 PARAMETER FOR get-valid-acc-record
667 PASSED-BY result;
668 PARAMETER FOR get-valid-acc-number
669 PASSED-BY result;
670 PARAMETER FOR get-valid-source-code
671 PASSED-BY result;
672 PARAMETER FOR get-valid-acc-value
673 PASSED-BY result;
674 DEFINE VARIABLE type-of-module;
675 # Last changed - May 26, 1984 15:45:08
676 SYNONYMS ARE t-o-m;
677 DESCRIPTION;
678 It indicates the type of module to be processed.;
679 KEYWORD 'maintain-routine';
680 LOCAL-DATA FOR general-ledger-process;
681 PARAMETER FOR take-module-type-process
682 PASSED-BY result;
683 DEFINE VARIABLE valid-acc-number;
684 # Last changed - May 26, 1984 15:45:08
685 SYNONYMS ARE v-a-n;
686 DESCRIPTION;
687 This is checked and valid account number.;
688 KEYWORD 'maintain-routine';
689 PARAMETER FOR get-valid-acc-number
690 PASSED-BY result;
691 DEFINE VARIABLE valid-acc-value;
692 # Last changed - May 26, 1984 15:45:08
693 SYNONYMS ARE v-a-v;
694 DESCRIPTION;
695 This is checked and valid account value.;
696 KEYWORD 'maintain-routine';
697 PARAMETER FOR get-valid-acc-value
698 PASSED-BY result;
699 DEFINE VARIABLE valid-current-entry-number;
700 # Last changed - May 26, 1984 15:45:08
701 SYNONYMS ARE v-c-e-n;
702 DESCRIPTION;
703 This is checked and valid current entry number.;
704 KEYWORD 'maintain-routine';
705 PARAMETER FOR get-current-entry-number
706 PASSED-BY result;
DEFINE VARIABLE valid-source-code;

# Last changed - May 26, 1984 15:45:08

SYNONYMS ARE v-s-c;

DESCRIPTION;

This is checked and valid source code.

KEYWORD 'maintain-routine';

PARAMETER FOR get-valid-source-code

PASSED-BY result;

DEFINE VARIABLE value-error;

# Last changed - May 26, 1984 15:50:32

SYNONYMS ARE v-e;

DESCRIPTION;

It indicates the entered type of value error.

KEYWORD 'maintain-routine';

LOCAL-DATA FOR take-module-type-process;

LOCAL-DATA FOR take-type-of-acc-maintain;

LOCAL-DATA FOR select-acc-info-maintain;

LOCAL-DATA FOR select-type-acc-update;

LOCAL-DATA FOR select-report-type;

732 lines printed. 395 statements printed.
DEFINE MODULE account-maintain;

# Last changed - May 26, 1984 16:03:08
SYNONYMS ARE a-m;
DESCRIPTION;
This module handles those things that relate to the account information, account entry and adjust-account-entry.

KEYWORD 'moduls';
INTERFACE ROUTINE account-maintain-process;
ROUTINE take-type-of-acc-maintain,
append-to-entered-acc-file,
maintain-acc-info;
MODULE IN general-ledger-system;

DEFINE MODULE account-update;
# Last changed - May 26, 1984 16:03:08
SYNONYMS ARE a-u;
DESCRIPTION;
This process uses the entered-account-file to update the general-ledger-file at the end of the period. It cleans the entered-account-file and also appends to the previous-entered-account-file. It also cleans the revenues and expense accounts at the end of a period. At the end of a year, it will close the accounts,

KEYWORD 'moduls';
INTERFACE ROUTINE account-update-process;
ROUTINE select-type-acc-update,
update-G-l-F,
clean-old-entries,
clean-revenues-expense,
end-of-year-update;
MODULE IN general-ledger-system;

DEFINE MODULE report-generator;
# Last changed - May 26, 1984 16:03:08
SYNONYMS ARE r-g;
DESCRIPTION;
This module generates the financial-report and account-report, when requested.

KEYWORD 'moduls';
INTERFACE ROUTINE report-generator-process;
ROUTINE select-report-type,
print-chart-of-account,
print-list-of-account-entered,
print-audit-trail,
print-ledger-sheet,
print-trial-balance,
print-balance-sheet,
print-income-statement;
MODULE IN general-ledger-system;
DEFINE ROUTINE account-maintain-process;
SYNONYMS ARE a-m-p;
ALGORITHM;
MOVE 'n' to adjust-or-not.
REPEAT
PERFORM take-type-of-acc-maintain.
CASE account-maintain-type:
  '1': PERFORM append-to-entered-acc-file
  '2': MOVE 'y' to adjust-or-not
  '3': PERFORM maintain-acc-info.
UNTIL account-maintain-type = 'x'.
;  DESCRIPTION;
This is an interface routine for
account-maintain module.;
CALLS append-to-entered-acc-file,
maintain-acc-info,
take-type-of-acc-maintain;
KEYWORD 'routin-maintain';
LOCAL-DATA IS adjust-or-not;
LOCAL-DATA IS account-maintain-type;
CALLED-BY general-ledger-process;
INTERFACE ROUTINE FOR
account-maintain;

DEFINE ROUTINE account-update-process;
SYNONYMS ARE a-u-p;
ALGORITHM;
REPEAT
PERFORM select-type-acc-update.
CASE acc-update-type:
  '1': PERFORM update-G-L-F
  '2': PERFORM clean-old-entries
  '3': PERFORM clean-revenues-expense
  '4': PERFORM end-of-year-update
UNTIL acc-update-type = 'x'.
;  DESCRIPTION;
This is the interface routine for the
account-update module.;
CALLS select-type-acc-update,
update-G-L-F,
clean-revenues-expense,
clean-old-entries,
end-of-year-update;
KEYWORD 'update-routine';
LOCAL-DATA IS acc-update-type;
CALLED-BY general-ledger-process;
INTERFACE ROUTINE FOR
107 account-update;

109 DEFINE ROUTINE general-ledger-process;

110 # Last changed - May 26, 1984 16:03:08
111 SYNONYMS ARE g-l-p;
112 ALGORITHM;

114 REPEAT
115 PERFORM take-module-type-process.
116 CASE type-of-module:
117 '1': PERFORM account-maintain-process
118 '2': PERFORM account-update-process
119 '3': PERFORM report-generator-process.
120 UNTIL type-of-module = 'x'.
121 ;

122 DESCRIPTION;
123 This is a interface routine of general-ledger-system.
124 ;
125 CALLS account-maintain-process,
126 account-update-process,
127 report-generator-process;
128 KEYWORD 'sub-sys';
129 LOCAL-DATA IS type-of-module;
130 MAIN ROUTINE FOR general-ledger-system;

132 DEFINE ROUTINE report-generator-process;
133 # Last changed - May 26, 1984 15:50:32
134 SYNONYMS ARE r-g-p;
135 ALGORITHM;
136

137 REPEAT
138 PERFORM select-report-type.
139 CASE report-type:
140 '1': PERFORM print-chart-of-account.
141 '2': PERFORM print-list-of-account-entered.
142 '3': PERFORM print-audit-trail.
143 '4': PERFORM print-ledger-sheet.
144 '5': PERFORM print-trial-balance.
145 '6': PERFORM print-income-statement.
146 '7': PERFORM print-balance-sheet.
147 UNTIL report-type = 'x'.
148 ;

149 DESCRIPTION;
150 This routine is interface routine of report-generator
151 module. It calls the requested report routine.;
152 CALLS select-report-type,
153 print-chart-of-account,
154 print-list-of-account-entered,
155 print-audit-trail,
156 print-ledger-sheet,
157 print-trial-balance,
158 print-income-statement,
159 print-balance-sheet;
160 KEYWORD 'report-routine';
161 LOCAL-DATA IS report-type;
162 CALLED-BY general-ledger-process;
163 INTERFACE ROUTINE FOR
164     report-generator;
165
166 DEFINE SUBSYSTEM general-ledger-system;
167 # Last changed - May 26, 1984 16:03:08
168 SYNONYMS ARE g-l-s;
169 DESCRIPTION;
170 The general-ledger system is part of accounting
171 system. It keeps the accounts in balance and
172 produce the financial reports according to the
173 request.;
174 DESIGNER;
175 Chung-Chieh Wu;
176 KEYWORD 'subsystem';
177 MAIN ROUTINE IS general-ledger-process;
178 MODULE report-generator,
179     account-update,
180     account-maintain;
180 lines printed. 64 statements printed.
1
2 DEFINE LIBRARY general-ledger-library;
3 # Last changed - May 26, 1984 15:50:32
4 SYNONYMS ARE g-l-l;
5 DESCRIPTION;
6 This is a library routine for the General ledger system.;
7 COLLECTION OF get-valid-acc-number,
8 print-company-title;
9 KEYWORD 'maintain-routine';
10
11 DEFINE LIBRARY-ROUTINE
get-valid-acc-number;
12 # Last changed - May 26, 1984 15:42:46
13 SYNONYMS ARE g-v-a-n;
14 ALGORITHM;
15
16 REPEAT
17 MOVE 'n' to acc-number-error.
18 MOVE 'y' to try-again.
19 ACCEPT acc-number.
20 IF acc-number length <> 6 THEN
21 acc-number-error = 'y'
22 ELSE IF acc-number not in general-ledger-file
23 THEN acc-number-error = 'y'
24 display 'account number error'
25 ACCEPT try-again
26 ELSE move acc-number to valid-acc-number.
27 UNTIL acc-number-error = 'n' OR try-again= 'n'.
28 erase error message.
29 ;
30 DESCRIPTION;
31 This routine gets the valid account number.;
32 KEYWORD 'routin-maintain';
33 LOCAL-DATA IS acc-number;
34 LOCAL-DATA IS acc-number-error;
35 PARAMETER try-again PASSED-BY result;
36 PARAMETER valid-acc-number
37 PASSED-BY result;
38 COLLECTED IN general-ledger-library;
39 UTILIZED BY get-valid-acc-record,
40 delete-g-l-record,
41 modify-g-l-record;
42
43 DEFINE MODULE take-module-type;
44 # Last changed - May 26, 1984 15:41:45
45 SYNONYMS ARE p-t-s;
46 DESCRIPTION;
47 This module handles the selection of the type of the
48 module to be processed,
52 KEYWORD 'moduls';
53 INTERFACE ROUTINE take-module-type-process;
54
55 DEFINE ROUTINE append-to-entered-acc-file;
56 # Last changed - May 26, 1984 15:42:46
57 SYNONYMS ARE a-t-e-a-f;
58 ALGORITHM;
59
60 IF adjust-or-not = 'y' THEN source-code:= '4'.
61 PERFORM get-entry-info-screen.
62 OPEN entered-account-file
63 previous-entered-account-file
64 general-ledger-file.
65 WHILE not eof(entered-acc-file)
66 BEGIN
67 read (entered-acc-file)
68 END {while}
69 REPEAT
70 PERFORM get-valid-acc-record
71 IF try-again = 'y' THEN
72 write(entered-acc-file)
73 ACCEPT stop-enter
74 UNTIL stop-enter = 'y' OR try-again = 'n'.
75 CLOSE entered-account-file
76 previous-entered-account-file
77 general-ledger-file.
78
79 DESCRIPTION;
80 This routine accept entered account entries
81 or adjust account entries and append to the
82 end of entered-account-file.;
83 CALLS get-entry-info-screen,
84 get-valid-acc-record;
85 KEYWORD 'routin-maintain';
86 LOCAL-DATA IS stop-enter;
87 PARAMETER adjust-or-not PASSED-BY
88 result;
89 CALLED-BY account-maintain-process;
90 ROUTINE IN account-maintain;
91
92 DEFINE ROUTINE get-current-entry-number;
93 # Last changed - May 26, 1984 15:42:46
94 SYNONYMS ARE g-c-e-n;
95 ALGORITHM;
96 REPEAT
97 READ (entered-account-file).
98 UNTIL EOF(entered-account-file).
99 IF E-entry-number = ' ' THEN
100 REPEAT
101 READ (pre-entered-account-file)
102 UNTIL EOF(pre-entered-account-file)
103 MOVE P-entry-number to valid-current-entry-number
104 ELSE Move E-entry-number to valid-current-entry-number.
105 ;
106 DESCRIPTION;
This routine gets the current entry number.

KEYWORD 'routin-maintain';

KNOWS-OF P-entry-number,

E-entry-number;

PARAMETER valid-current-entry-number

PASSED-BY result;

CALLED-BY get-valid-acc-record;

DEFINE ROUTINE get-entry-info-screen;

SYNONYMS ARE g-e-i-s;

ALGORITHM;

DISPLAY SCREEN(2,33).

DISPLAY 'ENTER ACCOUNT '.

DISPLAY SCREEN(5,10)

DISPLAY '+++++++++++++++++++++++++++++++++++++++++++++++'.

DISPLAY SCREEN(6,10).

DISPLAY 'DATE ** ** **'.

DISPLAY SCREEN(6,60).

DISPLAY 'SOURCE CODE _'.

DISPLAY SCREEN(8,10).

DISPLAY 'ENTRY NUMBER _'.

DISPLAY SCREEN(10,10).

DISPLAY 'ACCOUNT NUMBER _'.

DISPLAY SCREEN(12,10).

DISPLAY 'ACCOUNT NAME ____________________________'.

DISPLAY SCREEN(14,10).

DISPLAY 'ACCOUNT VALUE _______='.;

DESCRIPTION;

This routine shows the screen for enter entries

or enter adjusting entries.

CALLED-BY append-to-entered-acc-file;

DEFINE ROUTINE get-valid-acc-record;

SYNONYMS ARE g-v-a-r;

ALGORITHM;

MOVE 'y' to try-again.

PERFORM get-current-entry-number.

PERFORM get-valid-acc-number.

IF try-again = 'y' THEN

MOVE G-account-name to account-name.

IF adjust-or-not <> 'y' THEN

PERFORM get-valid-acc-value.

IF source-code <> '4' THEN

PERFORM get-valid-source-code.

DESCRIPTION;

This routine accept the valid entered account entry

or valid adjust account entry.

CALLS get-current-entry-number,
get-valid-acc-value.
get-valid-source-code; 'routin-maintain';
KEYWORD get-valid-source-code;
LOCAL-DATA IS current-entry-number;
LOCAL-DATA IS acc-name;
LOCAL-DATA IS source-code;
LOCAL-DATA IS acc-value;
PARAMETER adjust-or-not PASSED-BY result;
PARAMETER try-again PASSED-BY result;
UTILIZES get-valid-acc-number;
CALLED-BY append-to-entered-acc-file;

DEFINE ROUTINE get-valid-acc-value;
SYNONYMS ARE g-v-a-v;
ALGORITHM;

REPEAT
MOVE 'n' to acc-value-error.
MOVE 'y' to try-again.
ACCEPT acc-value.
IF acc-value length <> 11 THEN
acc-value-error = 'y'
ELSE IF acc-value < -100000000 OR acc-value > 100000000 THEN acc-value-error = 'y'
display 'account value error'
ACCEPT try-again
ELSE move acc-value to valid-acc-value.
UNTIL acc-value-error = 'n' OR try-again = 'n'.

DESCRIPTION;
This routine gets the valid account value.;
KEYWORD 'routin-maintain';
LOCAL-DATA IS acc-value;
LOCAL-DATA IS acc-value-error;
PARAMETER try-again PASSED-BY result;
PARAMETER valid-acc-value PASSED-BY result;
CALLED-BY get-valid-acc-record;

DEFINE ROUTINE get-valid-source-code;
SYNONYMS ARE g-v-s-c;
ALGORITHM;

REPEAT
MOVE 'n' to source-code-error.
MOVE 'y' to try-again.
ACCEPT source-code.
IF source-code length <> 1 THEN
source-code-error = 'y'
ELSE IF source-code not in general-ledger-file
   THEN acc-number-error = 'y'
   display 'source code error'
   ACCEPT try-again
   ELSE move source-code to valid-source-code.
   UNTIL source-code-error = 'n' OR try-again = 'n'.
erase error message.

DESCRIPTION;
This routine gets the valid source code.;
KEYWORD
'routine-maintain';
LOCAL-DATA IS
source-code;
LOCAL-DATA IS
source-code-error;
PARAMETER
try-again PASSED-BY
result;
PARAMETER
valid-source-code PASSED-BY
result;
CALLED-BY
get-valid-acc-record;

DEFINE ROUTINE select-module-type-screen;
# Last changed - May 26, 1984 15:54:08
SYNONYMS ARE
s-m-t-s;
ALGORITHM;

DISPLAY SCREEN(2,27) .
DISPLAY 'GENERAL LEDGER SYSTEM'.
DISPLAY SCREEN(4,20) .
DISPLAY 'SELECT ONE OF FOLLOWING TYPE TO PROCESS'.
DISPLAY SCREEN(5,32) .
DISPLAY 'ENTER SELECTION _'.
DISPLAY SCREEN(7,18) .
DISPLAY '================================='.
DISPLAY SCREEN(9,18) .
DISPLAY ' * : FOR EXIT THE G-L-S.'
DISPLAY SCREEN(11,18) .
DISPLAY ' 1 : ACCOUNT MAINTAIN PROCESS'.
DISPLAY SCREEN(13,18) .
DISPLAY ' 2 : ACCOUNT UPDATE PROCESS'.
DISPLAY SCREEN(15,18) .
DISPLAY ' 3 : REPORT GENERATOR PROCESS'.
DISPLAY SCREEN(17,18) .
DISPLAY '================================='.

DESCRIPTION;
This screen shows the three categorys process type.

CALLED-BY
    take-module-type-process;

DEFINE ROUTINE take-acc-maintain-type-screen;
# Last changed - May 26, 1984 15:54:08
SYNONYMS ARE
t-a-m-t-s;
ALGORITHM;

DISPLAY SCREEN(2,27) .
272 DISPLAY '*** ACCOUNT MAINTAIN ***'.
273 DISPLAY SCREEN(4,20).
274 DISPLAY 'SELECT ONE OF FOLLOWING TYPE TO PROCESS '.
275 DISPLAY SCREEN(5,32).
276 DISPLAY 'ENTER SELECTION _'.
277 DISPLAY SCREEN(7,15).
278 DISPLAY '================================='.  
279 DISPLAY SCREEN(9,15).
280 DISPLAY ' * : FOR EXIT THE ACCOUNT MAINTAIN.
281 DISPLAY SCREEN(11,15).
282 DISPLAY ' 1 : ENTER ENTRIES '.
283 DISPLAY SCREEN(13,15).
284 DISPLAY ' 2 : ENTER ADJUST ENTRIES '.
285 DISPLAY SCREEN(15,15).
286 DISPLAY ' 3 : ACCOUNT INFORMATION MAINTAIN'.
287 DISPLAY 'SCREEN(17,15).
288 DISPLAY '================================='.  
289 ;
290 DESCRIPTION;
291 This screen shows the three types for account maintain.
292 ;
293 CALLED-BY take-type-of-acc-maintain;
294 DEFINE ROUTINE take-module-type-process;
295 # Last changed - May 26, 1984 15:41:45
296 SYNONYMS ARE t-m-t-p;
297 ALGORITHM;
298
299 PERFORM select-module-type-screen.
300 MOVE 'n' to value-error.
301 REPEAT
302 ACCEPT type-of-process
303 IF type-of-process length <> 1
304 THEN display 'accept value not correct'
305 MOVE 'y' to value-error
306 ELSE IF (type-of-process>'*') and
307 (type-of-process>'1') and
308 (type-of-process>'2') and
309 (type-of-process>'3')
310 THEN MOVE 'y' to value-error
311 ELSE display 'process type unproper'.
312 UNTIL value-error = 'n'.
313 ;
314 DESCRIPTION;
315 This is a interface routine for the
take-module-type module.
316 CALLS select-module-type-screen;
317 KEYWORD 'inter-routine';
318 LOCAL-DATA IS value-error;
319 PARAMETER type-of-module PASSED-BY
320 result;
321 INTERFACE ROUTINE FOR take-module-type;
322 DEFINE ROUTINE take-type-of-acc-maintain;
327 # Last changed - May 26, 1984 15:42:46
328 SYNONYMS ARE t-t-o-a-m;
329 ALGORITHM;
330
331 PERFORM take-acc-maintain-type-screen.
332 MOVE 'n' to value-error.
333 REPEAT
334   ACCEPT acc-maintain-type.
335   IF acc-maintain-type length <> 1
336     THEN MOVE 'y' to value-error.
337     display 'accept value incorrect'.
338   ELSE IF (account-maintain-typeO'1') and
339       (account-maintain-typeO'2') and
340       (account-maintain-typeO'3')
341     THEN MOVE 'y' to value-error
342       display 'process type improper'.
343   UNTIL value-error='n'.
344 ;
346 DESCRIPTION;
347 This routine decides the type of account
348 maintain type.;
349 CALLS take-acc-maintain-type-screen;
350 KEYWORD routin-maintain';
351 LOCAL-DATA IS value-error;
352 PARAMETER account-maintain-type
353 PASSED-BY result;
354 CALLED-BY account-maintain-process;
355 ROUTINE IN account-maintain;
355 lines printed. 107 statements printed.
DEFINE ROUTINE add-g-l-record;
SYNONYMS ARE a-g-l-r;
ALGORITHM;

PERFORM add-g-l-screen.
REPEAT
  MOVE 'n' to acc-number-error.
  MOVE 'y' to try-more.
  ACCEPT acc-number.
  IF acc-number length <> 6 THEN
    acc-number-error = 'y' ELSE
    IF 500000 < acc-number < 100000 THEN
      acc-number-error = 'y'
    ELSE READ (general-ledger-file)
    END-IF.
    IF general-ledger-record exist THEN
      acc-number-error = 'y'.
    END-IF.
  END-IF.
  UNTIL acc-number-error = 'n' OR try-more = 'n'.
REPEAT
  MOVE 'N' to account-name-error.
  ACCEPT acc-name.
  IF acc-name length < 30 THEN
    MOVE acc-name to G-acc-name
  ELSE acc-name-error = 'y'.
  END-IF.
  UNTIL acc-name-error = 'n'.
REPEAT
  MOVE 'n' to acc-type-error.
  ACCEPT acc-type.
  IF acc-type length <> 1 THEN
    MOVE 'Y' to acc-type-error
  ELSE IF (account-type <> '0') AND (account-type <> '1')
    AND (account-type <> '2')
    AND (account-type <> '3')
    THEN acc-type-error = 'y'
  ELSE MOVE acc-type to G-acc-type.
  END-IF.
  UNTIL acc-type-error = 'n'.
WRITE (general-ledger-record),
; DESCRIPTION;
This routine add a new general ledger account to the general ledger file;
CALLS add-g-l-screen;
KEYWORD 'routine-maintain';
KNOWS-OF G-acc-number,
G-acc-type,
G-acc-name;
LOCAL-DATA IS acc-number-error;
LOCAL-DATA IS acc-type-error;
LOCAL-DATA IS acc-name;
LOCAL-DATA IS acc-number;
LOCAL-DATA IS acc-type;
CALLED-BY maintain-acc-info;
DEFINE ROUTINE add-g-l-screen;

# Last changed - May 26, 1984 15:54:08
SYNONYMS ARE a-g-l-s;

ALGORITHM;

DISPLAY SCREEN(2,27).
DISPLAY 'ADD GENERAL LEDGER RECORD'.
DISPLAY SCREEN(4,10).
DISPLAY '+++++++++++++++++++++++++++++++++++++++++++++++'.
DISPLAY SCREEN(6,10).
DISPLAY 'ACCOUNT NUMBER ______'.
DISPLAY SCREEN(10,10).
DISPLAY 'ACCOUNT NAME ________________________________'.
DISPLAY SCREEN(12,10).
DISPLAY 'ACCOUNT TYPE '.
DISPLAY '+++++++++++++++++++++++++++++++++++++++++++++++++++++'.

DESCRIPTION;
This routine shows the screen for adding general ledger record.

CALLED-BY add-g-l-record;

DEFINE ROUTINE delete-g-l-record;

# Last changed - May 26, 1984 15:42:46
SYNONYMS ARE d-g-l-r;

ALGORITHM;

PERFORM delete-g-l-screen.
PERFORM get-valid-acc-number.
READ (general-ledger-file).
DISPLAY general-ledger-record content,
ACCEPT delete-confirm,
IF delete-confirm = 'y' THEN
DELETE (general-ledger-record).

DESCRIPTION;
This routine deletes the existed general ledger account from the general ledger file.

CALLS delete-g-l-screen;
KEYWORD 'routine-maintain';
LOCAL-DATA IS delete-confirm;
UTILIZES get-valid-acc-number;
CALLED-BY maintain-acc-info;

DEFINE ROUTINE delete-g-l-screen;

# Last changed - May 26, 1984 15:54:08
SYNONYMS ARE d-g-l-s;

ALGORITHM;
107 DISPLAY SCREEN(2,27).
108 DISPLAY 'DELETE GENERAL LEDGER RECORD'.
109 DISPLAY SCREEN(4,10).
110 DISPLAY 'ENTER ACCOUNT NUMBER ARE GOING TO BE DELETE'
111 DISPLAY 'ENTER HERE _____'.
112 DISPLAY SCREEN(6,10).
113 DISPLAY '========================================='. 
114 DISPLAY SCREEN(7,10).
115 DISPLAY 'ACCOUNT NUMBER _____'.
116 DISPLAY SCREEN(9,10).
117 DISPLAY 'ACCOUNT NAME ____________________________'.
118 DISPLAY SCREEN(12,10).
119 DISPLAY 'ACCOUNT TYPE '.
120 DISPLAY SCREEN(14,10).
121 DISPLAY 'ACCOUNT CURRENT TOTAL _______'.
122 DISPLAY '++++++++++++++++++++++++++++++'.
123 ;
124 DESCRIPTION;
125 This routine shows the screen for deleting the general ledger file.
126 ;
127 CALLED-BY delete-g-l-record;
128
129 DEFINE ROUTINE maintain-acc-info;
130 # Last changed - May 26, 1984 15:42:46
131 SYNONYMS ARE m-a-i;
132 ALGORITHM;
133
134 OPEN I/O general-ledger-file.
135 REPEAT
136     PERFORM select-acc-info-maintain.
137     MOVE 'y' to maintain-more.
138     CASE maintain-acc-info-type:
139         '1': PERFORM add-g-l-record
140         '2': PERFORM delete-g-l-record
141         '3': PERFORM modify-g-l-record
142     ACCEPT maintain-more.
143     UNTIL maintain-more='n'
144     OR maintain-acc-info-maintain='*'.
145 CLOSE general-ledger-file.
146 ;
147 DESCRIPTION;
148 This process maintains the general-ledger-file. It adds, deletes and modifies the account in the general-ledger-file.
149 CALLS select-acc-info-maintain,
150 add-g-l-record,
151 delete-g-l-record,
152 modify-g-l-record;
153 KEYWORD 'routine-maintain';
154 LOCAL-DATA IS maintain-more;
155 LOCAL-DATA IS maintain-acc-info-type;
156 CALLED-BY account-maintain-process;
157 ROUTINE IN account-maintain;
158
159 DEFINE ROUTINE modify-g-l-record;
SYNONYMS ARE m-g-l-r;

ALGORITHM;

PERFORM modify-g-l-screen.

PERFORM get-valid-acc-number.

IF try-again <> 'n' THEN READ display general-ledger-record content

REPEAT

ACCEPT modify-type,
CASE modify-type;
'1': ACCEPT acc-name
IF acc-name length < 30
THEN MOVE acc-name to G-acc-name.

'2': ACCEPT acc-type
IF (acc-type = '0')
OR (acc-type = '1')
OR (acc-type = '2')
OR (acc-type = '3')
THEN MOVE acc-type to G-acc-type.

UNTIL modify-type = '*'.

REWRITE (general-ledger-file).

DESCRIPTION;

This routine modify the exist general ledger account from the general ledger file.;

CALLS modify-g-l-screen;

KEYWORD 'routine-maintain';

LOCAL-DATA IS modify-type;

LOCAL-DATA IS try-again;

LOCAL-DATA IS acc-name;

LOCAL-DATA IS acc-type;

UTILIZES get-valid-acc-number;

CALLED-BY maintain-acc-info;

DEFINE ROUTINE modify-g-l-screen;

# Last changed - May 26, 1984 15:54:08
SYNONYMS ARE m-g-l-s;

ALGORITHM;

DISPLAY SCREEN(2,27).

DISPLAY 'MODIFY GENERAL LEDGER RECORD'.

DISPLAY SCREEN(4,10) .

DISPLAY ' ENTER ACCOUNT NUMBER ARE GOING TO BE MODIFY'

DISPLAY ' ENTER HERE _____'.

DISPLAY SCREEN(6,10)

DISPLAY '================================================'.

DISPLAY SCREEN(7,32).

DISPLAY 'ORIGINAL RECORD DATA'.

DISPLAY SCREEN(8,10) .

DISPLAY 'ACCOUNT NUMBER _____'.

DISPLAY SCREEN(9,10) .

DISPLAY 'ACCOUNT NAME ____________________________'.

DISPLAY SCREEN(10,10) .

DISPLAY 'ACCOUNT TYPE '.
Formatted Statements

217 DISPLAY SCREEN(11,10) .
218 DISPLAY 'ACCOUNT CURRENT TOTAL ____________ '
219 DISPLAY '+++++++++++++++++++++++++++++++++++++++'.
220 DISPLAY SCREEN(14,10)
221 DISPLAY 'ENTERED NEW VALUE FOR EACH FIELD'.
222 DISPLAY SCREEN(16,10) .
223 DISPLAY 'ACCOUNT NAME ________________________ '
224 DISPLAY SCREEN(18,10) .
225 DISPLAY 'ACCOUNT TYPE _ '.
226 ;
227 DESCRIPTION;
228 This routine shows the screen for modifying
229 general ledger file.
230 ;
231 CALLED-BY modify-g-l-record;
232 ;
233 DEFINE ROUTINE select-acc-info-maintain;
234 # Last changed - May 26, 1984 15:42:46
235 SYNONYMS ARE s-a-i-m;
236 ALGORITHM;
237
238 PERFORM select-info-maintain-screen.
239 REPEAT
240 MOVE 'n' to value-error.
241 ACCEPT maintain-acc-info-type.
242 IF maintain-acc-info-type length <> 1
243 THEN MOVE 'y' to value-error
244 ELSE IF (maintain-acc-info-type <>'*') and
245 (maintain-acc-info-type <>'1') and
246 (maintain-acc-info-type <>'2') and
247 (maintain-acc-info-type <>'3')
248 THEN MOVE 'y' to value-error
249 display 'maintain type improper'.
250 UNTIL value-error='n'.
251 ;
252 DESCRIPTION;
253 This routine decides the type of account information
254 maintain type.;
255 CALLS select-info-maintain-screen;
256 KEYWORD 'routine-maintain';
257 LOCAL-DATA IS value-error;
258 PARAMETER maintain-acc-info-type
259 PASSED-BY result;
260 CALLED-BY maintain-acc-info;
261
262 DEFINE ROUTINE select-info-maintain-screen;
263 # Last changed - May 26, 1984 15:54:08
264 SYNONYMS ARE s-i-m-s;
265 ALGORITHM;
266
267 DISPLAY SCREEN(2,26) .
268 DISPLAY 'ACCOUNT INFORMATION MAINTAIN'.
269 DISPLAY SCREEN(4,20) .
270 DISPLAY 'SELECT ONE OF FOLLOWING TYPE TO PROCESS '.
271 DISPLAY SCREEN(5,32) .
272 DISPLAY 'ENTER SELECTION _'.
273 DISPLAY SCREEN(7,10) .
274 DISPLAY '========================================'.
275 DISPLAY SCREEN(9,10) .
276 DISPLAY ' * : FOR EXIT THE ACCOUNT INFORMATION MAINTAIN.'
277 DISPLAY SCREEN(11,10) .
278 DISPLAY ' 1 : ADD GENERAL LEDGER ACCOUNT'.
279 DISPLAY SCREEN(13,10) .
280 DISPLAY ' 2 : DELETE GENERAL LEDGER ACCOUNT'.
281 DISPLAY SCREEN(15,10) .
282 DISPLAY ' 3 : MODIFY GENERAL LEDGER ACCOUNT'.
283 DISPLAY SCREEN(17,10) .
284 DISPLAY '========================================'.
285 ;
286 DESCRIPTION;
287 This routine shows the types of
288 account-information-maintain.;
289 CALLED-BY select-acc-info-maintain;
289 lines printed. 73 statements printed.
SELECT-TYPE-OF-ACC-UPDATE

UPDATE -G-L-FILE

CLEAN-OLD -ENTRIES

CLEAN-REVENUES-EXPENSE

END-OF -YEAR-UPDATE

`ACC-UPDATE-TYPE`

PARAMETER REFERENCE
DEFINE ROUTINE clean-old-entries;

# Last changed - May 26, 1984 15:48:21
SYNONYMS ARE c-o-e;

ALGORITHM;

OPEN entered-account-file,
pre-entered-account-file.

REPEAT
READ (pre-entered-account-file)
UNTIL eof(pre-entered-account-file).
MOVE 'n' to clean-old-entry-error.
WHILE not eof(entered-account-file)
BEGIN
READ (entered-account-file)
IF E-update-indicator = 'n' THEN
   display 'entered account record not updated'
   MOVE 'y' to clean-old-entry-error
   ELSE MOVE entered-account-record to
   pre-entered-account-record
   END. {while}
WRITE (pre-entered-account-file)
IF clean-old-entry-error THEN
   close entered-account-file,
   pre-entered-account-file
ELSE close entered-account-file
open input entered-account-file
read entered-account-file.
write pre-entered-account-file from
entered-account-record.
close entered-account-file
close pre-entered-account-file.

DESCRIPTION;
This process cleans the entered-account-file
after the entered accounts are posted to the
general-ledger-file. It appends the
entered-account to the previous-entered-
account-file to keep the audit trail.;

KEYWORD
   'update-routine';
KNOWS-OF
   E-update-indicator;
LOCAL-DATA IS
   clean-old-entry-error;
CALLED-BY
   account-update-process;
ROUTINE IN
   account-update;

DEFINE ROUTINE clean-revenues-expense;

# Last changed - May 26, 1984 15:48:21
SYNONYMS ARE c-r-e;

ALGORITHM;

OPEN general-ledger-file.
52 MOVE 0 to total-income.
53 WHILE not eof(general-ledger-file)
54 BEGIN
55 READ (general-ledger-file)
56 IF G-acc-number => 300000
57 THEN ADD G-acc-current-total to total-income
58 MOVE 0 to G-acc-current-total
59 END.{while}
60 IF total-income <> 0
61 THEN
62 READ (general-ledger-file) which is
63 account <retain earnings>
64 ADD total-income to G-acc-current-total
65 REWRITE (general-ledger-file).
66 CLOSE general-ledger-file.
67 ;
68 DESCRIPTION;
69 This process cleans all the revenues and expense
70 accounts in the general-ledger-file at the end
71 of the period according to the request.;
72 KEYWORD 'update-routine';
73 KNOWS-OF G-acc-number,
74 G-acc-current-total;
75 LOCAL-DATA IS total-income;
76 CALLED-BY account-update-process;
77 ROUTINE IN account-update;
78
79 DEFINE ROUTINE end-of-year-update;
80 # Last changed - May 26, 1984 15:48:21
81 SYNONYMS ARE e-o-y-u;
82 ALGORITHM;
83
84 OPEN entered-account-file,
85 pre-entered-account-file,
86 general-ledger-file.
87 MOVE 'n' end-of-year-update-error.
88 WHILE not eof(entered-account-file) OR
89 end-of-year-update-error 'y'
90 BEGIN
91 READ (entered-account-file)
92 IF E-update-indicator = 'n' THEN
93 MOVE 'Y' to end-of-year-update-error.
94 END.{while}
95 IF end-of-year-update-error = 'n' THEN
96 clean entered-account-file.
97 WHILE not eof(general-ledger-file) OR
98 end-of-year-update-error 'y'
99 BEGIN
100 READ (general-ledger-file)
101 IF G-acc-number < 300000
102 THEN MOVE G-acc-current-total to G-acc-pre-total
103 ELSE IF G-acc-current-total <> 0
104 THEN MOVE 'y' TO end-of-year-update-error
105 display 'revenues and expense not clean yet'
106 END.{while}
107 IF end-of-year-update-error = 'n' THEN
108 clean pre-entered-acc-file,
109 CLOSE entered-account-file,
110 pre-entered-account-file,
111 general-ledger-file.
112 ;
113 DESCRIPTION;
114 This routine close all the assets and liability
115 accounts in the general-ledger-file at the end of
116 the year.;
117 KEYWORD 'update-routine';
118 KNOWS-OF E-update-indicator,
119 G-acc-number,
120 G-acc-current-total,
121 G-acc-pre-total;
122 LOCAL-DATA IS end-of-year-update-error;
123 CALLED-BY account-update-process;
124 ROUTINE IN account-update;
125
126 DEFINE ROUTINE general-ledger-process;
127 # Last changed - May 26, 1984 16:03:08
128 SYNONYMS ARE g-l-p;
129 ALGORITHM;
130
131 REPEAT
132 PERFORM take-module-type-process.
133 CASE type-of-module;
134 '1': PERFORM account-maintain-process
135 '2': PERFORM account-update-process
136 '3': PERFORM report-generator-process.
137 UNTIL type-of-module = '*'.
138 ;
139 DESCRIPTION;
140 This is a interface routine of general-ledger-system.
141 ;
142 CALLS account-maintain-process,
143 account-update-process,
144 report-generator-process;
145 KEYWORD 'sub-sys';
146 LOCAL-DATA IS type-of-module;
147 MAIN ROUTINE FOR general-ledger-system;
148
149 DEFINE ROUTINE select-acc-update-screen;
150 # Last changed - May 26, 1984 15:54:08
151 SYNONYMS ARE s-a-u-s;
152 ALGORITHM;
153
154 DISPLAY SCREEN(2,33).
155 DISPLAY 'ACCOUNT UPDATE'.
156 DISPLAY SCREEN(4,20).
157 DISPLAY 'SELECT ONE OF FOLLOWING TYPE TO PROCESS '.
158 DISPLAY SCREEN(5,32).
159 DISPLAY 'ENTER SELECTION _'.
160 DISPLAY SCREEN(7,17).
161 DISPLAY '================================'. 
DISPLAY SCREEN(9,17).
163 DISPLAY ' * : FOR EXIT THE ACCOUNT UPDATE'.
164 DISPLAY SCREEN(11,17).
165 DISPLAY ' 1 : UPDATE GENERAL LEDGER FILE'.
166 DISPLAY SCREEN(13,17).
167 DISPLAY ' 2 : CLEAN REVENUES AND EXPENSE'.
168 DISPLAY SCREEN(15,17).
169 DISPLAY ' 3 : CLEAN OLD ENTRIES'.
170 DISPLAY SCREEN(17,17).
171 DISPLAY ' 4 : END OF YEAR PROCESS'.
172 DISPLAY SCREEN(19,17).
173 DISPLAY '================================'.
174 ;
175 DESCRIPTION;
176 This routine shows the type of account update process.
177 ;
178 CALLED-BY select-type-acc-update;
179 ;
180 DEFINE ROUTINE select-type-acc-update;
181 # Last changed - May 26, 1984 15:48:21
182 SYNONYMS ARE s-t-a-u;
183 ALGORITHM;
184
185 PERFORM select-acc-update-screen.
186 MOVE 'n' to value-error.
187 REPEAT
188    ACCEPT acc-update-type.
189    IF acc-update-type length <> 1 THEN
190       MOVE 'y' to value-error
191       display 'accept value incorrect'
192    ELSE IF (acc-update-type <> '*') and
193         (acc-update-type <> '1') and
194         (acc-update-type <> '2') and
195         (acc-update-type <> '3') and
196         (acc-update-type <> '4')
197       THEN MOVE 'y' to value-error
198       display 'update type improper'.
199    Until value-error='n'.
200 ;
201 DESCRIPTION;
202 This routine decides the type of account update.;
203 CALLS select-acc-update-screen;
204 KEYWORD 'update-routine';
205 LOCAL-DATA IS value-error;
206 PARAMETER acc-update-type PASSED-BY
207 result;
208 CALLED-BY account-update-process;
209 ROUTINE IN account-update;
210
211 DEFINE ROUTINE update-G-L-F;
212 # Last changed - May 26, 1984 15:48:21
213 SYNONYMS ARE u-g-l-f;
214 ALGORITHM;
215
217 WHILE not enf(entered-acc-file)
218 BEGIN
219 READ (entered-acc-file).
220 IF E-update-indicator = 'y' THEN
221 
222 "display entered-acc-record"
223 "display 'this record updated already'
224 ELSE READ (general-ledger-file) with same E-acc-number
225 IF E-acc-number not in general-ledger-file
226 THEN display 'this account number not in
227 general-ledger-file'
228 ELSE ADD E-acc-value to G-acc-current-total
229 MOVE 'y' to E-update-indicator
230 REWRITE (general-ledger-file)
231 END.{while}
232 REWRITE (entered-acc-file)
233 }
235 DESCRIPTION;
236 "If the trial balance shows that the account is
237 balance, the entered account entries in entered
238 account file will be posted to general ledger file.
239 "KEYWORD 'update-routine';
240 "KNOWS-OF E-update-indicator,
241 E-acc-number,
242 G-acc-number,
243 E-acc-value,
244 G-acc-current-total;
245 "CALLED-BY account-update-process;
246 "ROUTINE IN account-update;"
1 DEFINE LIBRARY-ROUTINE
2     print-company-title;
3 # Last changed - May 26, 1984 15:50:32
4 SYNONYMS ARE p-c-t;
5 ALGORITHM;
6
7 2 blank lines.
8 DISPLAY 29 spaces,'************** COMPANY'.
9 ;
10 DESCRIPTION;
11 This routine prints the company title .;
12 KEYWORD 'report-routine';
13 COLLECTED IN general-ledger-library;
14 UTILIZED BY print-balance-sheet,
15     print-chart-of-account,
16     print-income-statement,
17     print-ledger-sheet,
18     print-list-of-account-entered,
19     print-trial-balance;
20
21 DEFINE ROUTINE print-audit-trail;
22 # Last changed - May 26, 1984 15:50:32
23 SYNONYMS ARE p-a-t;
24 ALGORITHM;
25
26 MOVE 0 TO repeat-counter.
27 MOVE 0 to total-page-number.
28 REPEAT
29 ACCEPT begin-date, end-date.
30 OPEN pre-entered-acc-file.
31 MOVE 1 to page-number.
32 MOVE 0 TO line-number.
33 IF repeat-counter = 1 THEN
34     PERFORM print-audit-trail-title.
35     ADD 7 to line-number.
36 REPEAT
37     READ (pre-entered-acc-file).
38     UNTIL P-date-of-entry = begin-date.
39 REPEAT
40     PERFORM print-audit-trail-heading.
41     ADD 10 line-number.
42     WHILE (line-number <> 50) or
43         not(eof(pre-entered-acc-file))
44     BEGIN
45         IF repeat-counter = 1 THEN
46             PERFORM print-audit-trail-line.
47             ADD 1 to line-number.
48         END.
49     READ (pre-entered-acc-file).
50 END.
51 MOVE 0 to line-number.
UNTIL eof(pre-entered-acc-file) or
P-date-of-entry > end-date.
CLOSE pre-entered-acc-file.
ADD 1 to repeat-counter.
MOVE page-number to total-page-number.
UNTIL repeat-counter = 2.

; 

DESCRIPTION;
This routine prints the audit trail.;

CALLS print-audit-trail-title,
print-audit-trail-heading,
print-audit-trail-line;
KEYWORD 'report-routine';
LOCAL-DATA IS repeat-counter;
LOCAL-DATA IS begin-date;
LOCAL-DATA IS end-date;
LOCAL-DATA IS line-number;
LOCAL-DATA IS page-number;
LOCAL-DATA IS total-page-number;
CALLED-BY report-generator-process;
ROUTINE IN report-generator;

DEFINE ROUTINE print-audit-trail-heading;
# Last changed - May 26, 1984 15:50:32
SYNONYMS ARE p-a-t-h;
ALGORITHM;

1 blank line.
DISPLAY 54 spaces,'AUDIT TRAIL',
with no advancing.
DISPLAY 30 spaces,'PAGE',page-number,'OF',
total-page-number.
3 blank lines.
DISPLAY ' JOURNAL ENTRY ACCOUNT SOURCE'
ACCOUNT SOURCE'
with no advancing.
DISPLAY 'ENTRY NO, DATE NO. CODE'
with no advancing.
DISPLAY 12 spaces,'ACCOUNT NAME DR'
with no advancing.
DISPLAY 11 spaces,'CR'.
DISPLAY '-----------------------'
DISPLAY '-----------------------'
with no advancing.
DISPLAY '-----------------------'.
2 blank lines.

DESCRIPTION;
This routine prints the heading of audit trail.;

KEYWORD 'report-routine';
PARAMETER page-number PASSED-BY VALUE;
PARAMETER total-page-number PASSED-BY VALUE;
CALLED-BY print-audit-trail;
DEFINE ROUTINE print-audit-trail-line;

SYNONYMS ARE p-a-t-l;

DISPLAY 'P-entry-number', with no advancing.
DISPLAY 'P-date-of-entry', with no advancing.
DISPLAY 'P-account-number', with no advancing.
CASE P-source-code:
  '1': DISPLAY 'A/R' with no advancing.
  '2': DISPLAY 'A/P' with no advancing.
  '3': DISPLAY 'P/R' with no advancing.
  '4': DISPLAY 'ADJ' with no advancing.
DISPLAY 'P-account-name', with no advancing.
IF P-account-value < 0 THEN
  DISPLAY 'P-account-value', with no advancing.
ELSE DISPLAY 'P-account-value', with no advancing.

DESCRIPTION;
This routine prints a line of audit trail.

KEYWORD 'report-routine';
KNOWS-OF P-entry-number,
  P-date-of-entry,
  P-acc-number,
  P-source-code,
  P-acc-value,
  P-acc-name;
CALLED-BY print-audit-trail;

DEFINE ROUTINE print-audit-trail-title;
SYNONYMS ARE p-a-t-t;
ALGORITHM;

3 blanks line.
DISPLAY 50 spaces, '************** COMPANY'.
DISPLAY 104 spaces, 'TO'.
DISPLAY 104 spaces, end-date.

DESCRIPTION;
This routine prints the title of audit trail.

KEYWORD 'report-routine';
PARAMETER end-date PASSED-BY value;
PARAMETER begin-date PASSED-BY value;
CALLED-BY print-audit-trail;

DEFINE ROUTINE print-chart-of-acc-heading;
SYNONYMS ARE p-c-o-a-h;
ALGORITHM;
162
163 1 blank line.
164 DISPLAY 23 spaces,'CHART OF ACCOUNT AS OF ',
165     with no advancing.
166 DISPLAY sys-date,'
167     ',PAGE ',page-number,
168     'OF ',total-page-number.
169 4 blank lines.
170 DISPLAY 10 spaces,'ACCT NO
171     ACCT NAME',
172     with no advancing.
173 ;
174 DESCRIPTION;
175 This routine prints the heading of the chart of account.;
176 KEYWORD 'report-routine';
177 PARAMETER total-page-number PASSED-BY value;
178 PARAMETER page-number PASSED-BY value;
179 CALLED-BY print-chart-of-account;
180
181 DEFINE ROUTINE print-chart-of-acc-line;
182 # Last changed - May 26, 1984 15:50:32
183 SYNONYMS ARE p-c-o-a-1;
184 ALGORITHM;
185
186 DISPLAY 10 spaces,G-acc-number,'
187     G-acc-name,
188     with no advancing.
189 CASE G-acc-type :
190    '0' : DISPLAY 'REGULAR' with no advancing.
191    '1' : DISPLAY 'TITLE' with no advancing.
192    '2' : DISPLAY 'TOTAL' with no advancing.
193    '3' : DISPLAY 'HEADING' with no advancing.
194 IF G-acc-current-total < 0 THEN
195    DISPLAY abs(G-acc-current-total),'DR'.
196 ELSE DISPLAY G-acc-current-total,'CR'.
197 ;
198 DESCRIPTION;
199 This routine prints a line of the chart of account.;
200 KEYWORD 'report-routine';
201 KNOWS-OF G-acc-current-total,
202     G-acc-name,
203     G-acc-number,
204     G-acc-type;
205 CALLED-BY print-chart-of-account;
206
207 DEFINE ROUTINE print-chart-of-account;
208 # Last changed - May 26, 1984 15:50:32
209 SYNONYMS ARE p-c-o-a;
210 ALGORITHM;
211
212 OPEN general-ledger-file.
213 MOVE 1 to page-number.
214 MOVE 0 TO line-number.
215 PERFORM print-company-title.
ADD 5 to line-number.
WHILE not EOF(general-ledger-file)
BEGIN
  PERFORM print-chart-of-acc-heading.
  ADD 6 line-number.
  WHILE line-number <> 50 or
  not(eof(general-ledger-file))
  BEGIN
    READ (general-ledger-file)
    PERFORM print-chart-of-acc-line.
    ADD 1 to line-number.
  END.
  MOVE 0 to line-number.
END
ADD 1 to page-number.
CLOSE general-ledger-file.

DESCRIPTION;
This routine prints the chart of account.;

CALLS
  print-chart-of-acc-heading,
  print-chart-of-acc-line;

KEYWORD 'report-routine';
LOCAL-DATA IS
  line-number;
LOCAL-DATA IS
  page-number;
UTILIZES
  print-company-title;
CALLED-BY
  report-generator-process;
ROUTINE IN
  report-generator;

DEFINE ROUTINE
print-list-acc-entered-heading;
SYNONYMS ARE
  p-l-a-e-h;
ALGORITHM;

3 blank lines.
DISPLAY 102 spaces,'PAGE 'page-number,' OF ',
total-page-number.
2 blank lines.
DISPLAY 30 spaces,'JOURNAL ENTRY LIST AS OF',
sys-date.
DISPLAY 10 spaces,'JOURNAL ENTRY ACCOUNT SOURCE',
48 spaces,'AMOUNT'.
DISPLAY 10 spaces,'ENTRY NO. DATE NO. CODE',
with no advancing.
DISPLAY 11 spaces,'ACCOUNT NAME',14 spaces,'DR',
17 spaces,'CR'.
DISPLAY 10 spaces,'               ' with no advancing.
DISPLAY '_____________________________________________________________________'
with no advancing.
DISPLAY '_____________________________________________________________________'.

DESCRIPTION;
This routine prints the heading of list of account entered.;

KEYWORD 'report-routine';
272 PARAMETER total-page-number PASSED-BY value;
274 PARAMETER page-number PASSED-BY value;
276 CALLED-BY print-list-of-account-entered;
278 DEFINE ROUTINE print-list-acc-entered-line;
279 # Last changed - May 26, 1984 15:50:32
280 SYNONYMS ARE p-l-a-e;
282 ALGORITHM;
283 DISPLAY ' *E-entry-number,' ',
284 E-date-of-entry,' ',E-acc-number,
285 with no advancing.
286 CASE E-source-code :
287 '1' : DISPLAY 'A/R' with no advancing.
288 '2' : DISPLAY 'A/P' with no advancing.
289 '3' : DISPLAY 'P/R' with no advancing.
290 '4' : DISPLAY 'ADJ' with no advancing.
291 DISPLAY ' ',E-acc-name with no advance.
292 IF E-acc-value < 0 THEN
293 DISPLAY ' ', E-acc-value
294 ELSE
295 DISPLAY ' ', E-acc-value.
296 ;
297 DESCRIPTION;
298 This routine prints a line of list of account entered.;
299 KEYWORD 'report-routine';
300 CALLED-BY print-list-of-account-entered;
301 ALGORITHM;
302 DEFINE ROUTINE print-list-of-account-entered;
303 # Last changed - May 26, 1984 15:50:32
304 SYNONYMS ARE p-l-o-a-e;
305 ALGORITHM;
306 MOVE 0 TO repeat-counter.
307 MOVE 0 TO total-page-number.
308 REPEAT
309 ACCEPT sys-date.
310 OPEN entered-acc-file.
311 MOVE 1 to page-number.
312 MOVE 0 TO line-number.
313 IF repeat-counter = 1 THEN
314 PERFORM print-company-title.
315 ADD 4 to line-number.
316 REPEAT
317 READ (entered-acc-file).
318 UNTIL E-date-of-entry = sys-date.
319 REPEAT
320 Performs print-list-acc-entered-heading.
321 ADD 7 to line-number, to line-number.
322 WHILE (line-number <> 50) or
323 not(eof(entered-acc-file))
324 BEGIN
325 IF repeat-counter = 1 THEN
326 DISPLAY ' *E-entry-number,' ','
327 E-date-of-entry,' ',E-acc-number,
328 with no advancing.
329 CASE E-source-code :
330 '1' : DISPLAY 'A/R' with no advancing.
331 '2' : DISPLAY 'A/P' with no advancing.
332 '3' : DISPLAY 'P/R' with no advancing.
333 '4' : DISPLAY 'ADJ' with no advancing.
334 DISPLAY ' ',E-acc-name with no advance.
335 IF E-acc-value < 0 THEN
336 DISPLAY ' ', E-acc-value
337 ELSE
338 DISPLAY ' ', E-acc-value.
339 ;
340 DESCRIPTION;
341 This routine prints a line of list of account entered.;
342 KEYWORD 'report-routine';
343 CALLED-BY print-list-of-account-entered;
344 ALGORITHM;
345 MOVE 0 TO repeat-counter.
346 MOVE 0 TO total-page-number.
347 REPEAT
348 ACCEPT sys-date.
349 OPEN entered-acc-file.
350 MOVE 1 to page-number.
351 MOVE 0 TO line-number.
352 IF repeat-counter = 1 THEN
353 PERFORM print-company-title.
354 ADD 4 to line-number.
355 REPEAT
356 READ (entered-acc-file).
357 UNTIL E-date-of-entry = sys-date.
358 REPEAT
359 PERFORM print-list-acc-entered-heading.
360 ADD 7 to line-number, to line-number.
361 WHILE (line-number <> 50) or
362 not(eof(entered-acc-file))
363 BEGIN
364 IF repeat-counter = 1 THEN
365 DISPLAY ' *E-entry-number,' ','
366 E-date-of-entry,' ',E-acc-number,
PERFORM print-list-acc-entered-line.
ADD 1 to line-number.
READ (entered-acc-file).
END.
MOVE 0 to line-number.
ADD 1 to page-number.
UNTIL eof(entered-acc-file) or E-date-of-entry <> sys-date.
CLOSE entered-acc-file.
ADD 1 to repeat-counter.
MOVE page-number to total-page-number.
UNTIL repeat-counter = 2.

This routine prints the list of account entered.
This routine accepts report type.

CALLS print-list-acc-entered-heading,
print-list-acc-entered-line;

KEYWORD 'report-routine';
KNOWS-OF sys-date;
LOCAL-DATA IS page-number;
LOCAL-DATA IS line-number;
LOCAL-DATA IS total-page-number;
LOCAL-DATA IS repeat-counter;
UTILIZES print-company-title;
CALLED-BY report-generator-process;
ROUTINE IN report-generator;

DEFINE ROUTINE select-report-type;
# Last changed - May 26, 1984 15:50:32
SYNONYMS ARE s-r-t;
ALGORITHM;

PERFORM select-report-type-screen.
MOVE 'n' to value-error.
REPEAT
ACCEPT report-type.
IF report-type length <> 1
THEN MOVE 'y' to value-error
display 'value length error'
ELSE
IF (report-type<>'*') and
(report-type<>'1') and
(report-type<>'2') and
(report-type<>'3') and
(report-type<>'4') and
(report-type<>'5') and
(report-type<>'6') and
(report-type<>'7')
THEN MOVE 'y' to value-error
display 'report type improper'.
UNTIL value-error='n'.

This routine accepts report type.

CALLS select-type-of-report-screen;
382 KEYWORD 'report-routine';
383 LOCAL-DATA IS value-error;
384 PARAMETER report-type PASSED-BY result;
385 CALLED-BY report-generator-process;
386 ROUTINE IN report-generator;
387
388
389 DEFINE ROUTINE select-type-of-report-screen;
390 # Last changed - May 26, 1984 15:54:08
391 SYNONYMS ARE s-t-o-r-s;
392 ALGORITHM;
393
394 DISPLAY SCREEN(2,29) .
395 DISPLAY 'SELECT TYPE OF REPORT'.
396 DISPLAY SCREEN(4,20) .
397 DISPLAY 'SELECT ONE OF FOLLOWING TYPE TO PROCESS '.
398 DISPLAY SCREEN(5,32) .
399 DISPLAY 'ENTER SELECTION _'.
400 DISPLAY SCREEN(7,10) .
401 DISPLAY '============================================'.
402 DISPLAY SCREEN(9,10) .
403 DISPLAY ' * : FOR EXIT THE REPORT GENERATOR PROCESS .'
404 DISPLAY SCREEN(10,10) .
405 DISPLAY ' 1 : CHART OF ACCOUNT'.
406 DISPLAY SCREEN(11,10) .
407 DISPLAY ' 2 : LIST OF ACCOUNT ENTERED'.
408 DISPLAY SCREEN(12,10) .
409 DISPLAY ' 3 : AUDIT TRAIL '.
410 DISPLAY SCREEN(13,10) .
411 DISPLAY ' 4 : LEDGER SHEET '.
412 DISPLAY SCREEN(14,10) .
413 DISPLAY ' 5 : TRIAL BALANCE'.
414 DISPLAY SCREEN(15,10) .
415 DISPLAY ' 6 : INCOME STATEMENT'.
416 DISPLAY SCREEN(16,10) .
417 DISPLAY ' 7 : BALANCE SHEET'.
418 DISPLAY '============================================'.
419 ;
420 DESCRIPTION;
421 This routine shows the screen for report selection.
422 ;
423 CALLED-BY select-report-type;
424
425 lines printed. 113 statements printed.
1
2 DEFINE LIBRARY-ROUTINE
3       print-company-title;
4 # Last changed - May 26, 1984 15:50:32
5 SYNONYMS ARE
6 ALGORITHM;
7
8 2 blank lines.
9 DISPLAY 29 spaces,'************** COMPANY'.
10 ;
11 DESCRIPTION;
12 This routine prints the company title.;
13 KEYWORD 'report-routine';
14 COLLECTED IN general-ledger-library;
15 UTILIZED BY print-balance-sheet,
16 print-chart-of-account,
17 print-income-statement,
18 print-ledger-sheet,
19 print-list-of-account-entered,
20 print-trial-balance;
21
22 DEFINE ROUTINE caculate-current-acc-value;
23 # Last changed - May 26, 1984 15:50:32
24 SYNONYMS ARE c-c-a-v;
25 ALGORITHM;
26
27 OPEN entered-account-file.
28 MOVE G-acc-current-total to amount-value.
29 WHILE not eof(entered-account-file)
30 BEGIN
31 READ (entered-account-file)
32 IF E-acc-number = G-acc-number THEN
33 ADD E-acc-value to amount-value .
34 END. {while}
35 CLOSE enntered-account-file.
36 ;
37 DESCRIPTION;
38This routine caculate the current value for each account;
39 KEYWORD 'report-routine';
40 PARAMETER amount-value PASSED-BY
41 result;
42 CALLED-BY print-trial-balance;
43
44 DEFINE ROUTINE print-balance-sheet;
45 # Last changed - May 26, 1984 15:50:32
46 SYNONYMS ARE p-b-s;
47 ALGORITHM;
48
49 OPEN general-ledger-file.
50 MOVE 1 to page-number.
51 MOVE 0 TO line-number.
PERFORM print-company-title.
ADD 3 to line-number.
WHILE not EOF(general-ledger-file)
BEGIN
PERFORM print-balance-sheet-heading.
ADD 7 line-number.
WHILE line-number <> 50 or
not(eof(general-ledger-file))
BEGIN
READ (general-ledger-file),
IF G-acc-number < 300000 THEN
PERFORM print-balance-sheet-line
ADD 1 to line-number.
END.
MOVE 0 to line-number.
ADD 1 to page-number.
END.
CLOSE general-ledger-file.

This routine prints the balance sheet.

CALLS
print-balance-sheet-heading,
print-balance-sheet-line;

KEYWORD
'report-routine';

LOCAL-DATA IS
line-number;
page-number;

UTILIZES
print-company-title;

CALLED-BY
report-generator-process;

ROUTINE IN
report-generator;

DEFINE ROUTINE print-balance-sheet-heading;
SYNONYMS ARE
p-b-s-h;

KEYWORD
'report-routine';
KNOWS-OF
sys-date;
PARAMETER
total-page-number
PASSED-BY
value;
PARAMETER
page-number PASSED-BY
value;
CALLED-BY
print-balance-sheet;

DEFINE ROUTINE print-balance-sheet-line;
SYNONYMS ARE
p-b-s-1;
107 ALGORITHM;

108 DISPLAY 10 spaces,G-acc-name ,15 spaces,with no advancing.

110 CASE G-ACC-TYPE :
111 '0' : IF G-acc-current-total > 0 THEN
112 DISPLAY G-acc-current-total
113 ELSE DISPLAY '<',abs(G-acc-current-total),'>'.
114 FOR INDEX = index to 8 DO
115 ADD G-acc-current-total to total(INDEX).
116 '2' : IF total(index) > 0 THEN
117 DISPLAY 'total(index)
118 ELSE
119 DISPLAY '<',abs(total(index)),'>','.
120 ADD 1 to the index.
121 '3' : DECREASE index by 1.

122 ;

123 DESCRIPTION;
124 This routine prints a line of the balance sheet .;
125 KEYWORD 'report-routine';
126 CALLED-BY print-balance-sheet;

128 DEFINE ROUTINE print-income-statement;
129 # Last changed - May 26, 1984 15:50:32
130 SYNONYMS ARE p-i-s;
131 ALGORITHM;

132 OPEN general-ledger-file.
134 MOVE 1 to page-number.
135 MOVE 0 TO line-number.
137 PERFORM print-company-title.
138 WHILE not EOF(general-ledger-file)
139 BEGIN
140 PERFORM print-income-statement-heading.
141 ADD 7 line-number.
142 WHILE line-number <> 50 or
143 not(eof(general-ledger-file))
144 BEGIN
145 READ (general-ledger-file).
146 IF G-acc-number => 300000 THEN
147 PERFORM print-income-statement-line
148 END.
149 END.
150 MOVE 0 to line-number.
151 ADD 1 to page-number.
152 END.
153 CLOSE general-ledger-file.

155 DESCRIPTION;
156 This routine prints the income statement.;
157 CALLS print-income-statement-heading,
158 print-income-statement-line;
159 KEYWORD 'report-routine';
160 LOCAL-DATA IS line-number;
161 LOCAL-DATA IS page-number;
UTILIZES print-company-title;
CALLED-BY report-generator-process;
ROUTINE IN report-generator;

DEFINE ROUTINE print-income-statement-heading;
# Last changed - May 26, 1984 15:50:32
SYNONYMS ARE p-i-s-h;
ALGORITHM;

2 blank lines.
DISPLAY 39 spaces,'INCOME STATEMENT'.
DISPLAY 77 spaces,'PAGE ',page-number,' OF ',
total-page-number.
DISPLAY 30 spaces,'FOR THE PERIOD ENDING ',
sys-date.
2 blank lines.

DESCRIPTION;
This routine prints the heading of the income statement.;
KEYWORD 'report-routine';
PARAMETER total-page-number PASSED-BY value;
PARAMETER page-number PASSED-BY value;
CALLED-BY print-income-statement;

DEFINE ROUTINE print-income-statement-line;
# Last changed - May 26, 1984 15:50:32
SYNONYMS ARE p-i-s-l;
ALGORITHM;

DISPLAY 10 spaces,G-acc-name ,15 spaces,with no advancing.
CASE G-ACC-TYPE :
'0' : IF G-acc-current-total > 0 THEN
DISPLAY G-acc-current-total
ELSE DISPLAY '<',abs(G-acc-current-total),'>.'.
FOR INDEX = index to 8 DO
ADD G-acc-current-total to total(INDEX).
'2' : IF total(index) > 0 THEN
DISPLAY '
ELSE
DISPLAY '<',abs(total(index)),'>.'.
ADD 1 to the index.
'3' : DECREASE index by 1.

DESCRIPTION;
This routine prints a line of the income statement.;
KEYWORD 'report-routine';
KNOWS-OF G-acc-name,
G-acc-type,
G-acc-current-total;
CALLED-BY print-income-statement;

DEFINE ROUTINE print-ledger-sheet;
# Last changed - May 26, 1984 15:50:32
SYNONYMS ARE p-l-s;

ALGORITHM;

WHILE not eof(general-ledger-file)
BEGIN
  OPEN general-ledger-file.
  MOVE 1 to page-number.
  MOVE 0 TO line-number.
  READ (general-ledger-file).
  PERFORM print-company-title.
  ADD 4 to line-number.
  MOVE 0 to repeat-counter.
  MOVE 0 to total-page-number.
  REPEAT
    OPEN (entered-acc-file).
    MOVE 0 TO net-change.
    REPEAT
      MOVE 0 TO net-change.
      PERFORM print-ledger-sheet-heading.
      ADD 7 to line-number.
      REPEAT
        READ (entered-account-file).
        IF E-acc-number = G-acc-number THEN
          ADD 1 to line-number
          IF repeat-counter = 1 THEN
            ADD E-account-value TO net-change
            PERFORM print-ledger-sheet-line.
          END-IF.
        END-IF.
      END-REPEAT.
      ADD 1 to page-number.
      MOVE 0 to line-number.
      UNTIL eof(entered-account-file).
    END-REPEAT.
    MOVE 0 to line-number.
    CLOSE (entered-account-file).
    ADD 1 to repeat-counter.
  END-REPEAT.
  IF repeat-counter = 2 THEN
    DISPLAY 'NET CHANGE ' net-change
    DISPLAY 'ACCOUNT NAME'
    DISPLAY account-name
    DISPLAY 'OPEN BALANCE ' G-current-value
    DISPLAY 'CLOSING BALANCE '
    G-current-value + net-change.
    IF repeat-counter = 1 THEN MOVE page-number to total-page-number.
  END-IF.
END-REPEAT While.
CLOSE (general-ledger-file).

DESCRIPTION;
This routine prints the ledger sheet.;

CALLS print-ledger-sheet-heading,
print-ledger-sheet-line;

'report-routine';

LOCAL-DATA IS repeat-counter;
 LOCAL-DATA IS net-change;
 LOCAL-DATA IS line-number;
 LOCAL-DATA IS total-page-number;
 LOCAL-DATA IS page-number;
272 UTILIZES print-company-title;
273 CALLED-BY report-generator-process;
274 ROUTINE IN report-generator;
275
276 DEFINE ROUTINE print-ledger-sheet-heading;
277 # Last changed - May 26, 1984 15:50:32
278 SYNONYMS ARE p-l-s-h;
279 ALGORITHM;

280 2 blank lines.
281 DISPLAY ' ', sys-date.
282 DISPLAY ' with no advancing.
283 DISPLAY 'LEDGER SHEET ' with no advancing.
284 DISPLAY ' PAGE', page-number, ' OF ',
285 DISPLAY ' total-page-number.
286 2 blank lines.
287 DISPLAY ' ACCOUNT SOURCE JOURNAL'
288 DISPLAY ' with no advancing.
289 DISPLAY ' AMOUNT'.
290 DISPLAY ' NO. CODE ENTRY NO.'
291 DISPLAY ' with no advancing.
292 DISPLAY ' DATE DR CR'.
293 DISPLAY ' with no advancing.
294 DISPLAY '------ ------ ------'.
295 DISPLAY '------ ------------------'.
296 DISPLAY '------ ------------------'.
297 ;
298 DESCRIPTION;
299 This routine prints the heading of ledger sheet.;
300 KEYWORD 'report-routine';
301 PARAMETER page-number PASSED-BY value;
302 PARAMETER total-page-number PASSED-BY value;
303 CALLED-BY print-ledger-sheet;

307 DEFINE ROUTINE print-ledger-sheet-line;
308 # Last changed - May 26, 1984 15:50:32
309 SYNONYMS ARE p-l-s-l;
310 ALGORITHM;
311
312 DISPLAY 'E-account-number, with no advancing.
313 DISPLAY ' ', with no advancing.
314 CASE E-source-code :
315 '1' : DISPLAY 'A/R' with no advancing.
316 '2' : DISPLAY 'A/P' with no advancing.
317 '3' : DISPLAY 'P/R' with no advancing.
318 '4' : DISPLAY 'ADJ' with no advancing.
319 DISPLAY ' E-entry-number ', with no advancing.
320 DISPLAY ' E-date-of-entry ', with no advancing.
321 IF E-acc-value < 0 THEN
322 DISPLAY ' ', E-acc-value
323 ELSE
324 DISPLAY ' ', E-acc-value.
325 ;
326 DESCRIPTION;
This routine prints a line of ledger sheet.;

KEYWORD 'report-routine';

KNOWS-OF E-source-code,
E-date-of-entry,
E-acc-value,
E-entry-number,
E-acc-number,
E-acc-name;

CALLED-BY print-ledger-sheet;

DEFINE ROUTINE print-trial-balance;

SYNONYMS ARE p-t-b;

ALGORITHM;

MOVE 0 to repeat-counter.

REPEAT

MOVE 0 to total-CR,total-DR.
OPEN general-ledger-file.
MOVE 1 to page-number.
MOVE 0 TO line-number.
IF repeat-counter = 1 THEN
PERFORM print-company-title.
ADD 4 to line-number.
WHILE not EOF(general-ledger-file)
BEGIN
IF repeat-counter = 1 THEN
PERFORM print-trial-balance-heading.
ADD 6 line-number.
REPEAT
READ (general-ledger-file).
IF G-acc-type = 0 THEN
PERFORM caculate-current-acc-value.
IF repeat-counter = 1 THEN
IF amount < 0 THEN
ADD amount-value to total-DR
ELSE
ADD amount-value to total-CR.
IF amount-value <> 0 THEN
ADD 1 to line-number
IF repeat-counter = 1 THEN
PERFORM print-trial-balance-line.
UNTIL line-number = 50.
ADD 1 to page-number.
MOVE 0 to line-number.
END.
CLOSE general-ledger-file.
ADD 1 to repeat-counter.
MOVE page-number to total-page-number.
UNTIL repeat-counter = 2.
IF total-CR = total-DR THEN
DISPLAY ' ', 75 '='.
ELSE
DISPLAY 'TOTAL 'total-DR,' = ',total-CR.
ELSE
DISPLAY 'TOTAL 'total-DR,' <> 'total-CR.
This routine prints the trial balance;

```
CALLS print-trial-balance-heading,
       print-trial-balance-line,
       calculate-current-acc-value;

KEYWORD 'report-routine';
LOCAL-DATA IS page-number;
LOCAL-DATA IS line-number;
LOCAL-DATA IS total-page-number;
LOCAL-DATA IS repeat-counter;
LOCAL-DATA IS total-CR;
LOCAL-DATA IS total-DR;
LOCAL-DATA IS amount-value;
UTILIZES print-company-title;
CALLED-BY report-generator-process;
ROUTINE IN report-generator;
```

```
DEFINE ROUTINE print-trial-balance-heading;
# Last changed - May 26, 1984 15:50:32
SYNONYMS ARE p-t-b-h;
ALGORITHM;

DISPLAY 37 spaces,'TRIAL BALANCE SHEET',
         15 spaces,'PAGE ',page-number,' OF ',
         total-page-number.
2 blank lines.
DISPLAY 10 spaces,'ACCOUNT',12 spaces,'ACCOUNT',
         27 spaces,'AMOUNT'.
DISPLAY 12 spaces,'no.','17 spaces,'NAME',20 spaces,
        'DR CR'.
DISPLAY 10 spaces,'________',30 '_',' ',
        31 ' '.

; This routine prints the heading of the trial balance .;
KEYWORD 'report-routine';
PARAMETER total-page-number PASSED-BY
         value;
PARAMETER page-number PASSED-BY
         value;
CALLED-BY print-trial-balance-line;
```

```
IF amount-value > 0 THEN
   DISPLAY ' ',amount-value
ELSE
   DISPLAY ' ',amount-value.
;```
DESCRIPTION;
This routine prints a line of the trial balance .;
KEYWORD
'report-routine';
PARAMETER
amount-value PASSED-BY value;
CALLED-BY
print-trial-balance;

DEFINE ROUTINE print-trial-balance-title;
# Last changed - May 26, 1984 15:50:32
SYNONYMS ARE p-t-b-t;
ALGORITHM;

3 blank lines.
DISPLAY 36 spaces,'*************** COMPANY',
18 spaces,sys-date.
DISPLAY 37 spaces,'TRIAL BALANCE SHEET',
15 spaces,'PAGE ',page-number,' OF ',total-page-number.

DESCRIPTION;
This routine prints the title of the trial balance .;
KEYWORD
'report-routine';
PARAMETER
total-page-number
PASSED-BY value;
PARAMETER
page-number PASSED-BY value;

462 lines printed. 132 statements printed.
APPENDIX C

IDENTIFICATION DIVISION.

PROGRAM-ID.

SCREEN.

AUTHOR.

CHUNG-CHIEH WU.

DATE-WRITTEN.

FEB 22, 1984.

ENVIRONMENT DIVISION.

CONFIGURATION SECTION.

SOURCE-COMPUTER.

DECSYSTEM-20.

OBJECT-COMPUTER.

DECSYSTEM-20.

DATA DIVISION.

WORKING-STORAGE SECTION.

01 CHARACTER-VALUES

   USAGE IS DISPLAY-7.

   05 FILLER PIC X VALUE ' '.
   05 FILLER PIC X VALUE '1'.
   05 FILLER PIC X VALUE '2'.
   05 FILLER PIC X VALUE '3'.
   05 FILLER PIC X VALUE '4'.
   05 FILLER PIC X VALUE '5'.
   05 FILLER PIC X VALUE '6'.
   05 FILLER PIC X VALUE '7'.
   05 FILLER PIC X VALUE '8'.
   05 FILLER PIC X VALUE '9'.
   05 FILLER PIC X VALUE '('.
   05 FILLER PIC X VALUE ')'.
   05 FILLER PIC X VALUE '+'.
   05 FILLER PIC X VALUE '-'.
   05 FILLER PIC X VALUE '/'.
   05 FILLER PIC X VALUE '0'.
   05 FILLER PIC X VALUE '1'.
   05 FILLER PIC X VALUE '2'.
   05 FILLER PIC X VALUE '3'.
   05 FILLER PIC X VALUE '4'.
   05 FILLER PIC X VALUE '5'.
   05 FILLER PIC X VALUE '6'.
   05 FILLER PIC X VALUE '7'.
   05 FILLER PIC X VALUE '8'.
   05 FILLER PIC X VALUE '9'.
05 FILLER PIC X VALUE '7'.
05 FILLER PIC X VALUE '8'.
05 FILLER PIC X VALUE '9'.
05 FILLER PIC X VALUE ':'.
05 FILLER PIC X VALUE '}'.
05 FILLER PIC X VALUE '<'.
05 FILLER PIC X VALUE '='.
05 FILLER PIC X VALUE '>'.
05 FILLER PIC X VALUE '?'.
05 FILLER PIC X VALUE '@'.
05 FILLER PIC X VALUE 'A'.
05 FILLER PIC X VALUE 'B'.
05 FILLER PIC X VALUE 'C'.
05 FILLER PIC X VALUE 'D'.
05 FILLER PIC X VALUE 'E'.
05 FILLER PIC X VALUE 'F'.
05 FILLER PIC X VALUE 'G'.
05 FILLER PIC X VALUE 'H'.
05 FILLER PIC X VALUE 'I'.
05 FILLER PIC X VALUE 'J'.
05 FILLER PIC X VALUE 'K'.
05 FILLER PIC X VALUE 'L'.
05 FILLER PIC X VALUE 'M'.
05 FILLER PIC X VALUE 'N'.
05 FILLER PIC X VALUE 'O'.
05 FILLER PIC X VALUE 'P'.
05 FILLER PIC X VALUE 'Q'.
05 FILLER PIC X VALUE 'R'.
05 FILLER PIC X VALUE 'S'.
05 FILLER PIC X VALUE 'T'.
05 FILLER PIC X VALUE 'U'.
05 FILLER PIC X VALUE 'V'.
05 FILLER PIC X VALUE 'W'.
05 FILLER PIC X VALUE 'X'.
05 FILLER PIC X VALUE 'Y'.
05 FILLER PIC X VALUE 'Z'.
05 FILLER PIC X VALUE '['.
05 FILLER PIC X VALUE '\'.
05 FILLER PIC X VALUE ']'.
05 FILLER PIC X VALUE '^'.
05 FILLER PIC X VALUE '_'.
05 FILLER PIC X VALUE '`'.
05 FILLER PIC X VALUE 'a'.
05 FILLER PIC X VALUE 'b'.
05 FILLER PIC X VALUE 'c'.
05 FILLER PIC X VALUE 'd'.
05 FILLER PIC X VALUE 'e'.
05 FILLER PIC X VALUE 'f'.
05 FILLER PIC X VALUE 'g'.
05 FILLER PIC X VALUE 'h'.
05 FILLER PIC X VALUE 'i'.
05 FILLER PIC X VALUE 'j'.
05 FILLER PIC X VALUE 'k'.
05 FILLER PIC X VALUE 'l'.
05 FILLER PIC X VALUE 'm'.
05 FILLER PIC X VALUE 'n'.

APPENDIX C

05 FILLER PIC X VALUE 'o'.

01 SCREEN-TABLE-C REDEFINES CHARACTER-VALUES USAGE IS DISPLAY-7.

05 SCREEN-POSITION PIC X OCCURS 80 TIMES.

01 CONTRO-CHARACTER PIC XX VALUES '^y' USAGE IS DISPLAY-7.

77 COLUMN-NUMBER PIC 99.

77 ROW-NUMBER PIC 99.

LINKAGE SECTION.

01 SCREEN-TABLE.

05 SCREEN-ROWS OCCURS 24 TIMES.

10 SCREEN-LOCATION PIC X(4) USAGE DISPLAY-7 OCCURS 80 TIMES.

01 CLEAR-SCREEN PIC XX DISPLAY-7.

01 RING-BELL PIC X DISPLAY-7.

01 REVERSE-VEDIO PIC XX DISPLAY-7.

01 EXIT-REVERSE PIC XX DISPLAY-7.

PROCEDURE DIVISION, USING SCREEN-TABLE CLEAR-SCREEN RING-BELL REVERSE-VEDIO EXIT-REVERSE.

MAIN-PROGRAM.

PERFORM BUILD-SCREEN-TABLE.

PERFORM DEFINE-CLEAR-BELL.

EXIT PROGRAM.

BUILD-SCREEN-TABLE.

MOVE 1 TO ROW-NUMBER.

PERFORM BUILD-ROW UNTIL ROW-NUMBER IS GREATER THAN 24.

BUILD-ROW.

MOVE 1 TO COLUMN-NUMBER.

PERFORM BUILD-COLUMN UNTIL COLUMN-NUMBER IS GREATER THAN 80.

ADD 1 TO ROW-NUMBER.

BUILD-COLUMN.

STRING CONTRO-CHARACTER DELIMITED BY SIZE SCREEN-POSITION(ROW-NUMBER) DELIMITED BY SIZE
SCREEN-POSITION(COLUMN-NUMBER) DELIMITED BY SIZE INTO SCREEN-LOCATION(ROW-NUMBER , COLUMN-NUMBER).
ADD 1 TO COLUMN-NUMBER.

DEFINE-CLEAR-BELL.
MOVE '^[E' TO CLEAR-SCREEN.
MOVE '^[G' TO RING-BELL.
MOVE '^[P' TO REVERSE-VIDEO.
MOVE '^[q' TO EXIT-REVERSE.
IDENTIFICATION DIVISION.
PROGRAM-ID.
  MODULE
AUTHOR.
  CCW
DATE-WRITTEN.
  25-MAR-1984
ENVIRONMENT DIVISION.
INPUT-OUTPUT SECTION.
DATA DIVISION.
WORKING-STORAGE SECTION.
  01 TEST-VALUE-A.
    05 FRONT-A PIC X.
    05 BEHIND-A PIC X(9).
  01 MODULE-TYPE PIC X.
  01 SCREEN-TABLE.
    05 SCREEN-ROWS OCCURS 24 TIMES.
      10 SCREEN PIC X(4) USAGE IS DISPLAY-7 OCCURS 80 TIMES.
  01 VALUE-ERROR PIC X.
  01 REVERSE-VIDEO PIC XX DISPLAY-7.
  01 EXIT-REVERSE PIC XX DISPLAY-7.
  01 CLEAR-SCREEN PIC XX DISPLAY-7.
  01 RING-BELL PIC X DISPLAY-7.
  77 TAKE-TYPE PIC X VALUE 'N'.
    88 TAKE-TYPE-OK VALUE 'Y'.
PROCEDURE DIVISION.
MAIN.
  CALL 'SCREEN' USING SCREEN-TABLE CLEAR-SCREEN RING-BELL REVERSE-VIDEO EXIT-REVERSE.
  PERFORM GENERAL-LEDGER-PROCESS UNTIL
APPENDIX C

MODULE-TYPE EQUAL 'N'.

STOP RUN.

GENERAL-LEDGER-PROCESS.
    PERFORM SELECT-MODULE-TYPE-SCREEN.
    MOVE 'N' TO TAKE-TYPE.
    MOVE 'N' TO VALUE-ERROR.
    PERFORM TAKE-MODULE-TYPE-PROCESS UNTIL TAKE-TYPE-OK.
    IF MODULE-TYPE = '1'
        PERFORM ACCOUNT-MAINTAIN-PROCESS
    ELSE IF MODULE-TYPE = '2'
        PERFORM ACCOUNT-UPDATE-PROCESS
    ELSE IF MODULE-TYPE = '3'
        PERFORM REPORT-GENERATOR-PROCESS.
    DISPLAY CLEAR-SCREEN WITH NO ADVANCING.

SELECT-MODULE-TYPE-SCREEN.
    DISPLAY CLEAR-SCREEN WITH NO ADVANCING.
    DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
    DISPLAY SCREEN(1,32) WITH NO ADVANCING.
    DISPLAY 'SCREEN - 1.0'
        WITH NO ADVANCING.
    DISPLAY SCREEN(2,27) WITH NO ADVANCING.
    DISPLAY 'GENERAL LEDGER SYSTEM'
        WITH NO ADVANCING.
    DISPLAY EXIT-REVERSE WITH NO ADVANCING.
    DISPLAY SCREEN (4,20) WITH NO ADVANCING.
    DISPLAY
        'SELECT ONE OF THE FOLLOWING TYPE TO PROCESS'
            WITH NO ADVANCING.
    DISPLAY SCREEN (5,32) WITH NO ADVANCING.
    DISPLAY 'ENTER SELECTION: _' WITH NO ADVANCING.
        DISPLAY SCREEN(7,18) WITH NO ADVANCING.
    DISPLAY '========================================'
        WITH NO ADVANCING.
    DISPLAY SCREEN(9,18) WITH NO ADVANCING.
    DISPLAY ' * : FOR EXIT GENERAL LEDGER SYSTEM'
        WITH NO ADVANCING.
    DISPLAY SCREEN(11,18) WITH NO ADVANCING.
    DISPLAY ' 1 : FOR ACCOUNT MAINTAIN PROCESS'
        WITH NO ADVANCING.
    DISPLAY SCREEN(13,18) WITH NO ADVANCING.
    DISPLAY ' 2 : FOR ACCOUNT UPDATE PROCESS'
        WITH NO ADVANCING.
    DISPLAY SCREEN(15,18) WITH NO ADVANCING.
    DISPLAY ' 3 : FOR REPORT GENERATOR PROCESS'
        WITH NO ADVANCING.
    DISPLAY SCREEN(17,18) WITH NO ADVANCING.
    DISPLAY '========================================'
        WITH NO ADVANCING.

TAKE-MODULE-TYPE-PROCESS.
    MOVE SPACES TO TEST-VALUE-A.
    MOVE 'N' TO TAKE-TYPE.
IF VALUE-ERROR EQUAL 'Y'
   PERFORM VALUE-ERROR-HANDLE-1.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY SCREEN(5,49) WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A EQUAL '
   IF FRONT-A EQUAL '1' OR EQUAL '2' OR EQUAL '3' 
      OR EQUAL ' *
      MOVE 'Y' TO TAKE-TYPE
      MOVE FRONT-A TO MODULE-TYPE.
   IF TAKE-TYPE = 'N' MOVE 'Y' TO VALUE-ERROR.
DISPLAY SCREEN(22,7) WITH NO ADVANCING.
DISPLAY ' 
   WITH NO ADVANCING.
DISPLAY SCREEN(23,7) WITH NO ADVANCING.
DISPLAY ' 
   WITH NO ADVANCING.

VALUE-ERROR-HANDLE-1.
DISPLAY SCREEN(22,7) WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY 'ERROR MESSAGE::' WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY SCREEN(23,7) WITH NO ADVANCING.
DISPLAY 'INVALID RESPONSE . PLEASE REENTER' 
   WITH NO ADVANCING.
DISPLAY SCREEN(5,49) WITH NO ADVANCING.
DISPLAY '_ ' WITH NO ADVANCING.
DISPLAY SCREEN(5,49) WITH NO ADVANCING.

ACCOUNT-MAINTAIN-PROCESS.
CALL 'MTAN' USING SCREEN-TABLE 
   CLEAR-SCREEN RING-BELL 
   REVERSE-VIDEO EXIT-REVERSE.

ACCOUNT-UPDATE-PROCESS.
CALL 'UPDAT' USING SCREEN-TABLE 
   CLEAR-SCREEN RING-BELL 
   REVERSE-VIDEO EXIT-REVERSE.

REPORT-GENERATOR-PROCESS.
CALL 'REPOT' USING SCREEN-TABLE 
   CLEAR-SCREEN RING-BELL 
   REVERSE-VIDEO EXIT-REVERSE.
APPENDIX C

IDENTIFICATION DIVISION.

PROGRAM-ID. 
REPOT

AUTHOR. 
CCW

DATE-WRITTEN. 
23-MAR-1984

ENVIRONMENT DIVISION.

INPUT-OUTPUT SECTION.

FILE-CONTROL.

SELECT ENTERED-ACCOUNT-FILE,
   ASSIGN TO DSK,
   RECORDING MODE IS ASCII.

SELECT PRE-ENTERED-ACCOUNT-FILE,
   ASSIGN TO DSK,
   RECORDING MODE IS ASCII.

SELECT GENERAL-LEDGER-FILE
   ASSIGN TO DSK
   ORGANIZATION IS INDEXED
   ACCESS MODE IS DYNAMIC
   RECORD KEY IS G-ACCOUNT-NUMBER
   RECORDING MODE IS ASCII.

SELECT PRINT-FILE
   ASSIGN TO LPT
   RECORDING MODE IS ASCII.

DATA DIVISION.

FILE SECTION.

FD GENERAL-LEDGER-FILE, 
   BLOCK CONTAINS 128 RECORDS, 
   VALUE OF ID IS 'GENFILINX'.

01 GENERAL-LEDGER-RECORD.
   05 G-ACCOUNT-NUMBER PIC 9(6).
   05 G-ACCOUNT-NAME PIC X(30).
   05 G-ACCOUNT-TYPE PIC X.
   05 G-CURRENT-TOTAL PIC 9(9)V99.
   05 G-YEAR-TO-NOW-TOTAL PIC 9(9)V99.
   05 G-PREVIOUS-YEAR-TOTAL PIC 9(9)V99.

FD ENTERED-ACCOUNT-FILE, 
   BLOCK CONTAINS 128 RECORDS,
APPENDIX C

LABEL RECORDS ARE STANDARD,
VALUE OF ID IS 'ACCOUNFIL'.

01 ENTERED-ACCOUNT-RECORD.
 05 ENTRY-NO PIC 999999.
 05 ACC-NUMBER.
    08 CATAGORY-NO PIC 9.
    08 SUBCATAGORY-NO PIC 999999.
 05 ACC-NAME PIC X(30).
 05 SOURCE-CODE PIC 9.
 05 ACCOUNT-VALUE PIC S99999999V99.
 05 DATE-OF-ENTRY PIC XXXXXX.
 05 UPDATE-CODE PIC X.

FD PRE-ENTERED-ACCOUNT-FILE,
BLOCK CONTAINS 128 RECORDS,
LABEL RECORDS ARE STANDARD,
VALUE OF ID IS 'PREVI FIL'.

01 PRE-ENTERED-ACCOUNT-RECORD.
 05 PRE-ENTRY-NO PIC 999999.
 05 PRE-ACC-NO.
    08 PRE-CATAGORY-NO PIC 9.
    08 PRE-SUBCATAGORY-NO PIC 999999.
 05 PRE-ACC-NAME PIC X(30).
 05 PRE-SOURCE-CODE PIC 9.
 05 PRE-ACCOUNT-VALUE PIC S99999999V99.
 05 PRE-DATE-OF-ENTRY PIC XXXXXX.
 05 PRE-UPDATE-CODE PIC X.

FD PRINT-FILE,
VALUE OF ID IS 'OUTPT OUT',
LABEL RECORDS ARE STANDARD.

01 PRINT-RECORD PIC X(132).

*-----------------------
WORKING-STORAGE SECTION.
*-----------------------

01 LEDGER-TABLE.
 05 LEDGER-COUNTER PIC X(6)
                   OCCURS 200 TIMES.

01 TRIAL-BALANCE-TABLE OCCURS 200 TIMES.
 05 TRIAL-TABLE-ACCOUNT PIC X(6).
 05 TRIAL-TABLE-VALUE PIC S99999999V99.

01 TOATL-AMOUNT-TABLE.
 05 TOTAL-AMOUNT PIC S99999999V99
                   OCCURS 8 TIMES.

01 SCREEN-SYS-DATE.
 05 SCREEN-SYS-YEAR PIC X(2).
 05 FILLER PIC X(1) VALUE '/'.
 05 SCREEN-SYS-MONTH PIC X(2).
APPENDIX C

05 FILLER PIC X(1) VALUE '/'.
05 SCREEN-SYS-DAY PIC X(2).

01 SYS-DATE.
  05 SYS-YEAR PIC 9(2).
  05 SYS-MONTH PIC 9(2).
  05 SYS-DAY PIC 9(2).

01 TEST-VALUE-A.
  05 FRONT-A PIC X.
  05 BEHIND-A PIC X(9).

01 TEST-VALUE-B.
  05 FRONT-B PIC XX.
  05 BEHIND-B PIC X.

01 COMPANY-TITLE.
  05 FILLER PIC X(24).
  05 COMPANY-NAME PIC X(14) VALUE '**************'.
  05 FILLER PIC X(8).
  05 FILLER PIC X(7) VALUE 'COMPANY'.
  05 FILLER PIC X(79).

01 CHART-OF-ACC-HEADING-LINE.
  05 FILLER PIC X(22).
  05 FILLER PIC X(28) VALUE 'CHART OF ACCOUNT AS OF '.
  05 CHART-SYSDATE PIC X(8).
  05 FILLER PIC X(13) VALUE ' PAGE '.
  05 CHART-PAGE-NUMBER PIC Z9.
  05 FILLER PIC X(6) VALUE ' OF 5 '.
  05 FILLER PIC X(53).

01 CHART-BALANCE-LINE.
  05 FILLER PIC X(27).
  05 FILLER PIC X(21) VALUE '*** BALANCE SHEET ***'.
  05 FILLER PIC X(84).

01 CHART-INCOME-LINE.
  05 FILLER PIC X(27).
  05 FILLER PIC X(24) VALUE '*** INCOME STATEMENT ***'.
  05 FILLER PIC X(81).

01 CHART-IDENTIFY-LINE.
  05 FILLER PIC X(17) VALUE ' ACCT NO'.
  05 FILLER PIC X(15) VALUE ' ACCT'.
  05 FILLER PIC X(11) VALUE ' NAME'.

Page C-10
APPENDIX C

05 FILLER PIC X(14)
VALUE ' TYPE'.
05 FILLER PIC X(22)
VALUE ' AMOUNT'.
05 FILLER PIC X(53).

01 CHART-ACCOUNT-BODY-LINE.
05 FILLER PIC X(10).
05 CHART-ACCOUNT-NUMBER PIC X(6).
05 FILLER PIC X(4).
05 CHART-ACCOUNT-NAME PIC X(30).
05 FILLER PIC X(3).
05 CHART-ACCOUNT-TYPE PIC X(7).
05 FILLER PIC X(6).
05 CHART-ACCOUNT-VALUE PIC ZZ,ZZZ,ZZ9.99.
05 CHART-ACCOUNT-VALUE-MARK PIC X(2).
05 FILLER PIC X(51).

01 AUDIT-TRAIL-LINE.
05 FILLER PIC X(10).
05 AUDIT-ENTRY-NUMBER PIC X(6).
05 FILLER PIC X(6).
05 AUDIT-DATE-OF-ENTRY.
  10 AUDIT-TRAIL-YEAR PIC XX.
  10 FILLER PIC X VALUE '/'.
  10 AUDIT-TRAIL-MONTH PIC XX.
  10 FILLER PIC X VALUE '/'.
  10 AUDIT-TRAIL-DAY PIC XX.
05 FILLER PIC X(3).
05 AUDIT-ACCOUNT-NUMBER PIC X(6).
05 FILLER PIC X(6).
05 AUDIT-SOURCE-CODE PIC X(3).
05 FILLER PIC X(5).
05 AUDIT-ACCOUNT-NAME PIC X(30).
05 FILLER PIC X(3).
05 AUDIT-ACCOUNT-DR PIC ZZZ,ZZZ,ZZZ.ZZ.
05 FILLER PIC X(3).
05 AUDIT-ACCOUNT-CR PIC ZZZ,ZZZ,ZZZ.ZZ.
05 FILLER PIC X(13).

01 COMPANY-TITLE-2.
05 FILLER PIC X(40).
05 FILLER PIC X(14)
VALUE '***************'.
05 FILLER PIC X(15)
VALUE ' COMPANY'.
05 FILLER PIC X(63).

01 LIST-ENTERED-ACC-HEADING.
05 FILLER PIC X(30).
05 FILLER PIC X(11)
VALUE ' JOURNAL '.
05 FILLER PIC X(9)
VALUE ' ENTRY '.
05 FILLER PIC X(8)
VALUE ' LIST '.
APPENDIX C

05 FILLER       PIC X(7)       VALUE 'AS '.
05 FILLER       PIC X(6)       VALUE 'OF '.
05 LIST-ACC-ENTERED-HEADING-YEAR PIC X(2).
05 FILLER       PIC X(1) VALUE '/'.
05 LIST-ACC-ENTERED-HEADING-MONTH PIC X(2).
05 FILLER       PIC X(1) VALUE '/'.
05 LIST-ACC-ENTERED-HEADING-DAY PIC X(2).
05 FILLER       PIC X(22).
05 FILLER       PIC X(5) VALUE 'PAGE '.
05 LIST-ACC-ENTERED-HEADING-PAGE PIC ZZ.
05 FILLER       PIC X(4) VALUE ' OF '.
05 LIST-ACC-ENTERED-HEADING-TOTAL-PAGE PIC ZZ.
05 FILLER       PIC X(18).

01 AUDIT-HEADING-1.
05 FILLER       PIC X(104).
05 AUDIT-HEADING-YEAR PIC X(2).
05 FILLER       PIC X(1) VALUE '/'.
05 AUDIT-HEADING-MONTH PIC X(2).
05 FILLER       PIC X(1) VALUE '/'.
05 AUDIT-HEADING-DAY PIC X(2).

01 AUDIT-HEADING-2.
05 FILLER       PIC X(107).
05 FILLER       PIC X(2) VALUE 'TO'.
05 FILLER       PIC X(23).

01 AUDIT-HEADING-3.
05 FILLER       PIC X(54).
05 FILLER       PIC X(15)       VALUE 'AUDIT '.
05 FILLER       PIC X(5)       VALUE 'TRAIL'.
05 FILLER       PIC X(30).
05 FILLER       PIC X(5)       VALUE 'PAGE '.
05 AUDIT-HEADING-PAGE PIC ZZ9.
05 FILLER       PIC X(4)       VALUE ' OF '.
05 AUDIT-HEADING-TOTAL-PAGE PIC ZZ9.
05 FILLER       PIC X(13).

01 AUDIT-HEADING-4.
05 FILLER       PIC X(17)       VALUE ' JOURNAL'.
05 FILLER       PIC X(10)       VALUE 'ENTRY'.
05 FILLER       PIC X(14)       VALUE 'ACCOUNT'.
05 FILLER       PIC X(9)       VALUE 'SOURCE'.
05 FILLER       PIC X(45).
05 FILLER       PIC X(6)       VALUE 'AMOUNT'.

APPENDIX C

05 FILLER       PIC X(31).

01 AUDIT-HEADING-5.
  05 FILLER       PIC X(19).
  VALUE ' ENTRY NO.'.
  05 FILLER       PIC X(7).
  VALUE ' DATE'.
  05 FILLER       PIC X(13).
  VALUE ' NO.'.
  05 FILLER       PIC X(9).
  VALUE ' CODE'.
  05 FILLER       PIC X(19).
  VALUE ' ACCOUNT'.
  05 FILLER       PIC X(9).
  VALUE ' NAME'.
  05 FILLER       PIC X(23).
  05 FILLER       PIC X(15).
  VALUE ' DR CR'.
  05 FILLER       PIC X(18).

01 AUDIT-HEADING-6.
  05 FILLER       PIC X(10).
  05 FILLER       PIC X(23).
  VALUE ' __________ __________ '.
  05 FILLER       PIC X(20).
  VALUE ' __________ __________ '.
  05 FILLER       PIC X(30).
  VALUE ' ________________________________ '.
  05 FILLER       PIC X(32).
  VALUE ' ________________________________ '.
  05 FILLER       PIC X(17).

01 LEDGER-SHEET-HEADING.
  05 FILLER       PIC X(28).
  05 FILLER       PIC X(14).
  VALUE ' LEDGER '.
  05 FILLER       PIC X(5).
  VALUE ' SHEET '.
  05 FILLER       PIC X(23).
  05 FILLER       PIC X(5).
  VALUE ' PAGE '.
  05 LEDGER-HEADING-PAGE       PIC ZZ.
  05 FILLER       PIC X(4).
  VALUE ' OF '.
  05 LEDGER-HEADING-TOTAL-PAGE       PIC ZZ.
  05 FILLER       PIC X(49).

01 LEDGER-SHEET-TITLE-LINE.
  05 FILLER       PIC X(73).
  05 LEDGER-SHEET-TITLE-DATE       PIC X(8).
  05 FILLER       PIC X(51).

01 LEDGER-SHEET-LINE.
  05 FILLER       PIC X(11).
  05 LEDGER-SHEET-ACCOUNT-NUMBER       PIC X(6).
  05 FILLER       PIC X(5).
APPENDIX C

05 LEDGER-SHEET-SOURCE-CODE PIC X(3).
05 FILLER PIC X(5).
05 LEDGER-SHEET-ENTRY-NUMBER PIC X(6).
05 FILLER PIC X(5).
05 LEDGER-SHEET-DATE-OF-ENTRY.
 08 LEDGER-SHEET-YEAR PIC XX.
 08 FILLER PIC X VALUE '/'.
 08 LEDGER-SHEET-MONTH PIC XX.
 08 FILLER PIC X VALUE '/'.
 08 LEDGER-SHEET-DAY PIC XX.
 08 FILLER PIC XXX.
05 LEDGER-SHEET-ACCOUNT-DR PIC ZZZ,ZZZ,ZZZ,ZZ.
05 FILLER PIC X(3).
05 LEDGER-SHEET-ACCOUNT-CR PIC ZZZ,ZZZ,ZZZ,ZZ.
05 FILLER PIC X(49).

01 LEDGER-SHEET-LINE-2.
 05 FILLER PIC X(21) VALUE 'ACCOUNT'.
 05 FILLER PIC X(9) VALUE 'SOURCE'.
 05 FILLER PIC X(7) VALUE 'JOURNAL'.
 05 FILLER PIC X(27).
 05 FILLER PIC X(19) VALUE 'AMOUNT'.

01 LEDGER-SHEET-LINE-3.
 05 FILLER PIC X(16) VALUE 'NO.'. 
 05 FILLER PIC X(14) VALUE 'CODE'.
 05 FILLER PIC X(9) VALUE 'ENTRY NO.'. 
 05 FILLER PIC X(7) VALUE 'DATE'.
 05 FILLER PIC X(12) VALUE 'DR'.
 05 FILLER PIC X(17) VALUE 'CR'.
 05 FILLER PIC X(38).

01 LEDGER-SHEET-LINE-4.
 05 FILLER PIC X(11).
 05 FILLER PIC X(10) VALUE '________'.
 05 FILLER PIC X(9) VALUE '________'.
 05 FILLER PIC X(11) VALUE '________'.
 05 FILLER PIC X(11) VALUE '________'.
 05 FILLER PIC X(28) VALUE '________________________'.
 05 FILLER PIC X(52).
01 LEDGER-SHEET-LINE-5.
  05 FILLER PIC X(42).
  05 FILLER PIC X(14)
    VALUE 'NTE CHANGE'.
  05 LEDGER-SHEET-LINE5-BALANCE PIC ZZZ,ZZZ,ZZZ.ZZ.
  05 LEDGER-SHEET-LINE5-MARK PIC XX.
  05 FILLER PIC X(60).

01 LEDGER-SHEET-LINE-6.
  05 FILLER PIC X(20).
  05 FILLER PIC X(12)
    VALUE 'ACCOUNT NAME'.
  05 FILLER PIC X(100).

01 LEDGER-SHEET-LINE-7.
  05 FILLER PIC X(11).
  05 LEDGER-SHEET-LINE7-NAME PIC X(30).
  05 FILLER PIC X(91).

01 LEDGER-SHEET-LINE-8.
  05 FILLER PIC X(11).
  05 LEDGER-SHEET-LINE8-LABEL PIC X(16).
  05 FILLER PIC X(2).
  05 LEDGER-SHEET-LINE8-VALUE PIC ZZZ,ZZZ,ZZZ.ZZ.
  05 LEDGER-SHEET-LINE8-MARK PIC X(2).
  05 FILLER PIC X(87).

01 TRIAL-BALANCE-TITLE.
  05 FILLER PIC X(35).
  05 FILLER PIC X(14)
    VALUE 'XXXXXXXXXXXXXXXX'.
  05 FILLER PIC X(11)
    VALUE 'COMPANY'.
  05 FILLER PIC X(18).
  05 TRIAL-BALANCE-TITLE-DATE PIC X(8).
  05 FILLER PIC X(46).

01 TRIAL-BALANCE-LINE-1.
  05 FILLER PIC X(36).
  05 FILLER PIC X(9)
    VALUE 'TRIAL'.
  05 FILLER PIC X(15)
    VALUE 'BALANCE SHEET'.
  05 FILLER PIC X(20)
    VALUE 'PAGE'.
  05 TRIAL-LINE1-PAGE-NUMBER PIC ZZ.
  05 FILLER PIC X(6)
    VALUE 'OF 4'.
  05 FILLER PIC X(44).

01 TRIAL-BALANCE-LINE-2.
  05 FILLER PIC X(17)
    VALUE 'ACCOUNT'.
  05 FILLER PIC X(19)
    VALUE 'ACCOUNT'.

APPENDIX C

05 FILLER       PIC X(31).
05 FILLER       PIC X(6)
                VALUE 'AMOUNT'.
05 FILLER       PIC X(59).

01 TRIAL-BALANCE-LINE-3.
  05 FILLER       PIC X(15)
                  VALUE ' NO.'.
  05 FILLER       PIC X(18)
                  VALUE ' NAME'.
  05 FILLER       PIC X(26).
  05 FILLER       PIC X(15)
                  VALUE 'DR
                  CR'.
  05 FILLER       PIC X(58).

01 TRIAL-BALANCE-LINE-4.
  05 FILLER       PIC X(10).
  05 FILLER       PIC X(11)
                  VALUE ' '.
  05 FILLER       PIC X(30)
                  VALUE ' '.
  05 FILLER       PIC X(4).
  05 FILLER       PIC X(30)
                  VALUE ' '.
  05 FILLER       PIC X(47).

01 TRIAL-BALANCE-LINE-5.
  05 FILLER       PIC X(10).
  05 FILLER       PIC X(6).
  05 FILLER       PIC X(4).
  05 FILLER       PIC X(30).
  05 FILLER       PIC X(4).
  05 FILLER       PIC X(14).
  05 FILLER       PIC X(4).
  05 FILLER       PIC X(14).
  05 FILLER       PIC X(46).

01 TRIAL-BALANCE-LINE-6.
  05 FILLER       PIC X(16).
  05 FILLER       PIC X(38)
                  VALUE '====================================='.
  05 FILLER       PIC X(37)
                  VALUE '====================================='.
  05 FILLER       PIC X(41).

01 TRIAL-BALANCE-LINE-7.
  05 FILLER       PIC X(45).
  05 FILLER       PIC X(9)
                  VALUE 'TOTAL '.
  05 FILLER       PIC ZZZ,ZZZ,ZZZ.ZZ.
  05 FILLER       PIC X(3)
                  VALUE '='.
  05 FILLER       PIC ZZZ,ZZZ,ZZZ.ZZ.
  05 FILLER       PIC X(47).

01 BALANCE-SHEET-LINE.
APPENDIX C

05 FILLER PIC X(5).
05 BALANCE-SHEET-ACC-NAME PIC X(48).
05 FILLER PIC X(4).
05 BALANCE-DOLLAR-SIGN-1 PIC X.
05 FILLER PIC X(2).
05 BALANCE-SHEET-MARK-1 PIC X.
05 BALANCE-SHEET-ACC-VALUE PIC Z999.99.
05 BALANCE-SHEET-MARK-2 PIC X.
05 FILLER PIC X(5).
05 BALANCE-DOLLAR-SIGN-2 PIC X.
05 FILLER PIC X(2).
05 BALANCE-SHEET-MARK-3 PIC X.
05 BALANCE-SHEET-TOTAL PIC Z999.99.
05 BALANCE-SHEET-MARK-4 PIC X.
05 FILLER PIC X(32).

01 BALANCE-SHEET-HEADING-1.
05 FILLER PIC X(38).
05 FILLER PIC X(13)
  VALUE 'BALANCE SHEET'.
05 FILLER PIC X(27)
  VALUE 'PAGE'.
05 BALANCE-SHEET-PAGE-NUMBER PIC Z9.
05 FILLER PIC X(6)
  VALUE 'OF 3'.
05 FILLER PIC X(31).

01 BALANCE-SHEET-HEADING-2.
05 FILLER PIC X(40).
05 FILLER PIC X(8)
  VALUE 'AS OF '.
05 BALANCE-SHEET-HEADING-DATE PIC X(8).
05 FILLER PIC X(76).

01 INCOME-STATEMENT-HEADING-2.
05 FILLER PIC X(46).
05 FILLER PIC X(8)
  VALUE 'AS OF '.
05 INCOME-STATEMENT-DATE PIC X(8).
05 FILLER PIC X(70).

01 INCOME-STATEMENT-HEADING-1.
05 FILLER PIC X(38).
05 FILLER PIC X(16)
  VALUE 'INCOME STATEMENT'.
05 FILLER PIC X(27)
  VALUE 'PAGE'.
05 INCOME-STATEMENT-PAGE-NUMBER PIC Z9.
05 FILLER PIC X(6)
  VALUE 'OF 3'.
05 FILLER PIC X(28).

01 TRIAL-LINES-ACCOUNT-TEMPLE PIC Z999.99.

01 LINE-NUMBER PIC 9(2).
APPENDIX C

01 TOTAL-PAGE-NUMBER PIC 99.
01 PAGE-NUMBER PIC 9(2).
01 REPORT-TYPE PIC X.
01 TABLE-INDEX PIC 999.
01 INDEX-A PIC 999.
01 INDENT-1-OUTPUT.
  05 INDENT-1 PIC X(30).
  05 FILLER PIC X(18).
01 INDENT-2-OUTPUT.
  05 FILLER PIC X(3).
  05 INDENT-2 PIC X(30).
  05 FILLER PIC X(15).
01 INDENT-3-OUTPUT.
  05 FILLER PIC X(6).
  05 INDENT-3 PIC X(30).
  05 FILLER PIC X(12).
01 INDENT-4-OUTPUT.
  05 FILLER PIC X(9).
  05 INDENT-4 PIC X(30).
  05 FILLER PIC X(9).
01 INDENT-5-OUTPUT.
  05 FILLER PIC X(12).
  05 INDENT-5 PIC X(30).
  05 FILLER PIC X(6).
01 INDENT-6-OUTPUT.
  05 FILLER PIC X(15).
  05 INDENT-6 PIC X(30).
  05 FILLER PIC X(3).
01 INDENT-7-OUTPUT.
  05 FILLER PIC X(18).
  05 INDENT-7 PIC X(30).
01 INDENT-LEVEL-INDEX PIC 9.
01 TOTAL-LEVEL-INDEX PIC 9.
01 PRE-ACC-TYPE PIC X(6).
01 REPEAT-COUNTER PIC 9.
01 CHECK-DATA PIC 99.
01 NET-CHANGE PIC S99999999V99.
APPENDIX C

01 CR-TOTAL
  PIC S99999999V99.

01 DR-TOTAL
  PIC S99999999V99.

01 INPUT-DATE.
  05 I-YEAR
    PIC 99.
  05 I-MONTH
    PIC 99.
  05 I-DAY
    PIC 99.

01 END-DATE.
  05 E-YEAR
    PIC 99.
  05 E-MONTH
    PIC 99.
  05 E-DAY
    PIC 99.

01 BEGIN-DATE.
  05 B-YEAR
    PIC 99.
  05 B-MONTH
    PIC 99.
  05 B-DAY
    PIC 99.

77 EOF-PRE-FILE-OR-NOT
  PIC X VALUE 'N'.
  88 EOF-PRE-FILE
    PIC X VALUE 'Y'.

77 EOF-ENTERED-ACC-FILE-OR-NOT
  PIC X VALUE 'N'.
  88 EOF-ENTERED-ACC-FILE
    PIC X VALUE 'Y'.

77 ACCOUNT-EXIST-OR-NOT
  PIC X VALUE 'N'.
  88 ACCOUNT-EXIST
    PIC X VALUE 'Y'.

77 OVER-END-DATE-OR-NOT
  PIC X VALUE 'N'.
  88 OVER-END-DATE
    PIC X VALUE 'Y'.

77 FIND-BEGIN-DATE-OK
  PIC X VALUE 'N'.
  88 FIND-BEGIN-DATE
    PIC X VALUE 'Y'.

77 EXIT-GENERATE-REPORT-OR-NOT
  PIC X VALUE 'N'.
  88 EXIT-GENERATE-REPORT
    PIC X VALUE 'Y'.

77 BUILD-TABLE-OK-OR-NOT
  PIC X VALUE 'N'.
  88 BUILD-TABLE-OK
    PIC X VALUE 'Y'.

77 TAKE-TYPE-OK-OR-NOT
  PIC X VALUE 'N'.
  88 TAKE-TYPE-OK
    PIC X VALUE 'Y'.

77 EOF-G-L-FILE-OR-NOT
  PIC X VALUE 'N'.
  88 EOF-G-L-FILE
    PIC X VALUE 'Y'.

LINKAGE SECTION.

01 SCREEN-TABLE.
  05 SCREEN-ROWS
    PIC X(4)
    OCCURS 24 TIMES.
  10 SCREEN
    USAGE IS DISPLAY-7
    OCCURS 80 TIMES.

01 REVERSE-VIDEO
  PIC XX DISPLAY-7.
APPENDIX C

PROCEDURE DIVISION USING SCREEN-TABLE
CLEAR-SCREEN RING-BELL
REVERSE-VIDEO EXIT-REVERSE.

MAIN.
ACCEPT SYS-DATE FROM DATE.
MOVE SYS-YEAR TO SCREEN-SYS-YEAR.
MOVE SYS-MONTH TO SCREEN-SYS-MONTH.
MOVE SYS-DAY TO SCREEN-SYS-DAY.
MOVE 'N' TO EXIT-GENERATE-REPORT-OR-NOT.
PERFORM SELECT-REPORT-TYPE-SCREEN.
PERFORM REPORT-GENERATOR-PROCESS UNTIL
EXIT-GENERATE-REPORT.

EXIT PROGRAM.

REPORT-GENERATOR-PROCESS.
MOVE 'N' TO TAKE-TYPE-OK-OR-NOT.
PERFORM TAKE-TYPE-OF-REPORT UNTIL TAKE-TYPE-OK.
IF REPORT-TYPE = '1'
   PERFORM PRINT-CHART-OF-ACCOUNT-SCREEN
   PERFORM PRINT-CHART-OF-ACCOUNT
ELSE IF REPORT-TYPE = '2'
   PERFORM PRINT-LIST-ACC-ENTERED-SCREEN
   PERFORM PRINT-LIST-OF-ACC-ENTERED
ELSE IF REPORT-TYPE = '3'
   PERFORM PRINT-AUDIT-TRAIL-SCREEN
   PERFORM PRINT-AUDIT-TRAIL
ELSE IF REPORT-TYPE = '4'
   PERFORM PRINT-LEDGER-SHEET-SCREEN
   PERFORM PRINT-LEDGER-SHEET
ELSE IF REPORT-TYPE = '5'
   PERFORM PRINT-TRIAL-BALANCE-SCREEN
   PERFORM PRINT-TRIAL-BALANCE
ELSE IF REPORT-TYPE = '6'
   PERFORM PRINT-INCOME-STATEMENT-SCREEN
   PERFORM PRINT-INCOME-STATEMENT
ELSE IF REPORT-TYPE = '7'
   PERFORM PRINT-BALANCE-SHEET-SCREEN
   PERFORM PRINT-BALANCE-SHEET
ELSE IF REPORT-TYPE = '*'
   MOVE 'Y' TO EXIT-GENERATE-REPORT-OR-NOT.
DISPLAY SCREEN(22,7) WITH NO ADVANCING.
DISPLAY '
WITH NO ADVANCING.

SELECT-REPORT-TYPE-SCREEN.
DISPLAY CLEAR-SCREEN WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
APPENDIX C

DISPLAY SCREEN(1,28) WITH NO ADVANCING.
DISPLAY 'SCREEN - 1.3'
    WITH NO ADVANCING.
DISPLAY SCREEN(2,21) WITH NO ADVANCING.
DISPLAY 'REPORT GENERATOR SELECTION'
    WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY SCREEN (4,17) WITH NO ADVANCING.
DISPLAY 'SELECT ONE OF THE FOLLOWING TYPE TO PROCESS'
    WITH NO ADVANCING.
DISPLAY SCREEN (5,31) WITH NO ADVANCING.
DISPLAY 'ENTER SELECTION: _'
    WITH NO ADVANCING.
DISPLAY SCREEN(7,16) WITH NO ADVANCING.
DISPLAY '=================================='
    WITH NO ADVANCING.
DISPLAY SCREEN(9,16) WITH NO ADVANCING.
DISPLAY '  * : FOR EXIT ACCOUNT UPDATE PROCESS '
    WITH NO ADVANCING.
DISPLAY SCREEN(10,16) WITH NO ADVANCING.
DISPLAY '  1 : PRINT CHART OF ACCOUNT '
    WITH NO ADVANCING.
DISPLAY SCREEN(11,16) WITH NO ADVANCING.
DISPLAY '  2 : PRINT LIST OF ACCOUNT ENTERED'
    WITH NO ADVANCING.
DISPLAY SCREEN(12,16) WITH NO ADVANCING.
DISPLAY '  3 : PRINT AUDIT TRAIL '
    WITH NO ADVANCING.
DISPLAY SCREEN(13,16) WITH NO ADVANCING.
DISPLAY '  4 : PRINT LEDGER SHEET'
    WITH NO ADVANCING.
DISPLAY SCREEN(14,16) WITH NO ADVANCING.
DISPLAY '  5 : PRINT TRIAL BALANCE '
    WITH NO ADVANCING.
DISPLAY SCREEN(15,16) WITH NO ADVANCING.
DISPLAY '  6 : PRINT INCOME STATEMENT '
    WITH NO ADVANCING.
DISPLAY SCREEN(16,16) WITH NO ADVANCING.
DISPLAY '  7 : PRINT BALANCE SHEET '
    WITH NO ADVANCING.
DISPLAY SCREEN(18,16) WITH NO ADVANCING.
DISPLAY '=================================='
    WITH NO ADVANCING.

TAKE-TYPE-OF-REPORT.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY SCREEN(5,49) WITH NO ADVANCING.
DISPLAY '_
    ' WITH NO ADVANCING.
DISPLAY SCREEN(5,49) WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A NOT EQUAL SPACES PERFORM ERROR-HANDLE
ELSE MOVE FRONT-A TO REPORT-TYPE.
IF REPORT-TYPE EQUAL '1' OR EQUAL '2' OR
   EQUAL '3' OR EQUAL '4' OR
   EQUAL '5' OR EQUAL '6' OR
   EQUAL '7' OR EQUAL '8'
   MOVE 'Y' TO TAKE-TYPE-OK-OR-NOT
ELSE
   PERFORM ERROR-HANDLE.
DISPLAY SCREEN(22,7) WITH NO ADVANCING.
DISPLAY '     WITH NO ADVANCING.
DISPLAY '     WITH NO ADVANCING.

ERROR-HANDLE.
DISPLAY SCREEN(22,7) WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY 'ERROR MESSAGE: ' WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY 'INVALID RESPONSE. PLEASE REENTER ' WITH NO ADVANCING.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY SCREEN(5,49) WITH NO ADVANCING.
DISPLAY '_' WITH NO ADVANCING.
DISPLAY SCREEN(5,49) WITH NO ADVANCING.
PERFORM TAKE-TYPE-OF-REPORT.

PRINT-COMPANY-TITLE.
WRITE PRINT-RECORD FROM COMPANY-TITLE AFTER ADVANCING 4 LINES.

PRINT-CHART-OF-ACCOUNT-SCREEN.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY SCREEN(22,17) WITH NO ADVANCING.
DISPLAY 'PRINT CHART OF ACCOUNT PROCESSING' WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.

PRINT-CHART-OF-ACCOUNT.
MOVE SCREEN-SYS-DATE TO CHART-SYSDATE.
MOVE 'N' TO EOF-G-L-FILE-OR-NOT.
OPEN INPUT GENERAL-LEDGER-FILE OUTPUT PRINT-FILE.
MOVE 1 TO PAGE-NUMBER.
MOVE 0 TO LINE-NUMBER.
PERFORM PRINT-COMPANY-TITLE.
ADD 4 TO LINE-NUMBER.
PERFORM CHART-OF-ACCOUNT-BODY UNTIL EOF-G-L-FILE.
CLOSE GENERAL-LEDGER-FILE PRINT-FILE.

CHART-OF-ACCOUNT-BODY.
IF PAGE-NUMBER EQUAL 1 OR LINE-NUMBER
   EQUAL 0 PERFORM CHART-OF-ACCOUNT-HEADING.
ADD 9 TO LINE-NUMBER.
READ GENERAL-LEDGER-FILE NEXT AT END.
MOVE 'Y' TO EOF-G-L-FILE-OR-NOT.
PERFORM CHART-OF-ACCOUNT-LINE UNTIL LINE-NUMBER GREATER THAN 50.
ADD 1 TO PAGE-NUMBER.
MOVE 0 TO LINE-NUMBER.

CHART-OF-ACCOUNT-HEADING.
MOVE PAGE-NUMBER TO CHART-PAGE-NUMBER.
IF PAGE-NUMBER GREATER THAN 1
WRITE PRINT-RECORD FROM SPACES AFTER ADVANCING PAGE.
WRITE PRINT-RECORD FROM CHART-OF-ACC-HEADING-LINE AFTER ADVANCING 3 LINES.
IF G-ACCOUNT-NUMBER LESS THAN 300000
WRITE PRINT-RECORD FROM CHART-BALANCE-LINE AFTER ADVANCING 5 LINES
ELSE
WRITE PRINT-RECORD FROM CHART-INCOME-LINE AFTER ADVANCING 5 LINES.
WRITE PRINT-RECORD FROM CHART-IDENTIFY-LINE AFTER ADVANCING 3 LINES.
WRITE PRINT-RECORD FROM SPACES.

CHART-OF-ACCOUNT-LINE.
IF CHART-ACCOUNT-NUMBER LESS THAN 300000 AND G-ACCOUNT-NUMBER NOT LESS THAN 300000
WRITE PRINT-RECORD FROM CHART-INCOME-LINE AFTER ADVANCING 2 LINES
WRITE PRINT-RECORD FROM SPACES
ADD 3 TO LINE-NUMBER.
MOVE G-ACCOUNT-NAME TO CHART-ACCOUNT-NAME.
MOVE G-ACCOUNT-NUMBER TO CHART-ACCOUNT-NUMBER.
IF G-ACCOUNT-TYPE EQUAL 0
MOVE 'REGULAR' TO CHART-ACCOUNT-TYPE
ELSE IF G-ACCOUNT-TYPE EQUAL 1
MOVE 'TITLE' TO CHART-ACCOUNT-TYPE
ELSE IF G-ACCOUNT-TYPE EQUAL 2
MOVE 'TOTAL' TO CHART-ACCOUNT-TYPE
ELSE IF G-ACCOUNT-TYPE EQUAL 3
MOVE 'HEADING' TO CHART-ACCOUNT-TYPE.
IF G-CURRENT-TOTAL GREATER THAN 0
MOVE 'CR' TO CHART-ACCOUNT-VALUE-MARK
ELSE MOVE 'DR' TO CHART-ACCOUNT-VALUE-MARK
MULTIPLY -1 BY G-CURRENT-TOTAL.
MOVE G-CURRENT-TOTAL TO CHART-ACCOUNT-VALUE.
WRITE PRINT-RECORD FROM CHART-ACCOUNT-BODY-LINE.
ADD 1 TO LINE-NUMBER.
READ GENERAL-LEDGER-FILE NEXT AT END.
MOVE 'Y' TO EOF-G-L-FILE-OR-NOT
MOVE 51 TO LINE-NUMBER.

PRINT-LIST-ACC-ENTERED-SCREEN.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY SCREEN(22,17) WITH NO ADVANCING.
APPENDIX C

DISPLAY 'PRINT LIST OF ACCOUNT ENTERED PROCESSING'
WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.

PRINT-LIST-OF-ACC-ENTERED.
MOVE 0 TO REPEAT-COUNTER.
MOVE 0 TO TOTAL-PAGE-NUMBER.
PERFORM LIST-ENTERED-ACCOUNT-PROCESS
UNTIL REPEAT-COUNTER EQUAL 2.
ACCEPT SYS-DATE FROM DATE.

LIST-ENTERED-ACCOUNT-PROCESS.
OPEN INPUT ENTERED-ACCOUNT-FILE.
IF REPEAT-COUNTER EQUAL 1
OPEN OUTPUT PRINT-FILE.
MOVE 1 TO PAGE-NUMBER.
MOVE 0 TO LINE-NUMBER.
IF REPEAT-COUNTER EQUAL 1
WRITE PRINT-RECORD FROM COMPANY-TITLE-2
AFTER ADVANCING 4 LINES.
ADD 4 TO LINE-NUMBER.
MOVE 'N' TO EOF-ENTERED-ACC-FILE-OR-NOT.
READ ENTERED-ACCOUNT-FILE AT END MOVE 'Y'
TO EOF-ENTERED-ACC-FILE-OR-NOT.
PERFORM LIST-ENTERED-ACC-BODY
UNTIL EOF-ENTERED-ACC-FILE.
CLOSE ENTERED-ACCOUNT-FILE.
IF REPEAT-COUNTER EQUAL 1
CLOSE PRINT-FILE.
ADD 1 TO REPEAT-COUNTER.
SUBTRACT 1 FROM PAGE-NUMBER GIVING PAGE-NUMBER.
MOVE PAGE-NUMBER TO TOTAL-PAGE-NUMBER.

LIST-ENTERED-ACC-BODY.
IF REPEAT-COUNTER EQUAL 1
PERFORM PRINT-LIST-ENTERED-ACC-HEADING.
ADD 10 TO LINE-NUMBER.
PERFORM LIST-ENTERED-ACC-LINE UNTIL
LINE-NUMBER EQUAL 50 OR EOF-ENTERED-ACC-FILE.
MOVE 0 TO LINE-NUMBER.
ADD 1 TO PAGE-NUMBER.

LIST-ENTERED-ACC-LINE.
MOVE ENTRY-NUMBER TO AUDIT-ENTRY-NUMBER.
MOVE ACC-NAME TO AUDIT-ACCOUNT-NAME.
MOVE ACC-NUMBER TO AUDIT-ACCOUNT-NUMBER.
MOVE PRE-DATE-OF-ENTRY TO SYS-DATE.
MOVE SYS-YEAR TO AUDIT-TRAIL-YEAR.
MOVE SYS-MONTH TO AUDIT-TRAIL-MONTH.
MOVE SYS-DAY TO AUDIT-TRAIL-DAY.
IF SOURCE-CODE EQUAL 1 MOVE 'A/R' TO
AUDIT-SOURCE-CODE
ELSE IF SOURCE-CODE EQUAL 2 MOVE 'A/P' TO
AUDIT-SOURCE-CODE
ELSE IF SOURCE-CODE EQUAL 3 MOVE 'F/R' TO
AUDIT-SOURCE-CODE
APPENDIX C

ELSE IF SOURCE-CODE EQUAL 4 MOVE 'ADJ' TO AUDIT-SOURCE-CODE.
MOVE SPACES TO AUDIT-ACCOUNT-CR.
MOVE SPACES TO AUDIT-ACCOUNT-DR.
IF ACCOUNT-VALUE GREATER THAN 0
  MOVE ACCOUNT-VALUE TO AUDIT-ACCOUNT-CR
ELSE
  MULTIPLY -1 BY ACCOUNT-VALUE
  MOVE ACCOUNT-VALUE TO AUDIT-ACCOUNT-DR.
IF DATE-OF-ENTRY EQUAL SYS-DATE
  ADD 1 TO LINE-NUMBER.
IF DATE-OF-ENTRY EQUAL SYS-DATE AND REPEAT-COUNTER EQUAL 1
  WRITE PRINT-RECORD FROM AUDIT-TRAIL-LINE.
READ ENTERED-ACCOUNT-FILE AT END
  MOVE 'Y' TO EOF-ENTERED-ACC-FILE-OR-NOT.

PRINT-LIST-ENTERED-ACC-HEADING.
  MOVE SYS-YEAR TO LIST-ACC-ENTERED-HEADING-YEAR.
  MOVE SYS-MONTH TO LIST-ACC-ENTERED-HEADING-MONTH.
  MOVE SYS-DAY TO LIST-ACC-ENTERED-HEADING-DAY.
  MOVE PAGE-NUMBER TO LIST-ACC-ENTERED-HEADING-PAGE.
  MOVE TOTAL-PAGE-NUMBER TO LIST-ACC-HEADING-TOTAL-PAGE.
  IF PAGE-NUMBER GREATER THAN 1 WRITE PRINT-RECORD FROM SPACES AFTER ADVANCING PAGE.
  WRITE PRINT-RECORD FROM LIST-ENTERED-ACC-HEADING AFTER ADVANCING 2 LINES.
  WRITE PRINT-RECORD FROM AUDIT-HEADING-4 AFTER ADVANCING 4 LINES.
  WRITE PRINT-RECORD FROM AUDIT-HEADING-5.
  WRITE PRINT-RECORD FROM AUDIT-HEADING-6 BEFORE ADVANCING 3 LINES.

PRINT-AUDIT-TRAIL-SCREEN.
  DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
  DISPLAY SCREEN(22,17) WITH NO ADVANCING.
  DISPLAY 'PRINT AUDIT TRAIL PROCESSING' WITH NO ADVANCING.
  DISPLAY EXIT-REVERSE WITH NO ADVANCING.

PRINT-AUDIT-TRAIL.
  PERFORM GET-BEGIN-END-DATE.
  MOVE 0 TO REPEAT-COUNTER.
  MOVE 0 TO TOTAL-PAGE-NUMBER.
  PERFORM PRINT-AUDIT-TRAIL-PROCESS UNTIL REPEAT-COUNTER EQUAL 2.
  ACCEPT SYS-DATE FROM DATE.

PRINT-AUDIT-TRAIL-PROCESS.
  OPEN INPUT PRE-ENTERED-ACCOUNT-FILE.
  IF REPEAT-COUNTER EQUAL 1
    OPEN OUTPUT PRINT-FILE.
    MOVE 1 TO PAGE-NUMBER.
    MOVE 0 TO LINE-NUMBER.
APPENDIX C

IF REPEAT-COUNTER EQUAL 1 PERFORM
  PRINT-AUDIT-TRAIL-TITLE.
ADD 7 TO LINE-NUMBER.
MOVE 'N' TO EOF-PRE-FILE-OR-NOT.
MOVE 'N' TO FIND-BEGIN-DATE-OK.
MOVE 'N' TO OVER-END-DATE-OR-NOT.
READ PRE-ENTERED-ACCOUNT-FILE AT END
  MOVE 'Y' TO EOF-PRE-FILE-OR-NOT.
PERFORM GO-TO-BEGIN-DATE UNTIL FIND-BEGIN-DATE
  OR EOF-PRE-FILE.
PERFORM PRINT-AUDIT-TRAIL-BODY UNTIL
  EOF-PRE-FILE OR OVER-END-DATE.
CLOSE PRE-ENTERED-ACCOUNT-FILE.
IF REPEAT-COUNTER EQUAL 1
  CLOSE PRINT-FILE.
ADD 1 TO REPEAT-COUNTER.
SUBTRACT 1 FROM PAGE-NUMBER GIVING PAGE-NUMBER.
MOVE PAGE-NUMBER TO TOTAL-PAGE-NUMBER.

PRINT-AUDIT-TRAIL-BODY.
IF REPEAT-COUNTER EQUAL 1
  PERFORM PRINT-AUDIT-TRAIL-HEADING.
ADD 10 TO LINE-NUMBER.
PERFORM PRINT-AUDIT-TRAIL-LINE UNTIL OVER-END-DATE
  OR LINE-NUMBER EQUAL 50 OR EOF-PRE-FILE.
MOVE 0 TO LINE-NUMBER.
ADD 1 TO PAGE-NUMBER.

PRINT-AUDIT-TRAIL-LINE.
MOVE PRE-ENTRY-NO TO AUDIT-ENTRY-NUMBER.
MOVE PRE-ACC-NAME TO AUDIT-ACCOUNT-NAME.
MOVE PRE-ACC-NO TO AUDIT-ACCOUNT-NUMBER.
MOVE PRE-DATE-OF-ENTRY TO SYS-DATE.
MOVE SYS-YEAR TO AUDIT-TRAIL-YEAR.
MOVE SYS-MONTH TO AUDIT-TRAIL-MONTH.
MOVE SYS-DAY TO AUDIT-TRAIL-DAY.
IF PRE-SOURCE-CODE EQUAL 1 MOVE 'A/R' TO
  AUDIT-SOURCE-CODE
ELSE IF PRE-SOURCE-CODE EQUAL 2 MOVE 'A/P' TO
  AUDIT-SOURCE-CODE
ELSE IF PRE-SOURCE-CODE EQUAL 3 MOVE 'P/R' TO
  AUDIT-SOURCE-CODE
ELSE IF PRE-SOURCE-CODE EQUAL 4 MOVE 'ADJ' TO
  AUDIT-SOURCE-CODE.
MOVE SPACES TO AUDIT-ACCOUNT-CR.
MOVE SPACES TO AUDIT-ACCOUNT-DR.
IF PRE-ACCOUNT-VALUE GREATER THAN 0
  MOVE PRE-ACCOUNT-VALUE TO AUDIT-ACCOUNT-CR
ELSE
  MULTIPLY -1 BY PRE-ACCOUNT-VALUE
  MOVE PRE-ACCOUNT-VALUE TO AUDIT-ACCOUNT-DR.
IF REPEAT-COUNTER EQUAL 1
  WRITE PRINT-RECORD FROM AUDIT-TRAIL-LINE.
ADD 1 TO LINE-NUMBER.
READ PRE-ENTERED-ACCOUNT-FILE AT END
  MOVE 'Y' TO EOF-PRE-FILE-OR-NOT.
APPENDIX C

IF PRE-DATE-OF-ENTRY GREATER THAN END-DATE
   MOVE 'Y' TO OVER-END-DATE-OR-NOT.

GET-BEGIN-END-DATE.
   PERFORM GET-BEGIN-DATE.
   PERFORM GET-END-DATE.
   IF BEGIN-DATE GREATER THAN END-DATE
      DISPLAY SCREEN(24,1) WITH NO ADVANCING
      DISPLAY 'THE END DATE GREATER THAN BEGIN DATE REENTER'
      WITH NO ADVANCING
      PERFORM GET-BEGIN-END-DATE.
      DISPLAY SCREEN(23,17) WITH NO ADVANCING.
      DISPLAY ' ' WITH NO ADVANCING.
   GET-BEGIN-DATE.
      DISPLAY SCREEN(23,17) WITH NO ADVANCING.
      DISPLAY 'ENTER BEGIN DATE: YEAR __ MONTH __ DATE __'
      WITH NO ADVANCING.
      PERFORM GET-YEAR.
      PERFORM GET-MONTH.
      PERFORM GET-DAY.
      MOVE INPUT-DATE TO BEGIN-DATE.

GET-END-DATE.
   DISPLAY SCREEN(23,17) WITH NO ADVANCING.
   DISPLAY 'ENTER END DATE: YEAR __ MONTH __ DATE __'
   WITH NO ADVANCING.
   PERFORM GET-YEAR.
   PERFORM GET-MONTH.
   PERFORM GET-DAY.
   MOVE INPUT-DATE TO END-DATE.

GET-YEAR.
   MOVE SPACES TO TEST-VALUE-B.
   DISPLAY SCREEN(23,39) WITH NO ADVANCING.
   DISPLAY ' ' WITH NO ADVANCING.
   DISPLAY SCREEN(24,17) WITH NO ADVANCING.
   DISPLAY 'ENTER LAST TWO DIGITAL '
   WITH NO ADVANCING.
   DISPLAY SCREEN(23,40) WITH NO ADVANCING.
   ACCEPT TEST-VALUE-B.
   IF BEHIND-B NOT EQUAL SPACES
      DISPLAY SCREEN(24,17) WITH NO ADVANCING
      DISPLAY 'ENTER TOO LONG '
      WITH NO ADVANCING
   PERFORM GET-YEAR.
   MOVE FRONT-B TO I-YEAR.
   SUBTRACT SYS-YEAR FROM I-YEAR GIVING CHECK-DATA.
   IF CHECK-DATA LESS THAN 2 AND CHECK-DATA
      GREATER THAN -1 MOVE FRONT-B TO I-YEAR
   ELSE
DISPLAY SCREEN(24,17) WITH NO ADVANCING.
DISPLAY 'YEAR NOT WITHIN PROPER RANGE'
     WITH NO ADVANCING.
PERFORM GET-YEAR.
DISPLAY SCREEN(24,17) WITH NO ADVANCING.
DISPLAY '   ' WITH NO ADVANCING.
GET-MONTH.
MOVE SPACES TO TEST-VALUE-B.
DISPLAY SCREEN(23,48) WITH NO ADVANCING.
DISPLAY '   ' WITH NO ADVANCING.
DISPLAY SCREEN(23,49) WITH NO ADVANCING.
ACCEPT TEST-VALUE-B.
IF BEHIND-B NOT EQUAL SPACES
     DISPLAY SCREEN(24,17) WITH NO ADVANCING
     DISPLAY 'ENTER TOO LONG'
     WITH NO ADVANCING.
PERFORM GET-MONTH.
IF FRONT-B NOT GREATER THAN 13 AND FRONT-B GREATER
    THAN 0 MOVE FRONT-B TO I-MONTH
ELSE
     DISPLAY SCREEN(24,17) WITH NO ADVANCING
     DISPLAY 'MONTH NOT WITH PROPER RANGE'
     WITH NO ADVANCING.
PERFORM GET-MONTH.
DISPLAY SCREEN(24,17) WITH NO ADVANCING.
DISPLAY '   ' WITH NO ADVANCING.
GET-DAY.
MOVE SPACES TO TEST-VALUE-B.
DISPLAY SCREEN(23,56) WITH NO ADVANCING.
DISPLAY '   ' WITH NO ADVANCING.
DISPLAY SCREEN(23,57) WITH NO ADVANCING.
ACCEPT TEST-VALUE-B.
IF BEHIND-B NOT EQUAL SPACES
     DISPLAY SCREEN(24,17) WITH NO ADVANCING
     DISPLAY 'ENTER TOO LONG'
     WITH NO ADVANCING.
PERFORM GET-DAY.
IF FRONT-B LESS THAN 32 AND FRONT-B GREATER THAN 0
    MOVE FRONT-B TO I-DAY
ELSE
     DISPLAY SCREEN(24,17) WITH NO ADVANCING
     DISPLAY 'DATE NOT WITH PROPER RANGE'
     WITH NO ADVANCING.
PERFORM GET-DAY.
DISPLAY SCREEN(24,17) WITH NO ADVANCING.
DISPLAY '   ' WITH NO ADVANCING.
PRINT-AUDIT-TRAIL-TITLE.
WRITE PRINT-RECORD FROM COMPANY-TITLE-2
AFTER ADVANCING 4 LINES.
MOVE B-YEAR TO AUDIT-HEADING-YEAR.
MOVE B-MONTH TO AUDIT-HEADING-MONTH.
MOVE B-DAY TO AUDIT-HEADING-DAY.
WRITE PRINT-RECORD FROM AUDIT-HEADING-1.
WRITE PRINT-RECORD FROM AUDIT-HEADING-2.
MOVE E-YEAR TO AUDIT-HEADING-YEAR.
MOVE E-MONTH TO AUDIT-HEADING-MONTH.
MOVE E-DAY TO AUDIT-HEADING-DAY.
WRITE PRINT-RECORD FROM AUDIT-HEADING-1.

PRINT-AUDIT-TRAIL-HEADING.
MOVE PAGE-NUMBER TO AUDIT-HEADING-PAGE.
MOVE TOTAL-PAGE-NUMBER TO
AUDIT-HEADING-TOTAL-PAGE.
IF PAGE-NUMBER GREATER THAN 1
WRITE PRINT-RECORD FROM SPACES
AFTER ADVANCING PAGE.
WRITE PRINT-RECORD FROM AUDIT-HEADING-3
AFTER ADVANCING 2 LINES.
WRITE PRINT-RECORD FROM AUDIT-HEADING-4
AFTER ADVANCING 4 LINES.
WRITE PRINT-RECORD FROM AUDIT-HEADING-5.
WRITE PRINT-RECORD FROM AUDIT-HEADING-6
BEFORE ADVANCING 3 LINES.

GO-TO-BEGIN-DATE.
IF PRE-DATE-OF-ENTRY NOT GREATER B-DAY
READ PRE-ENTERED-ACCOUNT-FILE AT END
MOVE 'Y' TO EOF-PRE-FILE-OR-NOT
ELSE MOVE 'Y' TO FIND-BEGIN-DATE-OK.

PRINT-LEDGER-SHEET-SCREEN.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY SCREEN(22,17) WITH NO ADVANCING.
DISPLAY 'PRINT LEDGER SHEET PROCESSING'
WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.

PRINT-LEDGER-SHEET.
OPEN INPUT ENTERED-ACCOUNT-FILE.
MOVE 1 TO TABLE-INDEX.
MOVE 'N' TO BUILD-TABLE-OK-OR-NOT.
READ ENTERED-ACCOUNT-FILE AT END MOVE 'Y'
TO BUILD-TABLE-OK-OR-NOT.
PERFORM BUILD-LEDGER-TABLE UNTIL
BUILD-TABLE-OK.
CLOSE ENTERED-ACCOUNT-FILE.
SUBTRACT 1 FROM TABLE-INDEX GIVING TABLE-INDEX.
PERFORM PROCESS-LEDGER-ACCOUNT VARYING INDEX-A
FROM 1 BY 1 UNTIL INDEX-A > TABLE-INDEX.
ACCEPT SYS-DATE FROM DATE.

BUILD-LEDGER-TABLE.
MOVE 'N' TO ACCOUNT-EXIST-OR-NOT.
IF TABLE-INDEX GREATER THAN 1
PERFORM CHECK-ACCOUNT-EXIST VARYING INDEX-A FROM
1 BY 1 UNTIL INDEX-A > TABLE-INDEX - 1
OR ACCOUNT-EXIST
ELSE
MOVE ACC-NUMBER TO LEDGER-COUNTER(TABLE-INDEX).
IF NOT ACCOUNT-EXIST MOVE ACC-NUMBER TO
LEDGER-COUNTER(TABLE-INDEX)
ADD 1 TO TABLE-INDEX.
READ ENTERED-ACCOUNT-FILE AT END MOVE
'Y' TO BUILD-TABLE-OK-OR-NOT.
CHECK-ACCOUNT-EXIST.
IF ACC-NUMBER EQUAL LEDGER-COUNTER(INDEX-A)
MOVE 'Y' TO ACCOUNT-EXIST-OR-NOT.
PROCESS-LEDGER-ACCOUNT.
MOVE 0 TO REPEAT-COUNTER.
PERFORM LEDGER-SHEET-PROCESS UNTIL
REPEAT-COUNTER EQUAL 2.
LEDGER-SHEET-PROCESS.
OPEN INPUT ENTERED-ACCOUNT-FILE.
IF REPEAT-COUNTER EQUAL 1
OPEN OUTPUT PRINT-FILE.
MOVE 1 TO PAGE-NUMBER.
MOVE 0 TO LINE-NUMBER.
IF REPEAT-COUNTER EQUAL 1 PERFORM
PRINT-LEDGER-SHEET-TITLE.
ADD 7 TO LINE-NUMBER.
MOVE 0 TO NET-CHANGE.
MOVE 'N' TO EOF-ENTERED-ACC-FILE-OR-NOT.
READ ENTERED-ACCOUNT-FILE AT END MOVE
'Y' TO EOF-ENTERED-ACC-FILE-OR-NOT.
PERFORM PRINT-LEDGER-SHEET-BODY UNTIL
EOF-ENTERED-ACC-FILE.
CLOSE ENTERED-ACCOUNT-FILE.
IF REPEAT-COUNTER EQUAL 1
CLOSE PRINT-FILE.
ADD 1 TO REPEAT-COUNTER.
SUBTRACT 1 FROM PAGE-NUMBER GIVING PAGE-NUMBER.
MOVE PAGE-NUMBER TO TOTAL-PAGE-NUMBER.
PRINT-LEDGER-SHEET-BODY.
IF REPEAT-COUNTER EQUAL 1
PERFORM PRINT-LEDGER-SHEET-HEADING.
ADD 10 TO LINE-NUMBER.
PERFORM PRINT-LEDGER-SHEET-LINE UNTIL
LINE-NUMBER EQUAL 50 OR EOF-ENTERED-ACC-FILE.
IF EOF-ENTERED-ACC-FILE
ADD 12 TO LINE-NUMBER
IF REPEAT-COUNTER EQUAL 1
PERFORM PRINT-NET-CHANGE-LINES.
MOVE 0 TO LINE-NUMBER.
ADD 1 TO PAGE-NUMBER.
APPENDIX C

PRINT-LEDGER-SHEET-HEADING.
MOVE PAGE-NUMBER TO LEDGER-HEADING-PAGE.
MOVE TOTAL-PAGE-NUMBER TO LEDGER-HEADING-TOTAL-PAGE.
IF PAGE-NUMBER GREATER THAN 1
WRITE PRINT-RECORD FROM SPACES AFTER ADVANCING PAGE.
WRITE PRINT-RECORD FROM LEDGER-SHEET-HEADING AFTER ADVANCING 4 LINES.
WRITE PRINT-RECORD FROM LEDGER-SHEET-LINE-2 AFTER ADVANCING 4 LINES.
WRITE PRINT-RECORD FROM LEDGER-SHEET-LINE-3 AFTER ADVANCING 1 LINES.
WRITE PRINT-RECORD FROM LEDGER-SHEET-LINE-4 BEFORE ADVANCING 1 LINES.

PRINT-NET-CHANGE-LINES.
OPEN INPUT GENERAL-LEDGER-FILE.
MOVE LEDGER-COUNTER(INDEX-A) TO G-ACCOUNT-NUMBER.
READ GENERAL-LEDGER-FILE INVALID KEY
DISPLAY SCREEN(24,17) WITH NO ADVANCING
DISPLAY 'ACCOUNT NUMBER ERROR ' WITH NO ADVANCING.
IF NET-CHANGE GREATER THAN 0
MOVE 'CR' TO LEDGER-SHEET-LINES-MARK
MOVE NET-CHANGE TO LEDGER-SHEET-LINES-BALANCE
ELSE
MOVE 'DR' TO LEDGER-SHEET-LINES-MARK
MULTIPLY -1 BY NET-CHANGE
MOVE NET-CHANGE TO LEDGER-SHEET-LINES-BALANCE
MULTIPLY -1 BY NET-CHANGE.
WRITE PRINT-RECORD FROM LEDGER-SHEET-LINE-5 AFTER ADVANCING 3 LINES.
WRITE PRINT-RECORD FROM LEDGER-SHEET-LINE-6 AFTER ADVANCING 4 LINES.
MOVE G-ACCOUNT-NAME TO LEDGER-SHEET-LINE7-NAME.
WRITE PRINT-RECORD FROM LEDGER-SHEET-LINE-7 AFTER ADVANCING 2 LINES.
IF G-CURRENT-TOTAL GREATER THAN 0
MOVE 'CR' TO LEDGER-SHEET-LINE8-MARK
MOVE G-CURRENT-TOTAL TO LEDGER-SHEET-LINE8-VALUE
ELSE
MOVE 'DR' TO LEDGER-SHEET-LINE8-MARK
MULTIPLY -1 BY G-CURRENT-TOTAL
MOVE G-CURRENT-TOTAL TO LEDGER-SHEET-LINE8-VALUE
MULTIPLY -1 BY G-CURRENT-TOTAL.
MOVE 'OPENING BALANCE ' TO LEDGER-SHEET-LINE8-LABEL.
WRITE PRINT-RECORD FROM LEDGER-SHEET-LINE-8 AFTER ADVANCING 3 LINES.
ADD NET-CHANGE TO G-CURRENT-TOTAL.
IF G-CURRENT-TOTAL GREATER THAN 0
MOVE 'CR' TO LEDGER-SHEET-LINE8-MARK
MOVE G-CURRENT-TOTAL TO LEDGER-SHEET-LINE8-VALUE
ELSE
MOVE 'DR' TO LEDGER-SHEET-LINE8-MARK
MULTIPLY -1 BY G-CURRENT-TOTAL.
APPENDIX C

MOVE G-CURRENT-TOTAL TO LEDGER-SHEET-LINE8-VALUE
MULTIPLY -1 BY G-CURRENT-TOTAL.

MOVE 'CLOSING BALANCE ' TO LEDGER-SHEET-LINE8-LABEL.
WRITE PRINT-RECORD FROM LEDGER-SHEET-LINE8
AFTER ADVANCING 1 LINES.

CLOSE GENERAL-LEDGER-FILE.

PRINT-LEDGER-SHEET-LINE.
IF LEDGER-COUNTER(INDEX-A) EQUAL ACC-NUMBER
PERFORM PROCESS-LEDGER-SHEET-LINE.
READ ENTERED-ACCOUNT-FILE AT END
MOVE 'Y' TO EOF-ENTERED-ACC-FILE-OR-NOT.

PROCESS-LEDGER-SHEET-LINE.
MOVE ENTRY-NO TO LEDGER-SHEET-ENTRY-NUMBER.
IF LINE-NUMBER EQUAL 17 AND PAGE-NUMBER EQUAL 1
MOVE ACC-NUMBER TO LEDGER-SHEET-ACCOUNT-NUMBER
ELSE MOVE SPACES TO LEDGER-SHEET-ACCOUNT-NUMBER.
MOVE DATE-OF-ENTRY TO SYS-DATE.
MOVE SYS-YEAR TO LEDGER-SHEET-YEAR.
MOVE SYS-MONTH TO LEDGER-SHEET-MONTH.
MOVE SYS-DAY TO LEDGER-SHEET-DAY.
IF SOURCE-CODE EQUAL 1 MOVE 'A/R'
TO LEDGER-SHEET-SOURCE-CODE
ELSE IF SOURCE-CODE EQUAL 2 MOVE 'A/P'
TO LEDGER-SHEET-SOURCE-CODE
ELSE IF SOURCE-CODE EQUAL 3 MOVE 'P/R'
TO LEDGER-SHEET-SOURCE-CODE
ELSE IF SOURCE-CODE EQUAL 4 MOVE 'ADJ'
TO LEDGER-SHEET-SOURCE-CODE.

ADD ACCOUNT-VALUE TO NET-CHANGE.
MOVE SPACES TO LEDGER-SHEET-ACCOUNT-CR.
MOVE SPACES TO LEDGER-SHEET-ACCOUNT-DR.
IF ACCOUNT-VALUE GREATER THAN 0 MOVE
ACCOUNT-VALUE TO LEDGER-SHEET-ACCOUNT-CR
ELSE
MULTIPLY -1 BY ACCOUNT-VALUE
MOVE ACCOUNT-VALUE TO LEDGER-SHEET-ACCOUNT-DR
MULTIPLY -1 BY ACCOUNT-VALUE.

IF REPEAT-COUNTER EQUAL 1
WRITE PRINT-RECORD FROM LEDGER-SHEET-LINE.
ADD 1 TO LINE-NUMBER.

PRINT-LEDGER-SHEET-TITLE.
WRITE PRINT-RECORD FROM SPACES AFTER ADVANCING PAGE.
PERFORM PRINT-COMPANY-TITLE.
MOVE SCREEN-SYS-DATE TO LEDGER-SHEET-TITLE-DATE.
WRITE PRINT-RECORD FROM LEDGER-SHEET-TITLE-LINE
AFTER ADVANCING 2 LINES.

PRINT-TRIAL-BALANCE-SCREEN.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY SCREEN(22,17) WITH NO ADVANCING.
DISPLAY 'PRINT TRIAL BALANCE PROCESSING'
WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.

PRINT-TRIAL-BALANCE.
OPEN INPUT ENTERED-ACCOUNT-FILE.
MOVE 1 TO TABLE-INDEX.
MOVE 'N' TO BUILD-TABLE-OK-OR-NOT.
READ ENTERED-ACCOUNT-FILE AT END MOVE 'Y'
TO BUILD-TABLE-OK-OR-NOT.
PERFORM BUILD-TRIAL-BALANCE-TABLE UNTIL
BUILD-TABLE-OK.
CLOSE ENTERED-ACCOUNT-FILE.
SUBTRACT 1 FROM TABLE-INDEX GIVING TABLE-INDEX.
OPEN INPUT GENERAL-LEDGER-FILE
OUTPUT PRINT-FILE.
MOVE 0 TO DR-TOTAL.
MOVE 0 TO CR-TOTAL.
MOVE 1 TO PAGE-NUMBER.
MOVE 0 TO LINE-NUMBER.
MOVE 'N' TO EOF-G-L-FILE-OR-NOT.
READ GENERAL-LEDGER-FILE NEXT AT END
MOVE 'Y' TO EOF-G-L-FILE-OR-NOT.
PERFORM PROCESS-TRIAL-BALANCE UNTIL EOF-G-L-FILE.
PERFORM PRINT-TRIAL-TOTAL-LINE.
CLOSE GENERAL-LEDGER-FILE
PRINT-FILE.
ACCEPT SYS-DATE FROM DATE.
BUILD-TRIAL-BALANCE-TABLE.
MOVE 'N' TO ACCOUNT-EXIST-OR-NOT.
IF TABLE-INDEX GREATER THAN 1
PERFORM CHECK-ENTERED-ACCOUNT-EXIST VARYING
INDEX-A FROM 1 BY 1 UNTIL INDEX-A >
TABLE-INDEX - 1 OR ACCOUNT-EXIST
ELSE
IF UPDATE-CODE EQUAL 'N'
MOVE ACC-NUMBER TO LEDGER-COUNTER(TABLE-INDEX)
ADD 1 TO TABLE-INDEX.
IF TABLE-INDEX NOT EQUAL 1
IF NOT ACCOUNT-EXIST AND UPDATE-CODE EQUAL 'N' MOVE
ACC-NUMBER TO TRIAL-TABLE-ACCOUNT(TABLE-INDEX)
MOVE ACCOUNT-VALUE TO
TRIAL-TABLE-VALUE(TABLE-INDEX)
ADD 1 TO TABLE-INDEX.
READ ENTERED-ACCOUNT-FILE AT END MOVE
'Y' TO BUILD-TABLE-OK-OR-NOT.
CHECK-ENTERED-ACCOUNT-EXIST.
IF ACC-NUMBER EQUAL TRIAL-TABLE-ACCOUNT(INDEX-A)
MOVE 'Y' TO ACCOUNT-EXIST-OR-NOT
IF UPDATE-CODE EQUAL 'N'
ADD ACCOUNT-VALUE TO TRIAL-TABLE-VALUE(INDEX-A).
PROCESS-TRIAL-BALANCE.
IF PAGE-NUMBER EQUAL 1 AND LINE-NUMBER EQUAL 0
PERFORM PRINT-TRIAL-BALANCE-TITLE
PERFORM PRINT-TRIAL-BALANCE-HEADING
ADD 14 TO LINE-NUMBER.
IF PAGE-NUMBER NOT EQUAL 1 AND LINE-NUMBER EQUAL 0
PERFORM PRINT-TRIAL-BALANCE-HEADING
ADD 11 TO LINE-NUMBER.
PERFORM PRINT-TRIAL-BALANCE-BODY UNTIL
LINE-NUMBER GREATER THAN 50 OR
EOF-G-L-FILE.
ADD 1 TO PAGE-NUMBER.
MOVE 0 TO LINE-NUMBER.

PRINT-TRIAL-BALANCE-BODY.
IF G-ACCOUNT-TYPE EQUAL 0
PERFORM PRINT-TRIAL-BALANCE-LINE.
READ GENERAL-LEDGER-FILE NEXT AT END
MOVE 'Y' TO EOF-G-L-FILE-OR-NOT.

PRINT-TRIAL-BALANCE-LINE.
MOVE 'N' TO ACCOUNT-EXIST-OR-NOT.
PERFORM ADD-ACCOUNT-VALUE VARYING INDEX-A
FROM 1 BY 1 UNTIL INDEX-A >
TABLE-INDEX OR ACCOUNT-EXIST.
MOVE G-ACCOUNT-NUMBER TO TRIAL-LINE5-ACC-NUMBER.
MOVE G-ACCOUNT-NAME TO TRIAL-LINE5-ACC-NAME.
IF G-CURRENT-TOTAL GREATER THAN 0 MOVE
G-CURRENT-TOTAL TO TRIAL-LINE5-ACCOUNT-TEMPLE
MOVE TRIAL-LINE5-ACCOUNT-TEMPLE TO
TRIAL-LINE5-ACCOUNT-CR
MOVE SPACES TO TRIAL-LINE5-ACCOUNT-DR
ADD G-CURRENT-TOTAL TO CR-TOTAL
ELSE
ADD G-CURRENT-TOTAL TO DR-TOTAL
MULTIPLY -1 BY G-CURRENT-TOTAL
MOVE G-CURRENT-TOTAL TO
TRIAL-LINE5-ACCOUNT-TEMPLE
MOVE TRIAL-LINE5-ACCOUNT-TEMPLE TO
TRIAL-LINE5-ACCOUNT-DR
MOVE SPACES TO TRIAL-LINE5-ACCOUNT-CR.
IF G-CURRENT-TOTAL EQUAL 0
MOVE ' 0.00' TO TRIAL-LINE5-ACCOUNT-CR
MOVE SPACES TO TRIAL-LINE5-ACCOUNT-DR.
WRITE PRINT-RECORD FROM TRIAL-BALANCE-LINE-5.
ADD 1 TO LINE-NUMBER.

ADD-ACCOUNT-VALUE.
IF TRIAL-TABLE-ACCOUNT(INDEX-A) EQUAL
G-ACCOUNT-NUMBER
ADD TRIAL-TABLE-VALUE(INDEX-A) TO G-CURRENT-TOTAL
MOVE 'Y' TO ACCOUNT-EXIST-OR-NOT.

PRINT-TRIAL-TOTAL-LINE.
WRITE PRINT-RECORD FROM TRIAL-BALANCE-LINE-6
AFTER ADVANCING 1 LINES.
MOVE CR-TOTAL TO TRIAL-LINE7-CR-VALUE.
MOVE DR-TOTAL TO TRIAL-LINE7-DR-VALUE.
WRITE PRINT-RECORD FROM TRIAL-BALANCE-LINE-7
APPENDIX C

AFTER ADVANCING 4 LINES.

PRINT-TRIAL-BALANCE-TITLE.
WRITE PRINT-RECORD FROM SPACES AFTER ADVANCING PAGE.
MOVE SCREEN-SYS-DATE TO TRIAL-BALANCE-TITLE-DATE.
WRITE PRINT-RECORD FROM TRIAL-BALANCE-TITLE
AFTER ADVANCING 4 LINES.

PRINT-TRIAL-BALANCE-HEADING.
IF PAGE-NUMBER GREATER THAN 1 WRITE
PRINT-RECORD FROM SPACES AFTER ADVANCING PAGE.
MOVE PAGE-NUMBER TO TRIAL-LINE1-PAGE-NUMBER.
WRITE PRINT-RECORD FROM TRIAL-BALANCE-LINE-1
AFTER ADVANCING 4 LINES.
WRITE PRINT-RECORD FROM TRIAL-BALANCE-LINE-2
AFTER ADVANCING 4 LINES.
WRITE PRINT-RECORD FROM TRIAL-BALANCE-LINE-3
AFTER ADVANCING 1 LINES.
WRITE PRINT-RECORD FROM TRIAL-BALANCE-LINE-4
BEFORE ADVANCING 1 LINES.

PRINT-INCOME-STATEMENT-SCREEN.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY SCREEN(22,17) WITH NO ADVANCING.
DISPLAY 'PRINT INCOME STATEMENT PROCESSING'
WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.

PRINT-INCOME-STATEMENT.
MOVE 1 TO PAGE-NUMBER.
MOVE 0 TO LINE-NUMBER.
MOVE SPACES TO PRE-ACC-TYPE.
MOVE 4 TO INDENT-LEVEL-INDEX.
MOVE 8 TO TOTAL-LEVEL-INDEX.
OPEN INPUT GENERAL-LEDGER-FILE
OUTPUT PRINT-FILE.
MOVE 'N' TO EOF-G-L-FILE-OR-NOT.
PERFORM CLEAN-TOTAL-TABLE VARYING INDEX-A
FROM 1 BY 1 UNTIL INDEX-A EQUAL 9.
MOVE 'N' TO TAKE-TYPE-OK-OR-NOT.
PERFORM GET-EXPENSE-ACCOUNT UNTIL TAKE-TYPE-OK.
PERFORM INCOME-STATEMENT-PROCESS UNTIL EOF-G-L-FILE.
CLOSE GENERAL-LEDGER-FILE
PRINT-FILE.

GET-EXPENSE-ACCOUNT.
READ GENERAL-LEDGER-FILE NEXT AT END MOVE 'Y'
TO TAKE-TYPE-OK-OR-NOT.
IF G-ACCOUNT-NUMBER EQUAL '300000' MOVE 'Y' TO TAKE-TYPE-OK-OR-NOT.

INCOME-STATEMENT-PROCESS.
IF PAGE-NUMBER EQUAL 1 PERFORM
PRINT-INCOME-STATEMENT-TITLE.
PERFORM PRINT-INCOME-STATEMENT-HEADING.
IF G-ACCOUNT-NUMBER NOT EQUAL '300000'
APPENDIX C

READ GENERAL-LEDGER-FILE NEXT AT END MOVE 'Y' TO EOF-G-L-FILE-OR-NOT.
PERFORM GET-INCOME-STATEMENT-BODY UNTIL EOF-G-L-FILE OR LINE-NUMBER GREATER THAN 55.
ADD 1 TO PAGE-NUMBER.
MOVE 0 TO LINE-NUMBER.

GET-INCOME-STATEMENT-BODY.
PERFORM DECIDE-TOTAL-LEVEL.
PERFORM DECIDE-INDENT-LEVEL.
IF G-ACCOUNT-TYPE EQUAL '0' OR G-ACCOUNT-TYPE EQUAL '2'
PERFORM TOTAL-AMOUNT-PROCESS
PERFORM DECIDE-DOLLAR-SIGN.
IF G-ACCOUNT-TYPE EQUAL '2' AND PRE-ACC-TYPE EQUAL '2'
WRITE PRINT-RECORD FROM SPACES
ADD 1 TO LINE-NUMBER.
IF G-ACCOUNT-TYPE EQUAL '3' AND PRE-ACC-TYPE EQUAL '2'
WRITE PRINT-RECORD FROM SPACES
ADD 1 TO LINE-NUMBER.
WRITE PRINT-RECORD FROM BALANCE-SHEET-LINE
ADD 1 TO LINE-NUMBER.
MOVE SPACES TO BALANCE-SHEET-LINE.
MOVE G-ACCOUNT-TYPE TO PRE-ACC-TYPE.
READ GENERAL-LEDGER-FILE NEXT AT END MOVE 'Y' TO EOF-G-L-FILE-OR-NOT.

PRINT-INCOME-STATEMENT-TITLE.
WRITE PRINT-RECORD FROM COMPANY-TITLE-2
AFTER ADVANCING 3 LINES.
ADD 3 TO LINE-NUMBER.

PRINT-INCOME-STATEMENT-HEADING.
IF PAGE-NUMBER GREATER THAN 1
WRITE PRINT-RECORD FROM SPACES
AFTER ADVANCING PAGE.
MOVE PAGE-NUMBER TO INCOME-STATEMENT-PAGE-NUMBER.
MOVE SCREEN-SYS-DATE TO INCOME-STATEMENT-DATE.
WRITE PRINT-RECORD FROM INCOME-STATEMENT-HEADING-1
AFTER ADVANCING 2 LINES.
WRITE PRINT-RECORD FROM INCOME-STATEMENT-HEADING-2
AFTER ADVANCING 2 LINES.
WRITE PRINT-RECORD FROM SPACES AFTER ADVANCING 4 LINES.
ADD 8 TO LINE-NUMBER.

PRINT-BALANCE-SHEET-SCREEN.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY SCREEN(22,17) WITH NO ADVANCING.
DISPLAY 'PRINT BALANCE SHEET PROCESSING'
WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.

PRINT-BALANCE-SHEET.
APPENDIX C

MOVE 1 TO PAGE-NUMBER.
MOVE 0 TO LINE-NUMBER.
MOVE SPACES TO PRE-ACC-TYPE.
MOVE 4 TO INDENT-LEVEL-INDEX.
MOVE 8 TO TOTAL-LEVEL-INDEX.
OPEN INPUT GENERAL-LEDGER-FILE
OUTPUT PRINT-FILE.
MOVE 'N' TO EOF-G-L-FILE-OR-NOT.
PERFORM CLEAN-TOTAL-TABLE VARYING INDEX-A
FROM 1 BY 1 UNTIL INDEX-A EQUAL 9.
PERFORM BALANCE-SHEET-PROCESS UNTIL EOF-G-L-FILE.
CLOSE GENERAL-LEDGER-FILE
PRINT-FILE.

CLEAN-TOTAL-TABLE.
MOVE SPACES TO TOTAL-AMOUNT(INDEX-A).

BALANCE-SHEET-PROCESS.
IF PAGE-NUMBER EQUAL 1 PERFORM
PRINT-BALANCE-SHEET-TITLE.
PERFORM PRINT-BALANCE-SHEET-HEADING.
READ GENERAL-LEDGER-FILE NEXT AT END MOVE 'Y'
TO EOF-G-L-FILE-OR-NOT.
PERFORM GET-BALANCE-SHEET-BODY UNTIL EOF-G-L-FILE
OR LINE-NUMBER GREATER THAN 55.
ADD 1 TO PAGE-NUMBER.
MOVE 0 TO LINE-NUMBER.

GET-BALANCE-SHEET-BODY.
IF G-ACCOUNT-TYPE EQUAL '1'
MOVE 4 TO INDENT-LEVEL-INDEX
MOVE 8 TO TOTAL-LEVEL-INDEX
PERFORM CLEAN-TOTAL-TABLE VARYING INDEX-A
FROM 1 BY 1 UNTIL INDEX-A EQUAL 9.
PERFORM DECIDE-TOTAL-LEVEL.
PERFORM DECIDE-INDENT-LEVEL.
IF G-ACCOUNT-TYPE EQUAL '0' OR
G-ACCOUNT-TYPE EQUAL '2'
PERFORM TOTAL-AMOUNT-PROCESS
PERFORM DECIDE-DOLLAR-SIGN.
IF G-ACCOUNT-TYPE EQUAL '2' AND
PRE-ACC-TYPE EQUAL '2'
WRITE PRINT-RECORD FROM SPACES
ADD 1 TO LINE-NUMBER.
IF G-ACCOUNT-TYPE EQUAL '3' AND
PRE-ACC-TYPE EQUAL '2'
WRITE PRINT-RECORD FROM SPACES
ADD 1 TO LINE-NUMBER.
WRITE PRINT-RECORD FROM BALANCE-SHEET-LINE
ADD 1 TO LINE-NUMBER.
MOVE SPACES TO BALANCE-SHEET-LINE.
MOVE G-ACCOUNT-TYPE TO PRE-ACC-TYPE.
READ GENERAL-LEDGER-FILE NEXT AT END MOVE 'Y'
TO EOF-G-L-FILE-OR-NOT.
IF G-ACCOUNT-NUMBER EQUAL '300000' MOVE 'Y'
TO EOF-G-L-FILE-OR-NOT.
APPENDIX C

DECIDE-TOTAL-LEVEL.
   IF G-ACCOUNT-TYPE EQUAL '2'
      ADD 1 TO TOTAL-LEVEL-INDEX
   ELSE
      IF G-ACCOUNT-TYPE EQUAL '3'
         ADD -1 TO TOTAL-LEVEL-INDEX
      ELSE
         IF G-ACCOUNT-TYPE EQUAL '1' AND
            PRE-ACC-TYPE NOT EQUAL '1'
            ADD -1 TO TOTAL-LEVEL-INDEX.

DECIDE-INDENT-LEVEL.
   IF G-ACCOUNT-TYPE EQUAL '3' AND PRE-ACC-TYPE
      NOT EQUAL '3' ADD 1 TO INDENT-LEVEL-INDEX
   ELSE
      IF G-ACCOUNT-TYPE EQUAL '3' AND PRE-ACC-TYPE
         NOT EQUAL '1' ADD 1 TO INDENT-LEVEL-INDEX
      ELSE IF G-ACCOUNT-TYPE EQUAL '2'
         ADD -1 TO INDENT-LEVEL-INDEX
      ELSE IF G-ACCOUNT-TYPE '2' AND PRE-ACC-TYPE
         EQUAL '2' ADD 1 TO INDENT-LEVEL-INDEX
      ELSE IF G-ACCOUNT-TYPE EQUAL '0'
         AND PRE-ACC-TYPE EQUAL '3'
         ADD 1 TO INDENT-LEVEL-INDEX.
   IF G-ACCOUNT-TYPE EQUAL '1'
      WRITE PRINT-RECORD FROM SPACES
      AFTER ADVANCING 3 LINES
      ADD 3 TO LINE-NUMBER
      MOVE 3 TO INDENT-LEVEL-INDEX.
   IF INDENT-LEVEL-INDEX EQUAL 1
      MOVE G-ACCOUNT-NAME TO INDENT-1
      MOVE INDENT-1-OUTPUT TO BALANCE-SHEET-ACC-NAME
   ELSE IF INDENT-LEVEL-INDEX EQUAL 2
      MOVE G-ACCOUNT-NAME TO INDENT-2
      MOVE INDENT-2-OUTPUT TO BALANCE-SHEET-ACC-NAME
   ELSE
      IF INDENT-LEVEL-INDEX EQUAL 3
         MOVE G-ACCOUNT-NAME TO INDENT-3
         MOVE INDENT-3-OUTPUT TO BALANCE-SHEET-ACC-NAME
      ELSE
         IF INDENT-LEVEL-INDEX EQUAL 4
            MOVE G-ACCOUNT-NAME TO INDENT-4
            MOVE INDENT-4-OUTPUT TO BALANCE-SHEET-ACC-NAME
         ELSE
            IF INDENT-LEVEL-INDEX EQUAL 5
               MOVE G-ACCOUNT-NAME TO INDENT-5
               MOVE INDENT-5-OUTPUT TO BALANCE-SHEET-ACC-NAME
            ELSE
               IF INDENT-LEVEL-INDEX EQUAL 6
                  MOVE G-ACCOUNT-NAME TO INDENT-6
                  MOVE INDENT-6-OUTPUT TO BALANCE-SHEET-ACC-NAME
               ELSE
                  IF INDENT-LEVEL-INDEX EQUAL 7
                     MOVE G-ACCOUNT-NAME TO INDENT-7
                     MOVE INDENT-7-OUTPUT TO BALANCE-SHEET-ACC-NAME.
TOTAL-AMOUNT-PROCESS.
   IF G-ACCOUNT-TYPE EQUAL '0'
      PERFORM REGULAR-ACCOUNT-PROCESS VARYING
         INDEX-A FROM 8 BY -1 UNTIL INDEX-A EQUAL
         TOTAL-LEVEL-INDEX - 1
   IF G-CURRENT-TOTAL LESS THAN 0
      MOVE '<' TO BALANCE-SHEET-MARK-1
      MOVE '>' TO BALANCE-SHEET-MARK-2
      MULTIPLY -1 BY G-CURRENT-TOTAL
      MOVE G-CURRENT-TOTAL TO BALANCE-SHEET-ACC-VALUE
   ELSE
      MOVE G-CURRENT-TOTAL TO BALANCE-SHEET-ACC-VALUE.
   IF G-ACCOUNT-TYPE EQUAL '2'
      IF TOTAL-AMOUNT(TOTAL-LEVEL-INDEX) LESS THAN 0
         MOVE '<' TO BALANCE-SHEET-MARK-3
         MOVE '>' TO BALANCE-SHEET-MARK-4
         MULTIPLY -1 BY TOTAL-AMOUNT(TOTAL-LEVEL-INDEX)
         MOVE TOTAL-AMOUNT(TOTAL-LEVEL-INDEX) TO
         BALANCE-SHEET-TOTAL
         MULTIPLY -1 BY TOTAL-AMOUNT(TOTAL-LEVEL-INDEX)
      ELSE
         MOVE TOTAL-AMOUNT(TOTAL-LEVEL-INDEX) TO
         BALANCE-SHEET-TOTAL.
      IF G-ACCOUNT-TYPE EQUAL '2'
         PERFORM SET-TOTAL-ZERO-PROCESS VARYING INDEX-A
         FROM 1 BY 1 UNTIL INDEX-A EQUAL TOTAL-LEVEL-INDEX.
   PRINT-BALANCE-SHEET-TITLE.
      WRITE PRINT-RECORD FROM COMPANY-TITLE-2
         AFTER ADVANCING 3 LINES.
      ADD 3 TO LINE-NUMBER.
   PRINT-BALANCE-SHEET-HEADING.
      IF PAGE-NUMBER GREATER THAN 1
         WRITE PRINT-RECORD FROM SPACES
         AFTER ADVANCING PAGE.
         MOVE PAGE-NUMBER TO BALANCE-SHEET-PAGE-NUMBER.
         MOVE SCREEN-SYS-DATE TO BALANCE-SHEET-HEADING-DATE.
         WRITE PRINT-RECORD FROM BALANCE-SHEET-HEADING-1
            AFTER ADVANCING 2 LINES.
         WRITE PRINT-RECORD FROM BALANCE-SHEET-HEADING-2
            AFTER ADVANCING 2 LINES.
         WRITE PRINT-RECORD FROM SPACES AFTER ADVANCING
            4 LINES.
      ADD 8 TO LINE-NUMBER.
   DECIDE-DOLLAR-SIGN.
      IF G-ACCOUNT-TYPE EQUAL '0' AND PRE-ACC-TYPE EQUAL
         '3' MOVE '$' TO BALANCE-DOLLAR-SIGN-1
      ELSE
         IF G-ACCOUNT-TYPE EQUAL '2' MOVE '$' TO
            BALANCE-DOLLAR-SIGN-2.
   SET-TOTAL-ZERO-PROCESS.
      MOVE SPACES TO TOTAL-AMOUNT(INDEX-A).
REGULAR-ACCOUNT-PROCESS.
ADD G-CURRENT-TOTAL TO TOTAL-AMOUNT(INDEX-A).
IDENTIFICATION DIVISION.

PROGRAM-ID.
UPDAT

AUTHOR.
CCW

DATE-WRITTEN.
15-MAR-1984

ENVIRONMENT DIVISION.

INPUT-OUTPUT SECTION.

FILE-CONTROL.

SELECT ENTERED-ACCOUNT-FILE,
ASSIGN TO DSK,
RECORDING MODE IS ASCII.

SELECT TEMP-ENTERED-ACCOUNT-FILE,
ASSIGN TO DSK,
RECORDING MODE IS ASCII.

SELECT PRE-ENTERED-ACCOUNT-FILE,
ASSIGN TO DSK,
RECORDING MODE IS ASCII.

SELECT GENERAL-LEDGER-FILE
ASSIGN TO DSK
ORGANIZATION IS INDEXED
ACCESS MODE IS DYNAMIC
RECORD KEY IS G-ACCOUNT-NUMBER
RECORDING MODE IS ASCII.

DATA DIVISION.

FILE SECTION.

FD GENERAL-LEDGER-FILE,
BLOCK CONTAINS 128 RECORDS,
VALUE OF ID IS "GENFILINX".

01 GENERAL-LEDGER-RECORD.

  05 G-ACCOUNT-NUMBER PIC 9(6).
  05 G-ACCOUNT-NAME PIC X(30).
  05 G-ACCOUNT-TYPE PIC X.
  05 G-CURRENT-TOTAL PIC S99999999V99.
  05 G-YEAR-TO-NOW-TOTAL PIC S99999999V99.
  05 G-PREVIOUS-YEAR-TOTAL PIC S99999999V99.

FD ENTERED-ACCOUNT-FILE,
BLOCK CONTAINS 128 RECORDS,
LABEL RECORDS ARE STANDARD,
APPENDIX C
VALUE OF ID IS 'ACCOUNTFILE'.

01 ENTERED-ACCOUNT-RECORD.
  05 ENTRY-NO PIC 999999.
  05 ACC-NO.
    08 CATAGORY-NO PIC 9.
    08 SUBCATAGORY-NO PIC 999999.
  05 ACC-NAME PIC X(30).
  05 SOURCE-CODE PIC 9.
  05 ACCOUNT-VALUE PIC S99999999V99.
  05 DATE-OF-ENTRY PIC XXXXXX.
  05 UPDATE-CODE PIC X.

FD TEMP-ENTERED-ACCOUNT-FILE
BLOCK CONTAINS 128 RECORDS,
LABEL RECORDS ARE STANDARD,
VALUE OF ID IS 'TEMPACTMP'.

01 TEMP-ENTERED-ACCOUNT-RECORD.
  05 T-ENTRY-NO PIC 999999.
  05 T-ACC-NO.
    08 T-CATAGORY-NO PIC 9.
    08 T-SUBCATAGORY-NO PIC 999999.
  05 T-ACC-NAME PIC X(30).
  05 T-SOURCE-CODE PIC 9.
  05 T-ACCOUNT-VALUE PIC S99999999V99.
  05 T-DATE-OF-ENTRY PIC XXXXXX.
  05 T-UPDATE-CODE PIC X.

FD PRE-ENTERED-ACCOUNT-FILE,
BLOCK CONTAINS 128 RECORDS,
LABEL RECORDS ARE STANDARD,
VALUE OF ID IS 'PREVI FIL'.

01 PRE-ENTERED-ACCOUNT-RECORD.
  05 PRE-ENTRY-NO PIC 999999.
  05 PRE-ACC-NO.
    08 PRE-CATAGORY-NO PIC 9.
    08 PRE-SUBCATAGORY-NO PIC 999999.
  05 PRE-ACC-NAME PIC X(30).
  05 PRE-SOURCE-CODE PIC 9.
  05 PRE-ACCOUNT-VALUE PIC S99999999V99.
  05 PRE-DATE-OF-ENTRY PIC XXXXXX.
  05 PRE-UPDATE-CODE PIC X.

*---------------------------------------------------------------
WORKING-STORAGE SECTION.

01 TEST-VALUE-A.
  05 FRONT-A PIC X.
  05 BEHIND-A PIC X(9).

01 UPDATE-TYPE PIC X.

01 CR-TOTAL PIC S99999999V99.

01 DR-TOTAL PIC S99999999V99.
APPENDIX C

01 TIMER PIC 99999.
01 TOTAL-INCOME PIC S99999999V99.
77 NOT-ACCOUNT-EXIST PIC X VALUE 'N'.
     88 ACCOUNT-NOT-EXIST VALUE 'Y'.
77 NOT-EOF-ENTERED-ACCOUNT-FILE PIC X VALUE 'N'.
     88 EOF-ENTERED-ACCOUNT-FILE VALUE 'Y'.
77 NOT-EOF-TEMP-FILE PIC X VALUE 'N'.
     88 EOF-TEMP-FILE VALUE 'Y'.
77 NOT-EOF-PRE-ENTERED-ACC-FILE PIC X VALUE 'N'.
     88 EOF-PRE-ENTERED-ACC-FILE VALUE 'Y'.
77 NOT-EOF-GENERAL-LEDGER-FILE PIC X VALUE 'N'.
     88 EOF-GENERAL-LEDGER-FILE VALUE 'Y'.
77 NOT-UPDATE-YET PIC X VALUE 'N'.
     88 NOT-UPDATE VALUE 'Y'.
77 EXIT-UPDATE-ACC-OR-NOT PIC X VALUE 'N'.
     88 EXIT-UPDATE-ACCOUNT VALUE 'Y'.
77 TAKE-TYPE PIC X VALUE 'N'.
     88 TAKE-TYPE-OK VALUE 'Y'.

LINKAGE SECTION.

01 SCREEN-TABLE.
     05 SCREEN-ROWS
     10 SCREEN OCCURS 24 TIMES.
     PIC X(4)
     USAGE IS DISPLAY-7
     OCCURS 80 TIMES.

01 REVERSE-VIDEO PIC XX DISPLAY-7.
01 EXIT-REVERSE PIC XX DISPLAY-7.
01 CLEAR-SCREEN PIC XX DISPLAY-7.
01 RING-BELL PIC X DISPLAY-7.

*---------------------------------------------------------------------
PROCEDURE DIVISION USING SCREEN-TABLE CLEAR-SCREEN RING-BELL
     REVERSE-VIDEO EXIT-REVERSE.
*---------------------------------------------------------------------

MAIN.

MOVE 'N' TO EXIT-UPDATE-ACC-OR-NOT.
PERFORM SELECT-UPDATE-TYPE-SCREEN.
PERFORM ACCOUNT-UPDATE-PROCESS
     UNTIL EXIT-UPDATE-ACCOUNT.

EXIT PROGRAM.
ACCOUNT-UPDATE-PROCESS.

MOVE 'N' TO TAKE-TYPE.
PERFORM TAKE-TYPE-OF-UPDATE UNTIL TAKE-TYPE-OK.
IF UPDATE-TYPE = '1'
  PERFORM UPDATE-G-L-FILE-SCREEN
  PERFORM UPDATE-GENERAL-LEDGER-FILE
ELSE IF UPDATE-TYPE = '2'
  PERFORM CLEAN-REVENUE-AND-EXP-SCREEN
  PERFORM CLEAN-REVENUES-AND-EXPENSE
  DISPLAY SCREEN(22,17) WITH NO ADVANCING
  DISPLAY ' ' WITH NO ADVANCING
ELSE IF UPDATE-TYPE = '3'
  PERFORM CLEAN-OLD-ENTRIES-SCREEN
  PERFORM CLEAN-OLD-ENTRIES
ELSE IF UPDATE-TYPE = '4'
  PERFORM END-OF-YEAR-PROCESS-SCREEN
  PERFORM END-OF-YEAR-PROCESS
ELSE IF UPDATE-TYPE = '*'
  MOVE 'Y' TO EXIT-UPDATE-ACC-OR-NOT.

SELECT-UPDATE-TYPE-SCREEN.
DISPLAY CLEAR-SCREEN WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY SCREEN(1,28) WITH NO ADVANCING.
DISPLAY 'SCREEN - 1.2' WITH NO ADVANCING.
DISPLAY SCREEN(2,22) WITH NO ADVANCING.
DISPLAY 'ACCOUNT UPDATE SELECTION' WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY SCREEN (4,17) WITH NO ADVANCING.
DISPLAY 'SELECT ONE OF THE FOLLOWING TYPE TO PROCESS'

WITH NO ADVANCING.

DISPLAY SCREEN (5,31) WITH NO ADVANCING.
DISPLAY 'ENTER SELECTION: _' WITH NO ADVANCING.
  DISPLAY SCREEN(7,16) WITH NO ADVANCING.
  DISPLAY '========================================='
  WITH NO ADVANCING.
  DISPLAY SCREEN(9,16) WITH NO ADVANCING.
  DISPLAY ' * : FOR EXIT ACCOUNT UPDATE PROCESS '
  WITH NO ADVANCING.
  DISPLAY SCREEN(11,16) WITH NO ADVANCING.
  DISPLAY ' 1 : UPDATE GENERAL LEDGER FILE '
  WITH NO ADVANCING.
  DISPLAY SCREEN(13,16) WITH NO ADVANCING.
  DISPLAY ' 2 : CLEAN REVENUES AND EXPENSE '
  WITH NO ADVANCING.
  DISPLAY SCREEN(15,16) WITH NO ADVANCING.
  DISPLAY ' 3 : CLEAN OLD ENTRIES '
  WITH NO ADVANCING.
  DISPLAY SCREEN(17,16) WITH NO ADVANCING.
  DISPLAY ' 4 : END OF YEAR PROCESS '
  WITH NO ADVANCING.
  DISPLAY SCREEN(19,16) WITH NO ADVANCING.
  DISPLAY '========================================='
TAKE-TYPE-OF-UPDATE.
  DISPLAY RING-BELL WITH NO ADVANCING.
  DISPLAY SCREEN(5,49) WITH NO ADVANCING.
  DISPLAY ' ' WITH NO ADVANCING.
  DISPLAY SCREEN(5,49) WITH NO ADVANCING.
  ACCEPT TEST-VALUE-A.
  IF BEHIND-A NOT EQUAL SPACES PERFORM ERROR-HANDLE-A
  ELSE MOVE FRONT-A TO UPDATE-TYPE.
  IF UPDATE-TYPE EQUAL '1' OR EQUAL '2' OR
  EQUAL '3' OR EQUAL '4' OR EQUAL 'A'
    MOVE 'Y' TO TAKE-TYPE
  ELSE
    PERFORM ERROR-HANDLE-A.
  DISPLAY SCREEN(22,7) WITH NO ADVANCING.
  DISPLAY ' ' WITH NO ADVANCING.
  DISPLAY ' ' WITH NO ADVANCING.
ERROR-HANDLE-A.
  DISPLAY SCREEN(22,7) WITH NO ADVANCING.
  DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
  DISPLAY 'ERROR MESSAGE::' WITH NO ADVANCING.
  DISPLAY EXIT-REVERSE WITH NO ADVANCING.
  DISPLAY 'INVALID RESPONSE . PLEASE REENTER ' WITH NO ADVANCING.
  DISPLAY RING-BELL WITH NO ADVANCING.
  DISPLAY SCREEN(5,49) WITH NO ADVANCING.
  DISPLAY ' ' WITH NO ADVANCING.
  DISPLAY SCREEN(5,49) WITH NO ADVANCING.
  PERFORM TAKE-TYPE-OF-UPDATE.

UPDATE-G-L-FILE-SCREEN.
  DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
  DISPLAY SCREEN(22,17) WITH NO ADVANCING.
  DISPLAY 'UPDATE GENERAL LEDGER FILE PROCESSING'
    WITH NO ADVANCING.
  DISPLAY EXIT-REVERSE WITH NO ADVANCING.

UPDATE-GENERAL-LEDGER-FILE.
  MOVE 0 TO CR-TOTAL.
  MOVE 0 TO DR-TOTAL.
  OPEN INPUT ENTERED-ACCOUNT-FILE
    GENERAL-LEDGER-FILE.
  MOVE 'N' TO NOT-EOF-ENTERED-ACCOUNT-FILE.
  READ ENTERED-ACCOUNT-FILE NEXT AT END MOVE 'Y'
    TO NOT-EOF-ENTERED-ACCOUNT-FILE.
  PERFORM TOTAL-ENTERED-ACCOUNT-FILE UNTIL
    EOF-ENTERED-ACCOUNT-FILE.
  MOVE 'N' TO NOT-EOF-GENERAL-LEDGER-FILE.
  READ GENERAL-LEDGER-FILE NEXT AT END MOVE 'Y'
PERFORM TOTAL-GENERAL-LEDGER-FILE UNTIL EOF-GENERAL-LEDGER-FILE.
CLOSE GENERAL-LEDGER-FILE
ENTERED-ACCOUNT-FILE.
IF CR-TOTAL EQUAL TO DR-TOTAL
PERFORM UPDATE-GENERAL-LEDGER-PROCESS
ELSE DISPLAY SCREEN(22,17) WITH NO ADVANCING
DISPLAY 'ACCOUNT NOT BALANCE. NO UPDATE ' WITH NO ADVANCING.
MOVE 0 TO TIMER.
PERFORM WAIT-FOR-A-WHILE UNTIL TIMER EQUAL 40000.
DISPLAY SCREEN(22,17) WITH NO ADVANCING.
DISPLAY ' WITH NO ADVANCING.

TOTAL-ENTERED-ACCOUNT-FILE.
IF UPDATE-CODE EQUAL 'N'
IF ACC-NUMBER LESS THAN 200000
AND ACC-NUMBER GREATER THAN 400000
ADD ACCOUNT-VALUE TO CR-TOTAL
ELSE ADD ACCOUNT-VALUE TO DR-TOTAL.
READ ENTERED-ACCOUNT-FILE NEXT AT END MOVE 'Y'
TO NOT-EOF-ENTERED-ACCOUNT-FILE.

TOTAL-GENERAL-LEDGER-FILE.
IF G-ACCOUNT-NUMBER LESS THAN 200000
AND G-ACCOUNT-NUMBER GREATER THAN 400000
ADD G-CURRENT-TOTAL TO CR-TOTAL
ELSE ADD G-CURRENT-TOTAL TO DR-TOTAL.
READ GENERAL-LEDGER-FILE NEXT AT END MOVE 'Y'
TO NOT-EOF-GENERAL-LEDGER-FILE.

UPDATE-GENERAL-LEDGER-PROCESS.
MOVE 'N' TO NOT-EOF-ENTERED-ACCOUNT-FILE.
OPEN INPUT-OUTPUT ENTERED-ACCOUNT-FILE
INPUT-OUTPUT GENERAL-LEDGER-FILE.
READ ENTERED-ACCOUNT-FILE AT END MOVE 'Y' TO
NOT-EOF-ENTERED-ACCOUNT-FILE.
PERFORM UPDATE-G-L-PROCESS UNTIL EOF-ENTERED-ACCOUNT-FILE.
CLOSE ENTERED-ACCOUNT-FILE
GENERAL-LEDGER-FILE.
DISPLAY SCREEN(22,17) WITH NO ADVANCING.
DISPLAY ' WITH NO ADVANCING.

UPDATE-G-L-PROCESS.
IF UPDATE-CODE EQUAL 'N'
PERFORM HANDLE-UPDATE.
READ ENTERED-ACCOUNT-FILE AT END MOVE 'Y' TO
NOT-EOF-ENTERED-ACCOUNT-FILE.

HANDLE-UPDATE.
MOVE 'N' TO NOT-ACCOUNT-EXIST.
MOVE ACC-NUMBER TO G-ACCOUNT-NUMBER.
READ GENERAL-LEDGER-FILE INVALID KEY
  MOVE SPACES TO G-ACCOUNT-NUMBER.
  ADD ACCOUNT-VALUE TO G-CURRENT-TOTAL.
REWRITE GENERAL-LEDGER-RECORD INVALID KEY
  MOVE 'Y' TO NOT-ACCOUNT-EXIST.
  MOVE 'Y' TO UPDATE-CODE.
  IF NOT ACCOUNT-NOT-EXIST
  REWRITE ENTERED-ACCOUNT-RECORD.

CLEAN-REVENUE-AND-EXP-SCREEN.
  DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
  DISPLAY SCREEN(22,17) WITH NO ADVANCING.
  DISPLAY 'CLEAN REVENUES AND EXPENSE PROCESSING'
    WITH NO ADVANCING.
  DISPLAY EXIT-REVERSE WITH NO ADVANCING.

CLEAN-REVENUES-AND-EXPENSE.
  OPEN INPUT-OUTPUT GENERAL-LEDGER-FILE.
  MOVE 0 TO TOTAL-INCOME.
  MOVE 'N' TO NOT-EOF-GENERAL-LEDGER-FILE.
  PERFORM REVENUES-EXPENSE-HANDLE UNTIL
    EOF-GENERAL-LEDGER-FILE.
  PERFORM INCOME-HANDLE.
  CLOSE GENERAL-LEDGER-FILE.

REVENUES-EXPENSE-HANDLE.
  READ GENERAL-LEDGER-FILE NEXT AT END
    MOVE 'Y' TO NOT-EOF-GENERAL-LEDGER-FILE.
    IF NOT EOF-GENERAL-LEDGER-FILE
      IF G-ACCOUNT-NUMBER GREATER THAN 299999
        ADD G-CURRENT-TOTAL TO G-YEAR-TO-NOW-TOTAL
        ADD G-CURRENT-TOTAL TO TOTAL-INCOME
        MOVE 0 TO G-CURRENT-TOTAL.
      REWRITE GENERAL-LEDGER-RECORD INVALID KEY
        MOVE 'Y' TO NOT-EOF-GENERAL-LEDGER-FILE.
  INCOME-HANDLE.
    MOVE 243200 TO G-ACCOUNT-NUMBER.
    READ GENERAL-LEDGER-FILE INVALID PERFORM
      ACCOUNT-NOT-EXIST-MESSAGE.
      ADD TOTAL-INCOME TO G-CURRENT-TOTAL.
    REWRITE GENERAL-LEDGER-RECORD INVALID PERFORM
      ACCOUNT-NOT-EXIST-MESSAGE.

ACCOUNT-NOT-EXIST-MESSAGE.
  DISPLAY SCREEN(22,17) WITH NO ADVANCING.
  DISPLAY ' ACCOUNT NOT EXIST ' WITH NO ADVANCING.

CLEAN-OLD-ENTRIES-SCREEN.
  DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
  DISPLAY SCREEN(22,23) WITH NO ADVANCING.
  DISPLAY 'CLEAN OLD ENTRIES PROCESSING'
    WITH NO ADVANCING.
  DISPLAY EXIT-REVERSE WITH NO ADVANCING.

CLEAN-OLD-ENTRIES.
PERFORM CHECK-UPDATE-INDICATOR.
IF NOT NOT-UPDATE
  PERFORM PUT-TO-TEMP-FILE
  PERFORM APPEND-TO-PRE-ENTERED-ACC-FILE
ELSE
  PERFORM CLEAN-OLD-ENTRY-ERROR-MESSAGE.
  DISPLAY SCREEN(22,23) WITH NO ADVANCING.
  DISPLAY ' ' WITH NO ADVANCING.
CHECK-UPDATE-INDICATOR.
  OPEN INPUT ENTERED-ACCOUNT-FILE.
  MOVE 'N' TO NOT-UPDATE-YET.
  MOVE 'N' TO NOT-EOF-ENTERED-ACCOUNT-FILE.
  READ ENTERED-ACCOUNT-FILE AT END MOVE 'Y' TO NOT-EOF-ENTERED-ACCOUNT-FILE.
  PERFORM CHECK-UPDATE UNTIL EOF-ENTERED-ACCOUNT-FILE.
  CLOSE ENTERED-ACCOUNT-FILE.
CHECK-UPDATE.
  IF UPDATE-CODE EQUAL 'N' MOVE 'Y' TO NOT-UPDATE-YET.
  READ ENTERED-ACCOUNT-FILE AT END MOVE 'Y' TO NOT-EOF-ENTERED-ACCOUNT-FILE.
PUT-TO-TEMP-FILE.
  OPEN INPUT PRE-ENTERED-ACCOUNT-FILE
  INPUT ENTERED-ACCOUNT-FILE
  OUTPUT TEMP-ENTERED-ACCOUNT-FILE.
  MOVE 'N' TO NOT-EOF-PRE-ENTERED-ACC-FILE.
  READ PRE-ENTERED-ACCOUNT-FILE AT END MOVE 'Y' TO NOT-EOF-PRE-ENTERED-ACC-FILE.
  PERFORM PUT-TO-TEMP-FILE-PROCESS UNTIL EOF-PRE-ENTERED-ACC-FILE.
  MOVE 'N' TO NOT-EOF-ENTERED-ACCOUNT-FILE.
  READ ENTERED-ACCOUNT-FILE AT END MOVE 'Y' TO NOT-EOF-ENTERED-ACCOUNT-FILE.
  PERFORM ADD-OLD-ENTRIES UNTIL EOF-ENTERED-ACCOUNT-FILE.
  CLOSE PRE-ENTERED-ACCOUNT-FILE ENTERED-ACCOUNT-FILE TEMP-ENTERED-ACCOUNT-FILE.
PUT-TO-TEMP-FILE-PROCESS.
  WRITE TEMP-ENTERED-ACCOUNT-RECORD FROM PRE-ENTERED-ACCOUNT-RECORD.
  READ PRE-ENTERED-ACCOUNT-FILE AT END MOVE 'Y' TO NOT-EOF-PRE-ENTERED-ACC-FILE.
ADD-OLD-ENTRIES.
  WRITE TEMP-ENTERED-ACCOUNT-RECORD FROM ENTERED-ACCOUNT-RECORD.
  READ ENTERED-ACCOUNT-FILE AT END MOVE 'Y' TO NOT-EOF-ENTERED-ACCOUNT-FILE.
APPEND-TO-PRE-ENTERED-ACC-FILE.
APPENDIX C

OPEN INPUT TEMP-ENTERED-ACCOUNT-FILE
OUTPUT PRE-ENTERED-ACCOUNT-FILE.
MOVE 'N' TO NOT-EOF-TEMP-FILE.
READ TEMP-ENTERED-ACCOUNT-FILE AT END MOVE 'Y' TO
NOT-EOF-TEMP-FILE.
PERFORM PUT-BACK-TO-PRE-FILE UNTIL EOF-TEMP-FILE.
CLOSE TEMP-ENTERED-ACCOUNT-FILE
PRE-ENTERED-ACCOUNT-FILE.
OPEN OUTPUT ENTERED-ACCOUNT-FILE
TEMP-ENTERED-ACCOUNT-FILE.
CLOSE TEMP-ENTERED-ACCOUNT-FILE
ENTERED-ACCOUNT-FILE.

PUT-BACK-TO-PRE-FILE.
WRITE PRE-ENTERED-ACCOUNT-RECORD FROM
TEMP-ENTERED-ACCOUNT-RECORD.
READ TEMP-ENTERED-ACCOUNT-FILE AT END MOVE 'Y' TO
NOT-EOF-TEMP-FILE.

CLEAN-OLD-ENTRY-ERROR-MESSAGE.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY SCREEN(22,15) WITH NO ADVANCING.
DISPLAY 'ACCOUNT NOT UPDATE YET. NO CLEAN MADE'
WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
MOVE 0 TO TIMER.
PERFORM WAIT-FOR-A-WHILE UNTIL TIMER EQUAL 40000.
DISPLAY SCREEN(22,15) WITH NO ADVANCING.
DISPLAY ' ' WITH NO ADVANCING.

WAIT-FOR-A-WHILE.
ADD 1 TO TIMER.

END-OF-YEAR-PROCESS-SCREEN.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY SCREEN(22,17) WITH NO ADVANCING.
DISPLAY 'END OF YEAR PROCESSING'
WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.

END-OF-YEAR-PROCESS.
PERFORM CHECK-UPDATE-INDICATOR.
IF NOT NOT-UPDATE PERFORM
END-OF-YEAR-ACCOUNT-HANDLE
ELSE
PERFORM END-OF-YEAR-ERROR-MESSAGE.

END-OF-YEAR-ERROR-MESSAGE.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY SCREEN(22,17) WITH NO ADVANCING.
DISPLAY 'ENTERED ACCOUNT NOT UPDATE YET '
WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
END-OF-YEAR-ACCOUNT-HANDLE.
   MOVE 'N' TO NOT-EOF-GENERAL-LEDGER-FILE.
   PERFORM CLEAN-REVENUES-AND-EXPENSE.
   OPEN INPUT-OUTPUT GENERAL-LEDGER-FILE.
   MOVE 'N' TO NOT-EOF-GENERAL-LEDGER-FILE.
   READ GENERAL-LEDGER-FILE NEXT AT END MOVE 'Y' TO NOT-EOF-GENERAL-LEDGER-FILE.
   PERFORM COPY-AMOUNT-TO-PRE-AMOUNT UNTIL EOF-GENERAL-LEDGER-FILE.
   CLOSE GENERAL-LEDGER-FILE.
   OPEN OUTPUT ENTERED-ACCOUNT-FILE
      PRE-ENTERED-ACCOUNT-FILE.
   CLOSE ENTERED-ACCOUNT-FILE
      PRE-ENTERED-ACCOUNT-FILE.
   DISPLAY SCREEN(22,17) WITH NO ADVANCING.
   DISPLAY ' ' WITH NO ADVANCING.

COPY-AMOUNT-TO-PRE-AMOUNT.
   ADD G-CURRENT-TOTAL TO G-YEAR-TO-NOW-TOTAL.
   MOVE G-YEAR-TO-NOW-TOTAL TO G-PREVIOUS-YEAR-TOTAL.
   MOVE 0 TO G-YEAR-TO-NOW-TOTAL.
   REWRITE GENERAL-LEDGER-RECORD INVALID KEY
      MOVE 'Y' TO NOT-EOF-GENERAL-LEDGER-FILE.
   READ GENERAL-LEDGER-FILE NEXT AT END MOVE 'Y' TO NOT-EOF-GENERAL-LEDGER-FILE.
IDENTIFICATION DIVISION.

PROGRAM-ID.
MAINTN

AUTHOR.
CCW

DATE-WRITTEN.
8-MAR-1984

ENVIRONMENT DIVISION.

INPUT-OUTPUT SECTION.

FILE-CONTROL.

SELECT GENERAL-LEDGER-FILE
  ASSIGN TO DSK
  ORGANIZATION IS INDEXED
  ACCESS MODE IS DYNAMIC
  RECORD KEY IS G-ACCOUNT-NUMBER
  RECORDING MODE IS ASCII.

DATA DIVISION.

FILE SECTION.

FD GENERAL-LEDGER-FILE,
  BLOCK CONTAINS 128 RECORDS,
  VALUE OF ID IS "GENFILINX".

01 GENERAL-LEDGER-RECORD.
  05 G-ACCOUNT-NUMBER PIC 9(6).
  05 G-ACCOUNT-NAME PIC X(30).
  05 G-ACCOUNT-TYPE PIC X.
  05 G-CURRENT-TOTAL PIC S99999999V99.
  05 G-YEAR-TO-NOW-TOTAL PIC S99999999V99.
  05 G-PREVIOUS-YEAR-TOTAL PIC S99999999V99.

WORKING-STORAGE SECTION.

01 TEST-VALUE-A.
  05 FRONT-A PIC X.
  05 BEHIND-A PIC X(9).

01 TEST-LENGTH-C.
  05 FRONT-C PIC X(6).
  05 BEHIND-C PIC X(7).

01 TEST-LENGTH-E.
  05 FRONT-E PIC X(30).
  05 BEHIND-E PIC X(5).
APPENDIX C

01 ADD-INFORMATION.
   05 ADD-ACC-NUMBER   PIC X(6).
   05 ADD-ACC-NAME    PIC X(30).
   05 ADD-ACC-TYPE    PIC X.

01 LEDGER-MAINTAIN-TYPE  PIC X.

01 CONFIRM-VALUE   PIC X VALUE 'N'.

01 RECORD-NOT-EXIST  PIC X.

01 DELETE-ACC-NUMBER PIC X(6).

01 MODIFY-ACC-NUMBER PIC X(6).

01 MODIFY-ACCOUNT-NAME PIC X(30).

01 TIMER        PIC 9999.

77 TRY-AGAIN-VALUE PIC X VALUE 'Y'.
   88 NOT-TRY-AGAIN  VALUE 'N'.

77 EXIT-MAINTAIN-ACC-OR-NOT  PIC X VALUE 'N'.
   88 EXIT-MAINTAIN-ACCOUNT VALUE 'Y'.

77 TAKE-TYPE       PIC X VALUE 'N'.
   88 TAKE-TYPE-OK   VALUE 'Y'.

77 END-GET-MODIFY-INFO-OR-NOT  PIC X VALUE 'N'.
   88 END-GET-MODIFY-INFO VALUE 'Y'.

77 END-GET-ADD-INFO-OR-NOT   PIC X VALUE 'N'.
   88 END-GET-ADD-INFO   VALUE 'Y'.

77 MORE-DELETE-VALUE PIC X VALUE 'Y'.
   88 NO-MORE-DELETE   VALUE 'N'.

77 DELETE-NOT-CONFIRM PIC X VALUE 'N'.
   88 DELETE-CONFIRM   VALUE 'Y'.

77 MORE-ADD-VALUE   PIC X VALUE 'Y'.
   88 NO-MORE-ADD     VALUE 'N'.

77 MORE-MODIFY-VALUE PIC X VALUE 'Y'.
   88 NO-MORE-MODIFY   VALUE 'N'.

77 ACC-NUMBER-ERROR-OR-NOT PIC X VALUE 'N'.
   88 ACC-NUMBER-ERROR VALUE 'Y'.

77 END-GET-DELETE-INFO-OR-NOT PIC X VALUE 'N'.
   88 END-GET-DELETE-INFO VALUE 'Y'.

77 ACC-NAME-ERROR-OR-NOT PIC X VALUE 'N'.
   88 ACC-NAME-ERROR   VALUE 'Y'.

APPENDIX C

*-------------------*
LINKAGE SECTION.
*-------------------*

01 SCREEN-TABLE.
   05 SCREEN-ROWS OCCURS 24 TIMES.
   10 SCREEN PIC X(4)
      USAGE IS DISPLAY-7
      OCCURS 80 TIMES.

01 REVERSE-VIDEO PIC XX DISPLAY-7.
01 EXIT-REVERSE PIC XX DISPLAY-7.
01 CLEAR-SCREEN PIC XX DISPLAY-7.
01 RING-BELL PIC X DISPLAY-7.

*-----------------------------*
PROCEDURE DIVISION USING SCREEN-TABLE
   CLEAR-SCREEN RING-BELL
   REVERSE-VIDEO EXIT-REVERSE.
*-----------------------------*

PROCEDURE DIVISION USING SCREEN-TABLE
   CLEAR-SCREEN RING-BELL
   REVERSE-VIDEO EXIT-REVERSE.

MAIN.
   MOVE 'N' TO EXIT-MAINTAIN-ACC-OR-NOT.
   OPEN INPUT-OUTPUT GENERAL-LEDGER-FILE.
   PERFORM MAINTAIN-INFO-PROCESS UNTIL
      EXIT-MAINTAIN-ACCOUNT.
   CLOSE GENERAL-LEDGER-FILE.
   EXIT PROGRAM.

MAINTAIN-INFO-PROCESS.
   MOVE ' ' TO LEDGER-MAINTAIN-TYPE.
   PERFORM MAINTAIN-INFO-PROCESS-SCREEN.
   MOVE 'N' TO TAKE-TYPE.
   PERFORM TAKE-TYPE-OF-INFO-MAINTAIN
      UNTIL TAKE-TYPE-OK.
   IF LEDGER-MAINTAIN-TYPE = '1'
      PERFORM ADD-G-L-RECORD-SCREEN
      MOVE 'Y' TO MORE-ADD-VALUE
      PERFORM ADD-GENERAL-LEDGER-RECORD
      UNTIL NO-MORE-ADD
   ELSE IF LEDGER-MAINTAIN-TYPE = '2'
      PERFORM DEL-GENERAL-LEDGER-SCREEN
      MOVE 'Y' TO MORE-DELETE-VALUE
      PERFORM DEL-GENERAL-LEDGER-RECORD
      UNTIL NO-MORE-DELETE
   ELSE IF LEDGER-MAINTAIN-TYPE = '3'
      PERFORM MODIFY-GENERAL-LEDGER-SCREEN
      MOVE 'Y' TO MORE-MODIFY-VALUE
      PERFORM MODIFY-GENERAL-LEDGER-RECORD
      UNTIL NO-MORE-MODIFY
   ELSE IF LEDGER-MAINTAIN-TYPE= '*'
MAINTAIN-INFO-PROCESS-SCREEN.
DISPLAY CLEAR-SCREEN WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY SCREEN(3,35) WITH NO ADVANCING.
DISPLAY 'SCREEN - 1.1.3' WITH NO ADVANCING.
DISPLAY SCREEN(4,23) WITH NO ADVANCING.
DISPLAY 'SELECT GENERAL-LEDGER MAINTAIN TYPE' WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY 'ENTER SELECTION: ' WITH NO ADVANCING.
DISPLAY SCREEN(10,20) WITH NO ADVANCING.
DISPLAY '=================================================================' WITH NO ADVANCING.
DISPLAY SCREEN(12,20) WITH NO ADVANCING.
DISPLAY ' * : FOR EXIT-GENERAL-LEDGER-MAINTAIN ' WITH NO ADVANCING.
DISPLAY SCREEN(14,20) WITH NO ADVANCING.
DISPLAY ' 1 : FOR ADD-GENERAL-LEDGER-RECORD ' WITH NO ADVANCING.
DISPLAY SCREEN(16,20) WITH NO ADVANCING.
DISPLAY ' 2 : FOR DELETE-GENERAL-LEDGER-RECORD ' WITH NO ADVANCING.
DISPLAY SCREEN(18,20) WITH NO ADVANCING.
DISPLAY ' 3 : FOR MODIFY-GENERAL-LEDGER-RECORD ' WITH NO ADVANCING.
DISPLAY SCREEN(20,20) WITH NO ADVANCING.
DISPLAY '=================================================================' WITH NO ADVANCING.

TAKE-TYPE-OF-INFO-MAINTAIN.
MOVE 'N' TO TAKE-TYPE.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY SCREEN(6,47) WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A NOT EQUAL SPACES PERFORM ERROR-HANDLE-A.
ELSE MOVE FRONT-A TO LEDGER-MAINTAIN-TYPE.
IF LEDGER-MAINTAIN-TYPE EQUAL '1' OR EQUAL '2' OR EQUAL '3' OR EQUAL '*' MOVE 'Y' TO TAKE-TYPE.
ELSE PERFORM ERROR-HANDLE-A.

ERROR-HANDLE-A.
APPENDIX C

DISPLAY SCREEN(22,7) WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY 'ERROR MESSAGE::' WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY 'INVALID RESPONSE . PLEASE REENTER'
WITH NO ADVANCING.
DISPLAY SCREEN(6,47) WITH NO ADVANCING.
DISPLAY ' • ' WITH NO ADVANCING.
DISPLAY SCREEN(6,47) WITH NO ADVANCING.
DISPLAY RING-BELL WITH NO ADVANCING.
PERFORM TAKE-TYPE-OF-INFO-MAINTAIN.

ADD-GENERAL-LEDGER-RECORD.
MOVE 'N' TO END-GET-ADD-INFO-OR-NOT.
MOVE 'N' TO ACC-NUMBER-ERROR-OR-NOT.
MOVE 'Y' TO TRY-AGAIN-VALUE.
PERFORM GET-VALID-ADD-ACC-RECORD UNTIL
END-GET-ADD-INFO.

IF NOT ACC-NUMBER-ERROR
PERFORM GET-ADD-CONFIRM-PROCESS.
IF NOT ACC-NUMBER-ERROR
IF CONFIRM-VALUE = 'Y'
PERFORM ADD-LEDGER-ACCOUNT-PROCESS
ELSE
PERFORM DISPLAY-NO-ADD-MESSAGE.
PERFORM ASK-ADD-MORE.
IF MORE-ADD-VALUE = 'Y' PERFORM
ADD-GENERAL-LEDGER-RECORD
ELSE
MOVE 'N' TO MORE-ADD-VALUE.

GET-VALID-ADD-ACC-RECORD.
MOVE 'N' TO ACC-NUMBER-ERROR-OR-NOT.
MOVE 'N' TO TRY-AGAIN-VALUE.
PERFORM GET-ADD-ACC-NUMBER.
IF ACC-NUMBER-ERROR-OR-NOT = 'N' PERFORM
CHECK-RECORD-NOT-EXIST.
IF ACC-NUMBER-ERROR PERFORM ASK-ADD-TRY-AGAIN
ELSE PERFORM GET-VALID-ADD-ACC-NAME
PERFORM GET-VALID-ADD-ACC-TYPE.
IF TRY-AGAIN-VALUE = 'Y' PERFORM
GET-VALID-ADD-ACC-RECORD.
IF ACC-NUMBER-ERROR-OR-NOT = 'Y' OR
NOT-TRY-AGAIN
MOVE 'Y' TO END-GET-ADD-INFO-OR-NOT.

ADD-G-L-RECORD-SCREEN.
DISPLAY CLEAR-SCREEN WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY SCREEN(3,31) WITH NO ADVANCING.
DISPLAY ' SCREEN - 1.1.3.1'
WITH NO ADVANCING.
DISPLAY SCREEN(4,26) WITH NO ADVANCING.
DISPLAY ' ADD GENERAL LEDGER RECORD '
APPENDIX C

WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY SCREEN (8,20) WITH NO ADVANCING.
DISPLAY 'ENTER ACCOUNT-NUMBER : ' WITH NO ADVANCING.
DISPLAY SCREEN(11,20) WITH NO ADVANCING.
DISPLAY 'ENTER ACCOUNT-NAME ; ' WITH NO ADVANCING.
DISPLAY SCREEN(14,20) WITH NO ADVANCING.
DISPLAY 'ENTER ACCOUNT-TYPE : ' WITH NO ADVANCING.
DISPLAY SCREEN(17,20) WITH NO ADVANCING.
DISPLAY 'ENTER ACCOUNT-VALUE : ' WITH NO ADVANCING.
GET-ADD-ACC-NUMBER.
DISPLAY SCREEN(11,43) WITH NO ADVANCING.
DISPLAY '__________________________' WITH NO ADVANCING.
PERFORM ERASE-ERROR-MESSAGE.
MOVE 'N' TO TRY-AGAIN-VALUE.
IF ACC-NUMBER-ERROR PERFORM
  DISPLAY-ERROR-MESSAGE.
  MOVE 'N' TO ACC-NUMBER-ERROR-OR-NOT.
  MOVE SPACES TO TEST-LENGTH-C.
  DISPLAY RING-BELL WITH NO ADVANCING.
  DISPLAY SCREEN(8,43) WITH NO ADVANCING.
  DISPLAY ' ' WITH NO ADVANCING.
  DISPLAY SCREEN(8,43) WITH NO ADVANCING.
ACCEPT TEST-LENGTH-C.
  IF BEHIND-C NOT EQUAL SPACES
    MOVE 'Y' TO ACC-NUMBER-ERROR-OR-NOT.
    MOVE FRONT-C TO ADD-ACC-NUMBER.
    IF ADD-ACC-NUMBER IS NOT NUMERIC
      MOVE 'Y' TO ACC-NUMBER-ERROR-OR-NOT.
    IF ADD-ACC-NUMBER LESS THAN 100000 OR GREATER THAN 500000 MOVE 'Y' TO
      ACC-NUMBER-ERROR-OR-NOT.
CHECK-RECORD-NOT-EXIST.
  MOVE 'N' TO RECORD-NOT-EXIST.
  MOVE ADD-ACC-NUMBER TO G-ACCOUNT-NUMBER.
  READ GENERAL-LEDGER-FILE INVALID KEY
  MOVE 'Y' TO RECORD-NOT-EXIST.
  IF RECORD-NOT-EXIST = 'N' MOVE 'Y' TO
    ACC-NUMBER-ERROR-OR-NOT.
GET-VALID-ADD-ACC-NAME.
  IF ACC-NUMBER-ERROR PERFORM
    DISPLAY-ERROR-MESSAGE.
    MOVE 'N' TO ACC-NUMBER-ERROR-OR-NOT.
    MOVE SPACES TO TEST-LENGTH-E.
    DISPLAY RING-BELL WITH NO ADVANCING.
    DISPLAY SCREEN(11,43) WITH NO ADVANCING.
    DISPLAY '__________________________'
ACCEPT TEST-VALUE-A.
IF BEHIND-A NOT EQUAL SPACES
MOVE 'Y' TO ACC-NUMBER-ERROR-OR-NOT
PERFORM GET-VALID-ADD-ACC-TYPE.
ELSE MOVE FRONT-A TO ADD-ACC-TYPE.
IF ADD-ACC-TYPE NOT EQUAL '0' AND NOT
EQUAL '1' AND NOT EQUAL '2'
AND NOT EQUAL '3'
MOVE 'Y' TO ACC-NUMBER-ERROR-OR-NOT
PERFORM GET-VALID-ADD-ACC-TYPE.
IF NOT ACC-NUMBER-ERROR MOVE FRONT-A TO
G-ACCOUNT-TYPE.

ADD-LEDGER-ACCOUNT-PROCESS.
MOVE SPACES TO GENERAL-LEDGER-RECORD.
MOVE ADD-ACC-NAME TO G-ACCOUNT-NAME.
MOVE ADD-ACC-TYPE TO G-ACCOUNT-TYPE.
MOVE ADD-ACC-NUMBER TO G-ACCOUNT-NUMBER.
MOVE ZEROS TO G-YEAR-TO-NOW-TOTAL.
MOVE ZEROS TO G-PREVIOUS-YEAR-TOTAL.
MOVE ZEROS TO G-CURRENT-TOTAL.
WRITE GENERAL-LEDGER-RECORD INVALID KEY
DISPLAY SCREEN(21,1) WITH NO ADVANCING
DISPLAY
'ADD A NEW GENERAL LEDGER RECORD
WITH NO ADVANCING.'

ASK-ADD-MORE.
MOVE SPACES TO TEST-VALUE-A.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY SCREEN (20,19) WITH NO ADVANCING.
DISPLAY ' ADD MORE ? (Y OR N) ' WITH NO ADVANCING.
DISPLAY SCREEN(20,41) WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A IS NOT EQUAL SPACES
PERFORM ADD-MORE-ERROR-HANDLE.
MOVE FRONT-A TO MORE-ADD-VALUE.
IF MORE-ADD-VALUE NOT EQUAL 'N' AND
MORE-ADD-VALUE NOT EQUAL 'Y'
PERFORM ADD-MORE-ERROR-HANDLE.

ADD-MORE-ERROR-HANDLE.
MOVE SPACES TO TEST-VALUE-A.
DISPLAY SCREEN(22,5) WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY 'ERROR MESSAGE::' WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY SCREEN(23,5) WITH NO ADVANCING.
DISPLAY 'INVALID RESPONSE . PLEASE REENTER'
WITH NO ADVANCING.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY SCREEN(20,31) WITH NO ADVANCING.
DISPLAY _ 'WITH NO ADVANCING.
DISPLAY SCREEN(20,32) WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A IS NOT EQUAL SPACES
PERFORM ADD-MORE-ERROR-HANDLE.
MOVE FRONT-A TO MORE-ADD-VALUE.
IF MORE-ADD-VALUE NOT EQUAL 'N' AND
MORE-ADD-VALUE NOT EQUAL 'Y'
PERFORM ADD-MORE-ERROR-HANDLE.

ASK-ADD-TRY-AGAIN.
MOVE 'Y' TO TRY-AGAIN-VALUE.
MOVE SPACES TO TEST-VALUE-A.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY SCREEN(23,1) WITH NO ADVANCING.
DISPLAY ' ACCOUNT ERROR :: '
WITH NO ADVANCING.
DISPLAY SCREEN(23,21) WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY ' WANT TRY AGAIN? ENTER Y OR N :: _
WITH NO ADVANCING.
DISPLAY SCREEN(23,55) WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A NOT EQUAL SPACES PERFORM
ASK-ADD-TRY-AGAIN
ELSE MOVE FRONT-A TO TRY-AGAIN-VALUE.
IF TRY-AGAIN-VALUE NOT EQUAL 'Y' AND NOT
EQUAL 'N' PERFORM ASK-ADD-TRY-AGAIN.

DISPLAY-NO-ADD-MESSAGE.
MOVE 0 TO TIMER.
DISPLAY SCREEN(20,19) WITH NO ADVANCING.
DISPLAY 'NO ADDITION MADE'
WITH NO ADVANCING.
PERFORM WAIT-FOR-A-WHILE UNTIL
TIMER EQUAL 9999.

WAIT-FOR-A-WHILE.
ADD 1 TO TIMER.

DISPLAY-ERROR-MESSAGE.
DISPLAY SCREEN(22,1) WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY 'ERROR MESSAGE:;'>' WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY SCREEN(23,1) WITH NO ADVANCING.
DISPLAY ' INVALID ENTERED DATA. PLEASE REENTER ' WITH NO ADVANCING.

ERASE-ERROR-MESSAGE.
DISPLAY SCREEN(14,43) WITH NO ADVANCING.
DISPLAY ' ' WITH NO ADVANCING.
DISPLAY SCREEN(17,43) WITH NO ADVANCING.
DISPLAY ' ' WITH NO ADVANCING.
DISPLAY SCREEN(20,1) WITH NO ADVANCING.
DISPLAY ' ' WITH NO ADVANCING.
DISPLAY SCREEN(22,1) WITH NO ADVANCING.
DISPLAY ' ' WITH NO ADVANCING.
DISPLAY SCREEN(23,1) WITH NO ADVANCING.
DISPLAY ' ' WITH NO ADVANCING.

GET-ADD-CONFIRM-PROCESS.
DISPLAY SCREEN(11,43) WITH NO ADVANCING.
DISPLAY ADD-ACC-NAME WITH NO ADVANCING.
DISPLAY SCREEN(14,43) WITH NO ADVANCING.
DISPLAY G-ACCOUNT-TYPE WITH NO ADVANCING.
DISPLAY SCREEN(17,43) WITH NO ADVANCING.
DISPLAY G-CURRENT-TOTAL WITH NO ADVANCING.
DISPLAY SCREEN(22,1) WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY 'ACTION CONFIRM ::' WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
PERFORM GET-ADD-CONFIRM-VALUE.

GET-ADD-CONFIRM-VALUE.
DISPLAY SCREEN(23,1) WITH NO ADVANCING.
DISPLAY 'IS THIS RECORD TO BE ADDED? ENTER Y OR N :: _ WITH NO ADVANCING.
DISPLAY SCREEN(23,45) WITH NO ADVANCING.
DISPLAY RING-BELL WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A NOT EQUAL SPACES PERFORM
GET-ADD-CONFIRM-VALUE
ELSE MOVE FRONT-A TO CONFIRM-VALUE.
IF CONFIRM-VALUE NOT EQUAL 'Y' AND NOT
EQUAL 'N'
APPENDIX C

PERFORM GET-ADD-CONFIRM-VALUE.

DEL-GENERAL-LEDGER-RECORD.
  MOVE 'N' TO END-GET-DELETE-INFO-OR-NOT.
  MOVE 'Y' TO TRY-AGAIN-VALUE.
  PERFORM GET-VALID-ACC-DELETE-RECORD UNTIL
      END-GET-DELETE-INFO.
  IF NOT ACC-NUMBER-ERROR
      PERFORM GET-CONFIRM-PROCESS.
  IF NOT ACC-NUMBER-ERROR
      IF CONFIRM-VALUE = 'Y'
          PERFORM DELETE-LEDGER-ACCOUNT-PROCESS
      ELSE
          PERFORM DISPLAY-NO-DELETE-MESSAGE.
          PERFORM ASK-DELETE-MORE.
      IF MORE-DELETE-VALUE = 'Y'
          PERFORM DEL-GENERAL-LEDGER-RECORD
      ELSE
          MOVE 'N' TO MORE-DELETE-VALUE.

DEL-GENERAL-LEDGER-SCREEN.
  DISPLAY CLEAR-SCREEN WITH NO ADVANCING.
  DISPLAY SCREEN(3,32) WITH NO ADVANCING.
  DISPLAY 'SCREEN - 1.1.3.2'
      WITH NO ADVANCING.
  DISPLAY SCREEN(4,26) WITH NO ADVANCING.
  DISPLAY 'DELETE GENERAL LEDGER RECORD'
      WITH NO ADVANCING.
  DISPLAY SCREEN (7,20) WITH NO ADVANCING.
  DISPLAY 'ENTER ACCOUNT-NUMBER'
      WITH NO ADVANCING.
  DISPLAY SCREEN (8,20) WITH NO ADVANCING.
  DISPLAY 'WHICH TO BE DELETED : ________'
      WITH NO ADVANCING.
  DISPLAY SCREEN(11,20) WITH NO ADVANCING.
  DISPLAY
      ' ACCOUNT-NAME : ' WITH NO ADVANCING.
  DISPLAY SCREEN(14,20) WITH NO ADVANCING.
  DISPLAY
      ' ACCOUNT-TYPE : ' WITH NO ADVANCING.
  DISPLAY SCREEN(17,20) WITH NO ADVANCING.
  DISPLAY
      ' ACCOUNT-VALUE : ' WITH NO ADVANCING.

GET-VALID-ACC-DELETE-RECORD.
  MOVE 'N' TO ACC-NUMBER-ERROR-OR-NOT.
  MOVE 'N' TO TRY-AGAIN-VALUE.
  PERFORM GET-VALID-ACC-NUMBER.
  IF NOT ACC-NUMBER-ERROR
      PERFORM CHECK-RECORD-EXIST.
  IF NOT ACC-NUMBER-ERROR
      PERFORM CHECK-CURRENT-VALUE-ZERO.
  IF ACC-NUMBER-ERROR PERFORM ASK-TRY-AGAIN.
  IF TRY-AGAIN-VALUE = 'Y'
      PERFORM GET-VALID-ACC-DELETE-RECORD.
IF NOT ACC-NUMBER-ERROR OR NOT-TRY-AGAIN
MOVE 'Y' TO END-GET-DELETE-INFO-OR-NOT.

GET-VALID-ACC-NUMBER.
PERFORM ERASE-DELETE-ERROR-MESSAGE.
MOVE 'N' TO TRY-AGAIN-VALUE.
IF ACC-NUMBER-ERROR PERFORM
  DISPLAY-ERROR-MESSAGE.
  MOVE 'N' TO ACC-NUMBER-ERROR-OR-NOT.
  MOVE SPACES TO TEST-LENGTH-C.
  DISPLAY RING-BELL WITH NO ADVANCING.
  DISPLAY SCREEN(8,43) WITH NO ADVANCING.
  DISPLAY ' ' WITH NO ADVANCING.
  DISPLAY SCREEN(8,43) WITH NO ADVANCING.
  ACCEPT TEST-LENGTH-C.
  IF BEHIND-C NOT EQUAL SPACES
    MOVE 'Y' TO ACC-NUMBER-ERROR-OR-NOT.
MOVE FRONT-C TO DELETE-ACC-NUMBER.
IF DELETE-ACC-NUMBER IS NOT NUMERIC
  MOVE 'Y' TO ACC-NUMBER-ERROR-OR-NOT.
IF DELETE-ACC-NUMBER LESS THAN 100000 OR
  GREATER THAN 500000 MOVE 'Y' TO
  ACC-NUMBER-ERROR-OR-NOT.

CHECK-RECORD-EXIST.
  MOVE DELETE-ACC-NUMBER TO G-ACCOUNT-NUMBER.
  READ GENERAL-LEDGER-FILE INVALID KEY
  MOVE 'Y' TO ACC-NUMBER-ERROR-OR-NOT.

CHECK-CURRENT-VALUE-ZERO.
  IF G-CURRENT-TOTAL NOT EQUAL ZERO
    MOVE 'Y' TO ACC-NUMBER-ERROR-OR-NOT.

GET-CONFIRM-PROCESS.
  DISPLAY SCREEN(11,43) WITH NO ADVANCING.
  DISPLAY G-ACCOUNT-NAME WITH NO ADVANCING.
  DISPLAY SCREEN(14,43) WITH NO ADVANCING.
  DISPLAY G-ACCOUNT-TYPE WITH NO ADVANCING.
  DISPLAY SCREEN(17,43) WITH NO ADVANCING.
  DISPLAY G-CURRENT-TOTAL WITH NO ADVANCING.
  DISPLAY SCREEN(22,1) WITH NO ADVANCING.
  DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
  DISPLAY 'ACTION CONFIRM ::' WITH NO ADVANCING.
  DISPLAY EXIT-REVERSE WITH NO ADVANCING.
  PERFORM GET-CONFIRM-VALUE.

GET-CONFIRM-VALUE.
  DISPLAY SCREEN(23,1) WITH NO ADVANCING.
  DISPLAY
    'IS THIS RECORD TO BE DELETED? ENTER Y OR N :: _
  WITH NO ADVANCING.
  DISPLAY SCREEN(23,47) WITH NO ADVANCING.
  ACCEPT TEST-VALUE-A.
  IF BEHIND-A NOT EQUAL SPACES PERFORM
    GET-CONFIRM-VALUE
APPENDIX C

ELSE MOVE FRONT-A TO CONFIRM-VALUE.
IF CONFIRM-VALUE NOT EQUAL 'Y' AND
   NOT EQUAL 'N'
PERFORM GET-CONFIRM-VALUE.

DELETE-LEDGER-ACCOUNT-PROCESS.
DELETE GENERAL-LEDGER-FILE INVALID KEY
DISPLAY 'RECORD NOT FOUND CAN NOT BE DELETE'
     WITH NO ADVANCING.

DISPLAY-NO-DELETE-MESSAGE.
MOVE 0 TO TIMER.
DISPLAY SCREEN(20,19) WITH NO ADVANCING.
DISPLAY 'NO DELETION MADE' WITH NO ADVANCING.
PERFORM WAIT-FOR-A-WHILE UNTIL
     TIMER EQUAL 9999.

ASK-DELETE-MORE.
MOVE SPACES TO TEST-VALUE-A.
DISPLAY SCREEN (20,19) WITH NO ADVANCING.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY 'DELETE MORE? (Y OR N) '
     WITH NO ADVANCING.
DISPLAY '
     WITH NO ADVANCING.
DISPLAY SCREEN(20,41) WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A IS NOT EQUAL SPACES
   PERFORM DELETE-MORE-ERROR-HANDLE.
MOVE FRONT-A TO MORE-DELETE-VALUE.
IF MORE-DELETE-VALUE NOT EQUAL 'N' AND
   MORE-DELETE-VALUE NOT EQUAL 'Y'
   PERFORM DELETE-MORE-ERROR-HANDLE.

DELETE-MORE-ERROR-HANDLE.
MOVE SPACES TO TEST-VALUE-A.
DISPLAY SCREEN(22,1) WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY ' ERROR MESSAGE:; '
     WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY SCREEN(23,5) WITH NO ADVANCING.
DISPLAY 'INVALID RESPONSE . PLEASE REENTER
     WITH NO ADVANCING.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY SCREEN(20,41) WITH NO ADVANCING.
DISPLAY '
     WITH NO ADVANCING.
DISPLAY SCREEN(20,41) WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A IS NOT EQUAL SPACES
   PERFORM DELETE-MORE-ERROR-HANDLE.
MOVE FRONT-A TO MORE-DELETE-VALUE.
IF MORE-DELETE-VALUE NOT EQUAL 'N' AND
   MORE-DELETE-VALUE NOT EQUAL 'Y'
   PERFORM DELETE-MORE-ERROR-HANDLE.
DISPLAY SCREEN(22,1) WITH NO ADVANCING.
APPENDIX C

DISPLAY

' WITH NO ADVANCING.

DISPLAY SCREEN(23,1) WITH NO ADVANCING.

DISPLAY

' WITH NO ADVANCING.

ASK-TRY-AGAIN.

MOVE 'Y' TO TRY-AGAIN-VALUE.

MOVE SPACES TO TEST-VALUE-A.

DISPLAY SCREEN(23,1) WITH NO ADVANCING.

DISPLAY REVERSE-VIDEO WITH NO ADVANCING.

DISPLAY ' RECORD NOT EXIST ::'

WITH NO ADVANCING.

DISPLAY SCREEN(23,21) WITH NO ADVANCING.

DISPLAY EXIT-REVERSE WITH NO ADVANCING.

DISPLAY ' WANT TRY AGAIN . ENTER Y OR N :: _'

WITH NO ADVANCING.

DISPLAY SCREEN(23,55) WITH NO ADVANCING.

ACCEPT TEST-VALUE-A.

IF BEHIND-A NOT EQUAL SPACES PERFORM

ASK-TRY-AGAIN

ELSE MOVE FRONT-A TO TRY-AGAIN-VALUE.

IF TRY-AGAIN-VALUE NOT EQUAL 'Y' AND

NOT EQUAL 'N' PERFORM ASK-TRY-AGAIN.

ERASE-DELETE-ERROR-MESSAGE.

DISPLAY SCREEN(11,43) WITH NO ADVANCING.

DISPLAY

' WITH NO ADVANCING.

DISPLAY SCREEN(14,43) WITH NO ADVANCING.

DISPLAY ' WITH NO ADVANCING.

DISPLAY SCREEN(17,43) WITH NO ADVANCING.

DISPLAY ' WITH NO ADVANCING.

DISPLAY SCREEN(20,1) WITH NO ADVANCING.

DISPLAY

' WITH NO ADVANCING.

DISPLAY SCREEN(22,1) WITH NO ADVANCING.

DISPLAY

' WITH NO ADVANCING.

DISPLAY SCREEN(23,1) WITH NO ADVANCING.

DISPLAY

' WITH NO ADVANCING.

MODIFY-GENERAL-LEDGER-RECORD.

MOVE 'N' TO END-GET-MODIFY-INFO-OR-NOT.

MOVE 'N' TO ACC-NUMBER-ERROR-OR-NOT.

MOVE 'Y' TO TRY-AGAIN-VALUE.

PERFORM GET-VALID-MODIFY-ACC-RECORD UNTIL

END-GET-MODIFY-INFO.

IF NOT ACC-NUMBER-ERROR

PERFORM GET-MODIFY-CONFIRM-PROCESS.

IF NOT ACC-NUMBER-ERROR AND

CONFIRM-VALUE = 'Y'


APPENDIX C

IF CONFIRM-VALUE = 'Y'
   PERFORM MODIFY-LEDGER-ACCOUNT-PROCESS
ELSE
   PERFORM DISPLAY-NO-MODIFY-MESSAGE 20 TIMES.
   PERFORM ASK-MODIFY-MORE.
   IF MORE-MODIFY-VALUE = 'Y' PERFORM
      MODIFY-GENERAL-LEDGER-RECORD
   ELSE
      MOVE 'N' TO MORE-MODIFY-VALUE.

MODIFY-GENERAL-LEDGER-SCREEN.
   DISPLAY CLEAR-SCREEN WITH NO ADVANCING.
   DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
   DISPLAY SCREEN(3,32) WITH NO ADVANCING.
   DISPLAY 'SCREEN - 1.1.3.3'
      WITH NO ADVANCING.
   DISPLAY SCREEN(4,26) WITH NO ADVANCING.
   DISPLAY 'MODIFY GENERAL LEDGER RECORD'
      WITH NO ADVANCING.
   DISPLAY EXIT-REVERSE WITH NO ADVANCING.
   DISPLAY SCREEN(8,20) WITH NO ADVANCING.
   DISPLAY 'ENTER ACCOUNT-NUMBER : ______'
      WITH NO ADVANCING.
   DISPLAY SCREEN(11,20) WITH NO ADVANCING.
   DISPLAY 'ENTER ACCOUNT-NAME : '
      WITH NO ADVANCING.
   DISPLAY SCREEN(14,20) WITH NO ADVANCING.
   DISPLAY 'ENTER ACCOUNT-TYPE :
      ' WITH NO ADVANCING.
   DISPLAY SCREEN(17,20) WITH NO ADVANCING.
   DISPLAY 'ENTER ACCOUNT-VALUE : '
      WITH NO ADVANCING.

GET-VALID-MODIFY-ACC-RECORD.
   MOVE 'N' TO ACC-NUMBER-ERROR-OR-NOT.
   MOVE 'N' TO TRY-AGAIN-VALUE.
   PERFORM GET-VALID-MODIFY-ACC-NUMBER.
   IF NOT ACC-NUMBER-ERROR
      PERFORM CHECK-MODIFY-RECORD-EXIST.
      IF ACC-NUMBER-ERROR
         PERFORM ASK-MODIFY-TRY-AGAIN.
      IF TRY-AGAIN-VALUE = 'Y' PERFORM
         GET-VALID-MODIFY-ACC-RECORD
      ELSE
         IF NOT ACC-NUMBER-ERROR
            PERFORM MODIFY-ACTION.
         IF NOT ACC-NUMBER-ERROR OR NOT-TRY-AGAIN
            MOVE 'Y' TO END-GET-MODIFY-INFO-OR-NOT.

GET-VALID-MODIFY-ACC-NUMBER.
   PERFORM ERASE-DELETE-ERROR-MESSAGE.
   MOVE 'N' TO TRY-AGAIN-VALUE.
   IF ACC-NUMBER-ERROR
      PERFORM DISPLAY-ERROR-MESSAGE.
   MOVE 'N' TO ACC-NUMBER-ERROR-OR-NOT.
   MOVE SPACES TO TEST-LENGTH-C.
APPENDIX C

DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY SCREEN(8,43) WITH NO ADVANCING.
DISPLAY '_____ ' WITH NO ADVANCING.
DISPLAY SCREEN(8,43) WITH NO ADVANCING.
ACCEPT TEST-LENGTH-C.
IF BEHIND-C NOT EQUAL SPACES
MOVE 'Y' TO ACC-NUMBER-ERROR-OR-NOT.
MOVE FRONT-C TO MODIFY-ACC-NUMBER.
IF MODIFY-ACC-NUMBER IS NOT NUMERIC
MOVE 'Y' TO ACC-NUMBER-ERROR-OR-NOT.
IF MODIFY-ACC-NUMBER LESS THAN 100000 OR
GREATER THAN 500000 MOVE 'Y' TO
ACC-NUMBER-ERROR-OR-NOT.

CHECK-MODIFY-RECORD-EXIST.
MOVE MODIFY-ACC-NUMBER TO G-ACCOUNT-NUMBER.
READ GENERAL-LEDGER-FILE INVALID KEY
MOVE 'Y' TO ACC-NUMBER-ERROR-OR-NOT.

MODIFY-ACTION.
PERFORM SHOW-RECORD-CONTENTS.
DISPLAY SCREEN(20,1) WITH NO ADVANCING.
DISPLAY ' NEW ACCOUNT NAME? '
WITH NO ADVANCING.
DISPLAY '______________________________'
WITH NO ADVANCING.
DISPLAY ' ' WITH NO ADVANCING.
DISPLAY SCREEN(20,22) WITH NO ADVANCING.
ACCEPT TEST-LENGTH-E.
IF BEHIND-E NOT EQUAL SPACES
DISPLAY SCREEN(23,1) WITH NO ADVANCING
DISPLAY REVERSE-VIDEO WITH NO ADVANCING
DISPLAY ' ACCOUNT NAME ERROR :::
WITH NO ADVANCING
DISPLAY EXIT-REVERSE WITH NO ADVANCING
PERFORM MODIFY-ACTION.
IF FRONT-E IS NOT ALPHABETIC
DISPLAY SCREEN(23,1) WITH NO ADVANCING
DISPLAY REVERSE-VIDEO WITH NO ADVANCING
DISPLAY ' ACCOUNT NAME ERROR :::
WITH NO ADVANCING
DISPLAY EXIT-REVERSE WITH NO ADVANCING
PERFORM MODIFY-ACTION.
MOVE FRONT-E TO MODIFY-ACCOUNT-NAME.
PERFORM SHOW-RECORD-CONTENTS.
DISPLAY SCREEN(20,1) WITH NO ADVANCING.
DISPLAY ' ' WITH NO ADVANCING.
DISPLAY ' RECORD CONTENT AFTER MODIFICATION'
WITH NO ADVANCING.
DISPLAY ' ' WITH NO ADVANCING.

SHOW-RECORD-CONTENTS.
DISPLAY SCREEN (8,43) WITH NO ADVANCING.
APPENDIX C

DISPLAY G-ACCOUNT-NUMBER WITH NO ADVANCING.
DISPLAY SCREEN(11,44) WITH NO ADVANCING.
IF MODIFY-ACCOUNT-NAME EQUAL SPACES
DISPLAY G-ACCOUNT-NAME
ELSE
DISPLAY MODIFY-ACCOUNT-NAME WITH NO ADVANCING.
DISPLAY SCREEN(14,44) WITH NO ADVANCING.
DISPLAY G-ACCOUNT-TYPE WITH NO ADVANCING.
DISPLAY SCREEN(17,44) WITH NO ADVANCING.
DISPLAY G-CURRENT-TOTAL WITH NO ADVANCING.

MODIFY-LEDGER-ACCOUNT-PROCESS.
MOVE MODIFY-ACCOUNT-NAME TO G-ACCOUNT-NAME.
REWRITE GENERAL-LEDGER-RECORD INVALID KEY
MOVE 'Y' TO ACC-NUMBER-ERROR-OR-NOT.

DISPLAY-NO-MODIFY-MESSAGE.
DISPLAY SCREEN(20,10) WITH NO ADVANCING.
DISPLAY 'NO MODIFY MADE'
WITH NO ADVANCING.

ASK-MODIFY-MORE.
MOVE SPACES TO TEST-VALUE-A.
DISPLAY SCREEN (20,10) WITH NO ADVANCING.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY 'MODIFY MORE? (Y OR N) '
WITH NO ADVANCING.
DISPLAY ' 
WITH NO ADVANCING.
DISPLAY SCREEN(20,32) WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A IS NOT EQUAL SPACES
PERFORM MODIFY-MORE-ERROR-HANDLE.
MOVE FRONT-A TO MORE-MODIFY-VALUE.
IF MORE-MODIFY-VALUE NOT EQUAL 'N' AND
MORE-MODIFY-VALUE NOT EQUAL 'Y'
PERFORM MODIFY-MORE-ERROR-HANDLE.

MODIFY-MORE-ERROR-HANDLE.
MOVE SPACES TO TEST-VALUE-A.
DISPLAY SCREEN(22,5) WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY 'ERROR MESSAGE::'WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY SCREEN(23,1) WITH NO ADVANCING.
DISPLAY 'INVALID RESPONSE . PLEASE REENTER'
WITH NO ADVANCING.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY SCREEN(20,31) WITH NO ADVANCING.
DISPLAY ' 
WITH NO ADVANCING.
DISPLAY SCREEN(20,32) WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A IS NOT EQUAL SPACES
PERFORM MODIFY-MORE-ERROR-HANDLE.
MOVE FRONT-A TO MORE-MODIFY-VALUE.
IF MORE-MODIFY-VALUE NOT EQUAL 'N' AND
MORE-MODIFY-VALUE NOT EQUAL 'Y'
PERFORM MODIFY-MORE-ERROR-HANDLE.
DISPLAY SCREEN(22,1) WITH NO ADVANCING.
DISPLAY ' WITH NO ADVANCING.
DISPLAY SCREEN(23,1) WITH NO ADVANCING.
DISPLAY ' WITH NO ADVANCING.

ASK-MODIFY-TRY-AGAIN.
MOVE 'Y' TO TRY-AGAIN-VALUE.
MOVE SPACES TO TEST-VALUE-A.
DISPLAY SCREEN(23,1) WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY ' ACCOUNT ERROR ::' WITH NO ADVANCING.
DISPLAY SCREEN(23,21) WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY ' WANT TRY AGAIN . ENTER Y OR N :: _ WITH NO ADVANCING.
DISPLAY SCREEN(23,55) WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A NOT EQUAL SPACES PERFORM
ASK-MODIFY-TRY-AGAIN
ELSE MOVE FRONT-A TO TRY-AGAIN-VALUE.
IF TRY-AGAIN-VALUE NOT EQUAL 'Y' AND NOT EQUAL 'N' PERFORM ASK-MODIFY-TRY-AGAIN.

GET-MODIFY-CONFIRM-PROCESS.
DISPLAY SCREEN(22,1) WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY ' MODIFY CONFIRM ::' WITH NO ADVANCING.
DISPLAY SCREEN(23,21) WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY SCREEN(23,1) WITH NO ADVANCING.
DISPLAY 'IS THIS RECORD TO BE MODIFY? ENTER Y OR N :: _ WITH NO ADVANCING.
DISPLAY SCREEN(23,46) WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A NOT EQUAL SPACES PERFORM
GET-MODIFY-CONFIRM-PROCESS
ELSE MOVE FRONT-A TO CONFIRM-VALUE.
IF CONFIRM-VALUE NOT EQUAL 'Y' AND NOT EQUAL 'N' PERFORM GET-MODIFY-CONFIRM-PROCESS.
IDENTIFICATION DIVISION.

PROGRAM-ID.  
MTAN

AUTHOR.  
CCW

DATE-WRITTEN.  
8-MAR-1984

ENVIRONMENT DIVISION.

INPUT-OUTPUT SECTION.

FILE-CONTROL.

SELECT ENTERED-ACCOUNT-FILE,
  ASSIGN TO DSK,
  RECORDING MODE IS ASCII.

SELECT TEMP-ENTERED-ACCOUNT-FILE,
  ASSIGN TO DSK,
  RECORDING MODE IS ASCII.

SELECT PRE-ENTERED-ACCOUNT-FILE,
  ASSIGN TO DSK,
  RECORDING MODE IS ASCII.

SELECT GENERAL-LEDGER-FILE 
  ASSIGN TO DSK 
  ORGANIZATION IS INDEXED 
  ACCESS MODE IS DYNAMIC 
  RECORD KEY IS G-ACCOUNT-NUMBER 
  RECORDING MODE IS ASCII.

DATA DIVISION.

FILE SECTION.

FD GENERAL-LEDGER-FILE,  
  BLOCK CONTAINS 128 RECORDS,  
  VALUE OF ID IS "GENFILINX".

01 GENERAL-LEDGER-RECORD.  
  05 G-ACCOUNT-NUMBER      PIC 9(6).  
  05 G-ACCOUNT-NAME        PIC X(30).  
  05 G-ACCOUNT-TYPE        PIC X.  
  05 G-CURRENT-TOTAL       PIC S99999999V99.  
  05 G-YEAR-TO-NOW-TOTAL   PIC S99999999V99.  
  05 G-PREVIOUS-YEAR-TOTAL PIC S99999999V99.

FD ENTERED-ACCOUNT-FILE,  
  BLOCK CONTAINS 128 RECORDS,
APPENDIX C

LABEL RECORDS ARE STANDARD,
VALUE OF ID IS 'ACCOUNT'.

01 ENTERED-ACCOUNT-RECORD.
  05 ENTRY-NO PIC 999999.
  05 ACC-NO.
    08 CATAGORY-NO PIC 9.
    08 SUBCATAGORY-NO PIC 99999.
  05 ACC-NAME PIC X(30).
  05 SOURCE-CODE PIC 9.
  05 ACCOUNT-VALUE PIC S99999999V99.
  05 DATE-OF-ENTRY PIC XXXXXX.
  05 UPDATE-CODE PIC X.

FD TEMP-ENTERED-ACCOUNT-FILE
BLOCK CONTAINS 128 RECORDS,
LABEL RECORDS ARE STANDARD,
VALUE OF ID IS 'TEMPACTMP'.

01 TEMP-ENTERED-ACCOUNT-RECORD.
  05 T-ENTRY-NO PIC 999999.
  05 T-ACC-NO.
    08 T-CATAGORY-NO PIC 9.
    08 T-SUBCATAGORY-NO PIC 99999.
  05 T-ACC-NAME PIC X(30).
  05 T-SOURCE-CODE PIC 9.
  05 T-ACCOUNT-VALUE PIC S99999999V99.
  05 T-DATE-OF-ENTRY PIC XXXXXX.
  05 T-UPDATE-CODE PIC X.

FD PRE-ENTERED-ACCOUNT-FILE,
BLOCK CONTAINS 128 RECORDS,
LABEL RECORDS ARE STANDARD,
VALUE OF ID IS 'PREVI FIL'.

01 PRE-ENTERED-ACCOUNT-RECORD.
  05 PRE-ENTRY-NO PIC 999999.
  05 PRE-ACC-NO.
    08 PRE-CATAGORY-NO PIC 9.
    08 PRE-SUBCATAGORY-NO PIC 99999.
  05 PRE-ACC-NAME PIC X(30).
  05 PRE-SOURCE-CODE PIC 9.
  05 PRE-ACCOUNT-VALUE PIC S99999999V99.
  05 PRE-DATE-OF-ENTRY PIC XXXXXX.
  05 PRE-UPDATE-CODE PIC X.

WORKING-STORAGE SECTION.

01 SCREEN-SYS-DATE.
  05 SCREEN-SYS-YEAR PIC X(2).
  05 FILLER PIC X(3) VALUE ' / '.
  05 SCREEN-SYS-MONTH PIC X(2).
  05 FILLER PIC X(3) VALUE ' / '.
  05 SCREEN-SYS-DAY PIC X(2).

01 SYS-DATE.
01 REVERSE-VIDEO PIC XX DISPLAY-7.
01 EXIT-REVERSE PIC XX DISPLAY-7.
01 CLEAR-SCREEN PIC XX DISPLAY-7.
01 RING-BELL PIC X DISPLAY-7.

*---------------------------------------------------------------
PROCEDURE DIVISION USING SCREEN-TABLE CLEAR-SCREEN
RING-BELL REVERSE-VIDEO EXIT-REVERSE.

*---------------------------------------------------------------
MAIN.

MOVE 'N' TO EXIT-MAINTAIN-ACC-OR-NOT.
PERFORM ACCOUNT-MAINTAIN-PROCESS UNTIL
EXIT-MAINTAIN-ACCOUNT.

EXIT PROGRAM.

ACCOUNT-MAINTAIN-PROCESS.

MOVE 'N' TO TAKE-TYPE.
PERFORM SELECT-MAINTAIN-TYPE-SCREEN.
PERFORM TAKE-TYPE-OF-MAINTAIN UNTIL
TAKE-TYPE-OK.

IF MAINTAIN-TYPE = '1'
    PERFORM ENTER-ACCOUNT-ENTRIES
ELSE IF MAINTAIN-TYPE = '2'
    PERFORM ADJUSTMENT-ACCOUNT-ENTRIES
ELSE IF MAINTAIN-TYPE = '3'
    PERFORM MAINTAIN-ACCOUNT-INFORMATION
ELSE IF MAINTAIN-TYPE = '*'
    MOVE 'Y' TO EXIT-MAINTAIN-ACC-OR-NOT
    DISPLAY CLEAR-SCREEN WITH NO ADVANCING.

SELECT-MAINTAIN-TYPE-SCREEN.

DISPLAY CLEAR-SCREEN WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY SCREEN(1,29) WITH NO ADVANCING.
DISPLAY 'SCREEN - 1.1'
    WITH NO ADVANCING.
DISPLAY SCREEN(2,22) WITH NO ADVANCING.
DISPLAY 'ACCOUNT MAINTAIN SELECTION'
    WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY SCREEN (4,17) WITH NO ADVANCING.
DISPLAY
'SELECT ONE OF THE FOLLOWING TYPE TO PROCESS'
    WITH NO ADVANCING.
DISPLAY SCREEN (5,31) WITH NO ADVANCING.
DISPLAY 'ENTER SELECTION: _'
    WITH NO ADVANCING.
DISPLAY SCREEN(7,16) WITH NO ADVANCING.
DISPLAY
'=================================================================='
APPENDIX C

WITH NO ADVANCING.
DISPLAY SCREEN(9,16) WITH NO ADVANCING.
DISPLAY ' * : FOR EXIT ACCOUNT MAINTAIN PROCESS ' WITH NO ADVANCING.
DISPLAY SCREEN(11,16) WITH NO ADVANCING.
DISPLAY ' 1 : FOR ENTER ENTRY ' WITH NO ADVANCING.
DISPLAY SCREEN(13,16) WITH NO ADVANCING.
DISPLAY ' 2 : FOR ENTER ADJUSTING ENTRY ' WITH NO ADVANCING.
DISPLAY SCREEN(15,16) WITH NO ADVANCING.
DISPLAY ' 3 ; FOR MAINTAIN ACCOUNT INFORMATION ' WITH NO ADVANCING.
DISPLAY '=================================================================================' WITH NO ADVANCING.

TAKE-TYPE-OF-MAINTAIN.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY SCREEN(5,49) WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A NOT EQUAL SPACES PERFORM ERROR-HANDLE-A
ELSE MOVE FRONT-A TO MAINTAIN-TYPE.
IF MAINTAIN-TYPE EQUAL '1' OR EQUAL '2' OR EQUAL '3' OR EQUAL '*' MOV 'Y' TO TAKE-TYPE
ELSE PERFORM ERROR-HANDLE-A.

ERROR-HANDLE-A.
DISPLAY SCREEN(22,7) WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY 'ERROR MESSAGE: ' WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY 'INVALID RESPONSE . PLEASE REENTER' WITH NO ADVANCING.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY SCREEN(5,49) WITH NO ADVANCING.
DISPLAY '_ ' WITH NO ADVANCING.
DISPLAY SCREEN(5,49) WITH NO ADVANCING.
PERFORM TAKE-TYPE-OF-MAINTAIN.

ENTER-ACCOUNT-ENTRIES.
MOVE 'N' TO UPDATE-CODE-INDICATOR.
MOVE 'N' TO END-OF-ENTRY.
PERFORM OPEN-FILES-1.
PERFORM APPEND-TO-ACCOUNT-FILE UNTIL END-OF-INPUT.
PERFORM CLOSE-FILES-1.
APPENDIX C

OPEN INPUT TEMP-ENTERED-ACCOUNT-FILE
OUTPUT ENTERED-ACCOUNT-FILE.
MOVE 'N' TO CLEAR-FILE-OK.
READ TEMP-ENTERED-ACCOUNT-FILE AT END
MOVE 'Y' TO CLEAR-FILE-OK.
PERFORM CLEAR-TEMP-FILE UNTIL CLEAR-OK.
PERFORM DELETE-TEMP-FILE.
CLOSE ENTERED-ACCOUNT-FILE.

ADJUSTMENT-ACCOUNT-ENTRIES.
MOVE 'N' TO END-OF-ENTRY.
MOVE 'Y' TO ADJUSTMENT-MODE-OR-NOT.
PERFORM ENTER-ACCOUNT-ENTRIES UNTIL
END-OF-INPUT.
MOVE 'N' TO ADJUSTMENT-MODE-OR-NOT.

MAINTAIN-ACCOUNT-INFORMATION.
CALL 'MAINTN' USING SCREEN-TABLE
CLEAR-SCREEN RING-BELL
REVERSE-VIDEO EXIT-REVERSE.

OPEN-FILES-1.
OPEN INPUT ENTERED-ACCOUNT-FILE.
OPEN OUTPUT TEMP-ENTERED-ACCOUNT-FILE.
OPEN INPUT GENERAL-LEDGER-FILE.

CLOSE-FILES-1.
CLOSE ENTERED-ACCOUNT-FILE.
CLOSE TEMP-ENTERED-ACCOUNT-FILE.
CLOSE GENERAL-LEDGER-FILE.

APPEND-TO-ACCOUNT-FILE.
ACCEPT SYS-DATE FROM DATE.
MOVE SYS-YEAR TO SCREEN-SYS-YEAR.
MOVE SYS-MONTH TO SCREEN-SYS-MONTH.
MOVE SYS-DAY TO SCREEN-SYS-DAY.
MOVE SYS-DATE TO DATE-OF-ENTRY.
MOVE 'N' TO END-ENTERED-ACC-FILE-OR-NOT.
MOVE 0 TO ENTRY-NUMBER.
READ ENTERED-ACCOUNT-FILE AT END MOVE
'Y' TO END-ENTERED-ACC-FILE-OR-NOT.
PERFORM OBTAIN-LAST-ENTRY-NUMBER
UNTIL END-ENTERED-ACC-FILE.
MOVE ENTRY-NO TO ENTRY-NUMBER.
IF ENTRY-NUMBER EQUAL 0
MOVE 'N' TO END-PRE-ENTRY-FILE-OR-NOT
ELSE MOVE 'Y' TO END-PRE-ENTRY-FILE-OR-NOT.
PERFORM OBTAIN-PRE-ENTRY-NO UNTIL
END-PRE-ENTRY-FILE.
ADD 1 TO ENTRY-NUMBER.
MOVE ENTRY-NUMBER TO ENTRY-NO.
IF ADJUSTMENT-MODE MOVE '4' TO SOURCE-CODE.
PERFORM SCREEN-DISPLAY.
PERFORM INPUT-DATA UNTIL END-OF-INPUT.

OBTAIN-PRE-ENTRY-NO.
OPEN INPUT PRE-ENTERED-ACCOUNT-FILE.
PERFORM OBTAIN-PRE-LAST-ENTRY-NO UNTIL END-PRE-ENTRY-FILE.
MOVE PRE-ENTRY-NO TO ENTRY-NUMBER.
CLOSE PRE-ENTERED-ACCOUNT-FILE.

OBTAIN-PRE-LAST-ENTRY-NO.
READ PRE-ENTERED-ACCOUNT-FILE AT END
MOVE 'Y' TO END-PRE-ENTRY-FILE-OR-NOT.

OBTAIN-LAST-ENTRY-NUMBER.
WRITE TEMP-ENTERED-ACCOUNT-RECORD FROM
ENTERED-ACCOUNT-RECORD.
READ ENTERED-ACCOUNT-FILE AT END MOVE
'Y' TO END-ENTERED-ACC-FILE-OR-NOT.

SCREEN-DISPLAY.
DISPLAY CLEAR-SCREEN WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
IF ADJUSTMENT-MODE
  DISPLAY SCREEN(1,30) WITH NO ADVANCING
  DISPLAY 'SCREEN - 1.1.2'
  WITH NO ADVANCING
  DISPLAY SCREEN(2,25) WITH NO ADVANCING
  DISPLAY 'ADJUSTMENT ENTRY PROCESS'
  WITH NO ADVANCING
ELSE
  DISPLAY SCREEN(1,30) WITH NO ADVANCING
  DISPLAY 'SCREEN - 1.1.1'
  WITH NO ADVANCING
  DISPLAY SCREEN(2,25) WITH NO ADVANCING
  DISPLAY 'ENTER ENTRY PROCESS'
  WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY SCREEN (6,10) WITH NO ADVANCING.
DISPLAY
'=========================================================================
WITH NO ADVANCING.
DISPLAY SCREEN (9,10) WITH NO ADVANCING.
DISPLAY ' DATE ' WITH NO ADVANCING.
DISPLAY SCREEN-SYS-DATE WITH NO ADVANCING.
  DISPLAY SCREEN(9,42) WITH NO ADVANCING.
DISPLAY ' SOURCE CODE _WITH NO ADVANCING.
IF ADJUSTMENT-MODE
  DISPLAY SCREEN(9,58) WITH NO ADVANCING
  DISPLAY SOURCE-CODE WITH NO ADVANCING.
  DISPLAY SCREEN(11,10) WITH NO ADVANCING.
  DISPLAY ' ENTRY NUMBER ++++++
  WITH NO ADVANCING.
DISPLAY SCREEN(13,10) WITH NO ADVANCING.
DISPLAY ' ACCOUNT NUMBER ______'
WITH NO ADVANCING.
DISPLAY SCREEN(15,10) WITH NO ADVANCING.
DISPLAY
' ACCOUNT NAME ____________________'
WITH NO ADVANCING.
INPUT-DATA.

MOVE 'Y' TO TRY-AGAIN-OR-NOT.
IF NOT ADJUSTMENT-MODE
PERFORM OBTAIN-SOURCE-CODE.
PERFORM OBTAIN-ENTRY-NUMBER.
PERFORM OBTAIN-ACCOUNT-NUMBER.
IF TRY-AGAIN
PERFORM OBTAIN-ACCOUNT-NAME
PERFORM OBTAIN-ACCOUNT-VALUE
MOVE UPDATE-CODE-INDICATOR TO UPDATE-CODE
MOVE SYS-DATE TO DATE-OF-ENTRY
WRITE TEMP-ENTERED-ACCOUNT-RECORD FROM
ENTERED-ACCOUNT-RECORD
ADD 1 TO ENTRY-NUMBER.
PERFORM MORE-DATA.
IF MORE-TO-ENTER
PERFORM NEXT-ENTRY-SCREEN
ELSE MOVE 'Y' TO END-OF-ENTRY.

OBTAIN-SOURCE-CODE.
MOVE SPACES TO TEST-LENGTH-B.
DISPLAY SCREEN(9,58) WITH NO ADVANCING.
ACCEPT TEST-LENGTH-B.
IF BEHIND-B NOT EQUAL SPACES PERFORM
SOURCE-CODE-ERROR-HANDLE.
IF FRONT-B IS NOT NUMERIC PERFORM
SOURCE-CODE-ERROR-HANDLE
ELSE MOVE FRONT-B TO SOURCE-CODE.
IF SOURCE-CODE GREATER THAN 3
OR SOURCE-CODE LESS THAN 1
PERFORM SOURCE-CODE-ERROR-HANDLE.

OBTAIN-ENTRY-NUMBER.
DISPLAY SCREEN(11,26) WITH NO ADVANCING.
DISPLAY ENTRY-NUMBER WITH NO ADVANCING.

OBTAIN-ACCOUNT-NUMBER.
MOVE SPACES TO G-ACCOUNT-TYPE.
MOVE SPACES TO TEST-LENGTH-C.
DISPLAY SCREEN(13,26) WITH NO ADVANCING.
ACCEPT TEST-LENGTH-C.
IF BEHIND-C NOT EQUAL SPACES PERFORM
ACCOUNT-NUMBER-ERROR-HANDLE.
IF TRY-AGAIN
IF FRONT-C IS NOT NUMERIC PERFORM
ACCOUNT-NUMBER-ERROR-HANDLE
ELSE MOVE FRONT-C TO ACC-NO.
IF TRY-AGAIN
IF ACC-NO LESS THAN 100000 OR GREATER THAN
500000 PERFORM ACCOUNT-NUMBER-ERROR-HANDLE.
MOVE SPACE TO GENERAL-LEDGER-RECORD.
MOVE ACC-NO TO G-ACCOUNT-NUMBER.
APPENDIX C

IF TRY-AGAIN
READ GENERAL-LEDGER-FILE INVALID KEY PERFORM
ACCOUNT-NUMBER-ERROR-HANDLE UNTIL TRY-AGAIN.
IF TRY-AGAIN
IF G-ACCOUNT-TYPE NOT EQUAL '0'
PERFORM ACCOUNT-NUMBER-ERROR-HANDLE.

OBTAIN-ACCOUNT-NAME.
MOVE G-ACCOUNT-NAME TO ACC-NAME.
DISPLAY SCREEN (15,26) WITH NO ADVANCING.
DISPLAY ACC-NAME WITH NO ADVANCING.

OBTAIN-ACCOUNT-VALUE.
MOVE SPACES TO TEST-LENGTH-D.
DISPLAY SCREEN(17,26) WITH NO ADVANCING.
ACCEPT TEST-LENGTH-D.
IF TEST-LENGTH-D NOT NUMERIC PERFORM
ACCOUNT-VALUE-ERROR-HANDLE.
IF TEST-LENGTH-D LESS THAN -99999999.99 OR
GREATER THAN 99999999.99
PERFORM ACCOUNT-VALUE-ERROR-HANDLE
ELSE MOVE TEST-LENGTH-D TO ACCOUNT-VALUE.

MORE-DATA.
MOVE SPACES TO TEST-VALUE-A.
DISPLAY SCREEN (20,10) WITH NO ADVANCING.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY 'ENTER MORE? (Y OR N) _'
WITH NO ADVANCING.
DISPLAY SCREEN(20,31) WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A IS NOT EQUAL SPACES
PERFORM MORE-ENTRY-ERROR-HANDLE.
MOVE FRONT-A TO MORE-ENTRY .
IF MORE-ENTRY NOT EQUAL 'N' AND MORE-ENTRY NOT
EQUAL 'Y' PERFORM MORE-ENTRY-ERROR-HANDLE.
MOVE ENTRY-NUMBER TO ENTRY-NO.
DISPLAY SCREEN(22,5) WITH NO ADVANCING.
DISPLAY
WITH NO ADVANCING.
DISPLAY SCREEN(23,5) WITH NO ADVANCING.
DISPLAY
WITH NO ADVANCING.

ACCOUNT-NUMBER-ERROR-HANDLE.
MOVE 'Y' TO TRY-AGAIN-OR-NOT.
MOVE SPACES TO TEST-LENGTH-C.
DISPLAY SCREEN(22,5) WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY 'ERROR MESSAGE::'WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY SCREEN(23,5) WITH NO ADVANCING.
IF G-ACCOUNT-TYPE EQUAL '0'
DISPLAY
APPENDIX C

'INVALID ACCOUNT NUMBER. PLEASE REENTER'
WITH NO ADVANCING

ELSE
DISPLAY
'IT IS NOT CORRECT ACCOUNT PLEASE CHECK AGAIN'
WITH NO ADVANCING.
PERFORM ENTER-TRY-AGAIN.
IF TRY-AGAIN PERFORM GET-NEW-ACCOUNT-NUMBER.

GET-NEW-ACCOUNT-NUMBER.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY SCREEN(13,26) WITH NO ADVANCING.
DISPLAY ' ' WITH NO ADVANCING.
DISPLAY SCREEN(13,26) WITH NO ADVANCING.
MOVE SPACES TO TEST-LENGTH-C.
ACCEPT TEST-LENGTH-C.
IF BEHIND-C NOT EQUAL SPACES PERFORM
ACCOUNT-NUMBER-ERROR-HANDLE.
IF FRONT-C IS NOT NUMERIC OR NOT TRY-AGAIN
PERFORM ACCOUNT-NUMBER-ERROR-HANDLE
ELSE MOVE FRONT-C TO ACC-NO.
IF ACC-NO LESS THAN 100000 OR GREATER THAN
500000 PERFORM ACCOUNT-NUMBER-ERROR-HANDLE.
MOVE SPACES TO G-ACCOUNT-TYPE.
MOVE ACC-NO TO G-ACCOUNT-NUMBER.
READ GENERAL-LEDGER-FILE INVALID KEY PERFORM
ACCOUNT-NUMBER-ERROR-HANDLE.
IF G-ACCOUNT-TYPE NOT EQUAL '0'
PERFORM ACCOUNT-NUMBER-ERROR-HANDLE.
DISPLAY SCREEN(20,1) WITH NO ADVANCING.
DISPLAY
',
DISPLAY SCREEN(22,5) WITH NO ADVANCING.
DISPLAY
',
DISPLAY SCREEN(23,5) WITH NO ADVANCING.
DISPLAY
',

ENTER-TRY-AGAIN.
MOVE SPACES TO TEST-VALUE-A.
DISPLAY SCREEN (20,10) WITH NO ADVANCING.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY 'TRY AGAIN ? (Y OR N) '
WITH NO ADVANCING.
DISPLAY SCREEN(20,31) WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A IS NOT EQUAL SPACES
PERFORM TRY-AGAIN-ERROR-HANDLE
ELSE MOVE FRONT-A TO TRY-AGAIN-OR-NOT.
IF TRY-AGAIN-OR-NOT NOT EQUAL 'N' AND
TRY-AGAIN-OR-NOT NOT EQUAL 'Y' PERFORM
TRY-AGAIN-ERROR-HANDLE.
DISPLAY SCREEN(20,1) WITH NO ADVANCING.
DISPLAY
'          
WITH NO ADVANCING.

TRY-AGAIN-ERROR-HANDLE.
MOVE SPACES TO TEST-VALUE-A.
DISPLAY SCREEN(22,5) WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY 'ERROR MESSAGE::'WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY SCREEN(23,5) WITH NO ADVANCING.
DISPLAY
'INVALID RESPONSE . PLEASE REENTER
WITH NO ADVANCING.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY SCREEN(20,31) WITH NO ADVANCING.
DISPLAY 'WITH NO ADVANCING.
DISPLAY SCREEN(20,31) WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A IS NOT EQUAL SPACES
PERFORM TRY-AGAIN-ERROR-HANDLE.
MOVE FRONT-A TO TRY-AGAIN-OR-NOT.
IF TRY-AGAIN-OR-NOT NOT EQUAL 'N' AND
TRY-AGAIN-OR-NOT NOT EQUAL 'Y'
PERFORM TRY-AGAIN-ERROR-HANDLE.
DISPLAY SCREEN(22,5) WITH NO ADVANCING.
DISPLAY
'          
WITH NO ADVANCING.
DISPLAY SCREEN(23,5) WITH NO ADVANCING.
DISPLAY
'          
WITH NO ADVANCING.

SOURCE-CODE-ERROR-HANDLE.
MOVE SPACES TO TEST-LENGTH-B.
DISPLAY SCREEN(22,5) WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY 'ERROR MESSAGE::'WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY SCREEN(23,5) WITH NO ADVANCING.
DISPLAY
'INVALID SOURCE CODE . PLEASE REENTER'
WITH NO ADVANCING.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY SCREEN(9,58) WITH NO ADVANCING.
DISPLAY 'WITH NO ADVANCING.
DISPLAY SCREEN(9,58) WITH NO ADVANCING.
DISPLAY SCREEN(9,58)WITH NO ADVANCING.
ACCEPT TEST-LENGTH-B.
IF BEHIND-B NOT EQUAL SPACES PERFORM
SOURCE-CODE-ERROR-HANDLE.
IF FRONT-B IS NOT NUMERIC PERFORM
SOURCE-CODE-ERROR-HANDLE
ELSE MOVE FRONT-B TO SOURCE-CODE.
IF SOURCE-CODE GREATER THAN 3
OR SOURCE-CODE LESS THAN 1
PERFORM SOURCE-CODE-ERROR-HANDLE.
DISPLAY SCREEN(22,5) WITH NO ADVANCING.
DISPLAY
' ' WITH NO ADVANCING.
DISPLAY SCREEN(23,5) WITH NO ADVANCING.
DISPLAY
' ' WITH NO ADVANCING.

ACCOUNT-VALUE-ERROR-HANDLE.
DISPLAY SCREEN(22,5) WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY 'ERROR MESSAGE::' WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY SCREEN(23,5) WITH NO ADVANCING.
DISPLAY
'INVALID ACCOUNT VALUE . PLEASE REENTER' WITH NO ADVANCING.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY SCREEN(17,26) WITH NO ADVANCING.
DISPLAY ' ' WITH NO ADVANCING.
DISPLAY SCREEN(17,26) WITH NO ADVANCING.
ACCEPT TEST-LENGTH-D.
IF TEST-LENGTH-D NOT NUMERIC PERFORM
ACCOUNT-VALUE-ERROR-HANDLE.
IF TEST-LENGTH-D LESS THAN -99999999.99 OR GREATER THAN 99999999.99
PERFORM ACCOUNT-VALUE-ERROR-HANDLE
ELSE MOVE TEST-LENGTH-D TO ACCOUNT-VALUE.
DISPLAY SCREEN(22,5) WITH NO ADVANCING.
DISPLAY
' ' WITH NO ADVANCING.
DISPLAY SCREEN(23,5) WITH NO ADVANCING.
DISPLAY
' ' WITH NO ADVANCING.

MORE-ENTRY-ERROR-HANDLE.
MOVE SPACES TO TEST-VALUE-A.
DISPLAY SCREEN(22,5) WITH NO ADVANCING.
DISPLAY REVERSE-VIDEO WITH NO ADVANCING.
DISPLAY 'ERROR MESSAGE::' WITH NO ADVANCING.
DISPLAY EXIT-REVERSE WITH NO ADVANCING.
DISPLAY SCREEN(23,5) WITH NO ADVANCING.
DISPLAY 'INVALID RESPONSE . PLEASE REENTER' WITH NO ADVANCING.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY SCREEN(20,31) WITH NO ADVANCING.
DISPLAY ' ' WITH NO ADVANCING.
DISPLAY SCREEN(20,32) WITH NO ADVANCING.
ACCEPT TEST-VALUE-A.
IF BEHIND-A IS NOT EQUAL SPACES
PERFORM MORE-ENTRY-ERROR-HANDLE.
MOVE FRONT-A TO MORE-ENTRY.
IF MORE-ENTRY NOT EQUAL 'N' AND MORE-ENTRY NOT
EQUAL 'Y' PERFORM MORE-ENTRY-ERROR-HANDLE.
DISPLAY SCREEN(22,5) WITH NO ADVANCING.
DISPLAY
' WITH NO ADVANCING.
DISPLAY SCREEN(23,5) WITH NO ADVANCING.
DISPLAY
' WITH NO ADVANCING.

NEXT-ENTRY-SCREEN.
DISPLAY RING-BELL WITH NO ADVANCING.
DISPLAY SCREEN(9,58) WITH NO ADVANCING.
IF NOT ADJUSTMENT-MODE
DISPLAY ' ' WITH NO ADVANCING.
DISPLAY SCREEN(11,26) WITH NO ADVANCING.
DISPLAY ' ' WITH NO ADVANCING.
DISPLAY SCREEN(13,26) WITH NO ADVANCING.
DISPLAY ' ' WITH NO ADVANCING.
DISPLAY SCREEN(15,26) WITH NO ADVANCING.
DISPLAY '++++++++++++++++++++++++++++++'
WITH NO ADVANCING.
DISPLAY SCREEN(17,26) WITH NO ADVANCING.
DISPLAY ' ' WITH NO ADVANCING.
DISPLAY SCREEN(18,1) WITH NO ADVANCING.
DISPLAY
' WITH NO ADVANCING.
DISPLAY SCREEN(19,1) WITH NO ADVANCING.
DISPLAY
' WITH NO ADVANCING.
DISPLAY SCREEN(20,1) WITH NO ADVANCING.
DISPLAY
' WITH NO ADVANCING.

CLEAR-TEMP-FILE.
WRITE ENTERED-ACCOUNT-RECORD FROM
TEMP-ENTERED-ACCOUNT-RECORD.
READ TEMP-ENTERED-ACCOUNT-FILE AT END
MOVE 'Y' TO CLEAR-FILE-OK.

DELETE-TEMP-FILE.
CLOSE TEMP-ENTERED-ACCOUNT-FILE.
OPEN OUTPUT TEMP-ENTERED-ACCOUNT-FILE.
CLOSE TEMP-ENTERED-ACCOUNT-FILE.
This general ledger system is based on the menu-driven method. Thus, this user's manual contains step-by-step instructions for use of all menus included in this system. With each screen menu in this system, an explanation of the menu's functions is provided for the user. Error recovery caused by the user are also included in this manual.

Before using this system, please reference the appendix d-3 at end of this manual to initialize this system for use. These files need only be created once. After the initialization, user does not need to re-initialize. After the user turns on the terminal and login successful, simply typing in "ledger <CR>" will start the general ledger system. This will reference a PCL (Program Command Language) command which will set the terminal width to be zero and will run the general ledger system. Termination of the general ledger system session will reset the terminal width, i.e. eighty columns. Please reference the appendix d-2 at end of this manual for the using of PCL. The general ledger system will begin by showing the user the main screen, i.e. SCREEN - 1.0. According to the user's selection of menu choice, the proper module is called upon. A flow chart of logical steps for using this system is attached at the end of this manual (appendix d-4). Starting from the main screen i.e. SCREEN - 1.0, the functions of
each screen will be described in hierarchical order. Each screen in this manual has a unique label number to identify it. The label number can be separated into two parts, the last digit and preceding digits. The preceding digits represent the screen number which is calling the current screen. The number chosen of that screen is represented in the last digit. Example: SCREEN - 1.1.2 is evoked by SCREEN - 1.1, which is the '2' function. When describing a screen, the label number will be combined to represent a particular screen for easy reference. The hierarchy of screen is shown in FIGURE D-1.

![Diagram of Screen Hierarchy](image.png)

Figure-D
The following notes must be observed when using the general ledger system. First, it is difficult to represent all the displays of the screen which have the same position as on paper. Thus, in this manual, the screen with the # sign is used to represent those places which are shared positions with other information. The messages shown on the # signs will be explained separately with each screen.

Second, this system is written for a terminal which accepts the VT52 mode. If this system is going to be used on another terminal mode, the screen.cbl (screen driver in COBOL) program needs to be adapted to fit the requirement of that terminal.

Third, it is unusual to modify the general ledger account. However, to add or delete the general ledger file, be sure it is processed properly. Before making any changes to the general ledger account, referencing the chart of account is suggested. This is because each heading account matches only one total account. Thus, to add or delete a heading type account, a total type account must added or deleted to the general ledger file.
SCREEN - 1.0

GENERAL LEDGER SYSTEM

SELECT ONE OF THE FOLLOWING TYPE TO PROCESS
ENTER SELECTION:

* : FOR EXIT GENERAL LEDGER SYSTEM
1 : FOR ACCOUNT MAINTAIN PROCESS
2 : FOR ACCOUNT UPDATE PROCESS
3 : FOR REPORT GENERATOR PROCESS

PURPOSE:

This screen is the first screen in this system. It allows the user four selections: to maintain all accounts, process different account updates, print different reports and exit the system. This screen is the driver of the system. To change from one of the above executed processes to another, one must go back to this screen then call the desired process.

The choice of '1' will call the SCREEN - 1.1 for entry, adjusting entry and maintaining account information. The choice of '2' will call the SCREEN - 1.2 for update general ledger file, clean revenues and expense, clean old entries and end of year process. The choice of '3' will call the screen-1.3 for print the chart of account, print the list of account entered, print the audit trial, print the ledger sheet, print the trial balance, print the income statement and print the balance sheet.

ERRORS RECOVERY:

The user can only enter '*', '1', '2' or '3' values. Otherwise, the error message ## <1> ## will be shown and the cursor will move to the enter position for reentering the selection. The response must be correct, otherwise the system will request again.

DISPLAY MESSAGES AND PROMPTS ::

### <1> ### :: ERROR MESSAGE::INVALID RESPONSE. PLEASE REENTER
-------------------------- SCREEN - 1.1 --------------------------
ACCOUNT MAINTAIN SELECTION

SELECT ONE OF THE FOLLOWING TYPE TO PROCESS
ENTER SELECTION: _

==============================================
* : FOR EXIT ACCOUNT MAINTAIN PROCESS
1 : FOR ENTER ENTRY
2 : FOR ENTER ADJUSTING ENTRY
3 : FOR MAINTAIN ACCOUNT INFORMATION
==============================================

##### <1> #######

PURPOSE:

This screen is shown through the selection '1' of screen-1.0 calling or back from its previous called screen. It has four different functions to choose from: Enter entry, enter adjusting entry, maintain account information and exit this screen back to the original SCREEN - 1.0.

The choice of '1' will call the SCREEN - 1.1.1 for enter entry data. The choice of '2' will call the SCREEN - 1.1.2 for enter adjusting entries. The choice of '3' will call the SCREEN - 1.1.3 for select type of the general ledger maintenance.

ERRORS RECOVERY:

The user can only enter '*', '1', '2' or '3' values. Otherwise, the error message ## <1> ## will be shown and the cursor will move to the enter position for reentering the selection. The response must be correct, otherwise the system will request again.

DISPLAY MESSAGES AND PROMPTS ::

### <1> ### : ERROR MESSAGE: INVALID RESPONSE. PLEASE REENTER
ACCOUNT UPDATE SELECTION

SELECT ONE OF THE FOLLOWING TYPE TO PROCESS
ENTER SELECTION: _

* : FOR EXIT ACCOUNT UPDATE PROCESS
1 : UPDATE GENERAL LEDGER FILE
2 : CLEAN REVENUES AND EXPENSE
3 : CLEAN OLD ENTRIES
4 : END OF YEAR PROCESS

PURPOSE:

This screen is shown through the selection '2' of SCREEN - 1.0 or back from its previous called screen. It has five different functions to choose from: update general ledger file, clean revenues and expense, clean old entries, end of year process and exit this process back to the calling SCREEN - 1.0.

No further screen for this screen's selection are available. Except for the '*' selection, any of the other four selections will show that the selected function is in progress at the bottom of the screen.

The choice of '1' will update the general ledger file. If the accounts are not balanced the error message will be displayed and the update will not be executed. Usually, it will be requested after the trial balance is printed and the balance of accounts is approved.

The choice of '2' will clean the revenues and expense accounts. The amount of revenues and expense account will be set to zero and the total income will be moved to the retain earning account for the beginning of the next accounting period. Usually, this selection is chosen after the income statement has been printed.
The choice of '3' will clean the old entries which are entered during a period of the accounting cycle. Usually, it is processed after any other accounting steps are finished and prepared for the beginning of the next accounting period. If the entered accounts are not used to update the general ledger file, an error message will be displayed. The requested function will not be executed. The choice of '4' will clean the entered account file, which keeps records of entered accounts of a accounting period, and the previous entered account file, which keeps all the records for a fiscal year.

ERRORS RECOVERY:

The user can only enter '*', '1', '2' or '3' values. Otherwise, the error message ## <1> ## will be shown and the cursor will move to the enter position for reentering the selection. The response must be correct, otherwise the system will request again.

DISPLAY MESSAGES AND PROMPTS ::

### <1> ### : ERROR MESSAGE::INVALID RESPONSE. PLEASE REENTER
### <2> ### : UPDATE GENERAL LEDGER FILE PROCESSING
### <3> ### : CLEAN REVENUES AND EXPENSE PROCESSING
### <4> ### : CLEAN OLD ENTRIES PROCESSING
### <5> ### : END OF YEAR PROCESSING
### <6> ### : ACCOUNT NOT UPDATE YET. NO CLEAN MADE
### <7> ### : ACCOUNT NOT BALANCE. NO UPDATE
REPORT GENERATOR SELECTION
SELECT ONE OF THE FOLLOWING TYPE TO PROCESS
ENTER SELECTION: _

* : FOR EXIT REPORT GENERATOR PROCESS
1 : PRINT CHART OF ACCOUNT
2 : PRINT LIST OF ACCOUNT ENTERED
3 : PRINT AUDIT TRIAL
4 : PRINT LEDGER SHEET
5 : PRINT TRIAL BALANCE
6 : PRINT INCOME STATEMENT
7 : PRINT BALANCE SHEET

PURPOSE:

This screen is shown through the selection '3' of SCREEN - 1.0. It allows the user to print the following reports: the chart of account, the list of account entered, the audit trial, the ledger sheet, the trial balance, the income statement and balance sheet.

The choice of '1' will print the chart of account. The chart is used to show how many accounts are in the general ledger file. Before modifying any of the general ledger account, it is important to reference this report to make sure the modification is correct.

The choice of '2' will print a list of entered accounts for the date which the entry is made. A hard copy will be printed for each entered account for future reference. Usually, after enter all the accounts, this listing can be requested.

The choice of '3' will print the audit trail for a certain period during the fiscal year. After selecting this process, the system will prompt the user to enter beginning and ending date of that period for this report. The dates the user entered must be within the fiscal year, otherwise the system will request to reenter again.
The choice of '4' will print the ledger sheet for each account which occurred during the period. It lists all transactions for that accounts and total change.

The choice of '5' will print the trial balance for the current accounts. It is a list of all accounts used by the business and the amount in each account. It is requested after finishing the entry and before updating the general ledger file.

The choice of '6' will print the income statement. It shows the ultimate effect on the business of its profit-seeking activities during a period.

The choice of '7' will print the balance sheet. It shows the balance of the asset and liability accounts for a business at the end of a period.

ERRORS RECOVERY:
The user can only enter '* ', '1', '2', ..., '6' and '7' values. Otherwise, the error message ## <1> ## will be shown and the cursor will move to the enter position for reentering the selection. The response must be correct otherwise the system will request again.

If the audit trail is request, the system will requested the beginning date and ending date for the desired period. If the date entered is not correct the cursor will move back for request reenter.

DISPLAY MESSAGES AND PROMPTS ::

### <1> ### :: ERROR MESSAGE::INVALID RESPONSE. PLEASE REENTER
### <2> ### :: PRINT CHART OF ACCOUNT IS PROCESSING
### <3> ### :: PRINT LIST OF ACCOUNT ENTERED IS PROCESSING
### <4> ### :: PRINT AUDIT TAIL IS PROCESSING
### <5> ### :: PRINT LEDGER SHEET IS PROCESSING
### <6> ### :: PRINT TRIAL BALANCE IS PROCESSING
### <7> ### :: PRINT INCOME STATEMENT IS PROCESSING
### <8> ### :: PRINT BALANCE SHEET IS PROCESSING
### <9> ### :: ENTER BEGIN DATE:YEAR __ MONTH __ DATE __
### <10>### :: ENTER END DATE:YEAR __ MONTH __ DATE __
### <11>### :: YEAR NOT IN PROPER RANGE PLEASE REENTER
### <12>### :: ENTER TOO LONG PLEASE REENTER
### <13>### :: MONTH NOT WITHIN PROPER RANGE
### SCREEN - 1.1.1

**ENTER ENTRY PROCESS**

<table>
<thead>
<tr>
<th><strong>DATE</strong> <strong>/</strong>/**</th>
<th><strong>SOURCE CODE</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ENTRY NUMBER</strong></td>
<td></td>
</tr>
<tr>
<td><strong>ACCOUNT NUMBER</strong></td>
<td></td>
</tr>
<tr>
<td><strong>ACCOUNT NAME</strong></td>
<td>++++++++++++++++++++</td>
</tr>
<tr>
<td><strong>ACCOUNT VALUE</strong></td>
<td>________________</td>
</tr>
</tbody>
</table>

#### PURPOSE:

This screen is shown through the selection '1' of SCREEN - 1.1. It moves the cursor around the screen to request the user to enter proper data of the entry. If the entered data is incorrect, the system will display an error message for reentering. After finishing each entry, the user can answer the 'N' to go back to the original SCREEN - 1.1 or 'Y' to reenter more entries with "enter more" prompt.

#### ERRORS RECOVERY:

For response to "source code" prompt, only '1', '2' or '3' will be accepted. Otherwise, the system will display the error message for reentering. The system will automatically display the current entry number on the screen which the user does not have to worry about. The user must enter the correct account number, otherwise the user will be asked to try again or not. Be sure the length and type of the account to be entered is correct. The amount needs to be entered correctly also. Illegal values will not be accepted and needs to be reenter. After entering each entry, the user may decide to enter more data or not.

#### DISPLAY MESSAGES AND PROMPTS:

```plaintext
### <1> ### ERROR MESSAGE: INVALID RESPONSE. PLEASE REENTER
### <2> ### TRY AGAIN (Y OR N) ?
### <3> ### ENTER MORE (Y OR N) ?
```
PURPOSE:

This screen is shown through the selection '2' of SCREEN - 1.1. It moves the cursor around the screen to request the user to enter proper data of the adjusting entry. If the entered data is incorrect, the system will display the error message for reentering. After finishing the entry, the user can answer the 'N' to go back to the original SCREEN - 1.1.1 or 'Y' to reenter more entries with "enter more" prompt. This screen functions all the same as the SCREEN - 1.1.1, except the "source code" '4'. For detail, please reference the SCREEN - 1.1.1.
PURPOSE:

This screen is shown through the selection '3' of SCREEN - 1.1 or back from its previous called screen. It has five different functions to choose from: add general ledger record, delete general ledger record, modify the current general ledger record or exit back to the calling SCREEN - 1.1.

The choice of '1' will go to SCREEN - 1.1.3.1 for adding a new general ledger record. The choice of '2' will go to SCREEN - 1.1.3.2 for deleting a general ledger record. The choice of '3' will go to SCREEN - 1.1.3.3 for modifying a general ledger record. The choice of '*' will back to the original SCREEN - 1.1.3.

ERRORS RECOVERY:

The user can only enter '*', '1', '2' or '3' values. Otherwise, the error message ## <1> ## will be shown and the cursor will move to the enter position for reentering the selection. The response must be correct, otherwise the system will request again.

DISPLAY MESSAGES AND PROMPTS ::

### <1> ### : ERROR MESSAGE: INVALID RESPONSE. PLEASE REENTER
PURPOSE:

This screen is shown through the selection '1' of SCREEN - 1.1.3. It moves the cursor around the screen to request the user to enter proper data for the new adding account. If the entered data is incorrect, the system will display the error message for reentering. After finishing the entry, the system will ask for confirmation. The user is requested to recheck the data to make sure the added data is correct. The account value of the added account must be zero. Thus, the user will not be requested to enter the account value.

ERRORS RECOVERY:

The new account number must not exist on the current system, otherwise the system will display the error message and request for reentering. The length of account must be correct too. The system only accepts alphabetic letters for the new account name. Only the '0', '1', '2' and '3' is allows for the account type. Any above errors, the user will be requested to reenter.

DISPLAY MESSAGES AND PROMPTS ::

### <1> ### :: ERROR MESSAGE::INVALID RESPONSE. PLEASE REENTER
### <2> ### :: ACCOUNT ERROR:: WANT TRY AGAIN? ENTER Y OR N:
### <3> ### :: ACTION CONFIRM :: IS THIS RECORD TO BE ADDED? ENTER Y OR N
### <4> ### :: NO ADDITION MADE
### <5> ### :: ADD MORE ? (Y OR N)
DELETE GENERAL LEDGER RECORD

ENTERED ACCOUNT NUMBER
WHICH TO BE DELETED : ______

ACCOUNT NAME : +++++++++++++++++++++++
ACCOUNT TYPE : +
ACCOUNT VALUE : ++++++++.++

##### <4>,<5> #####

############ <1>,<2>,<3> ###############

PURPOSE:

This screen is shown through the selection '2' of SCREEN - 1.1.3. The user will be asked to enter the account number, which will be deleted from the general ledger file. After the proper account number is entered, the system will display the information about that account and ask for confirmation. The account value of those deleted accounts must be zero, otherwise the deletion will not be executed.

ERRORS RECOVERY:

The deleting account number must exist on the current system, otherwise the system will display the error message and request for reentering. The system will not delete an account for which the account value is not zero.

DISPLAY MESSAGES AND PROMPTS :

### <1> ### :: ERROR MESSAGE::INVALID RESPONSE. PLEASE REENTER
### <2> ### :: ACCOUNT ERROR:: WANT TRY AGAIN? ENTER Y OR N:
### <3> ### :: ACTION CONFIRM :: IS THIS RECORD TO BE DELETE? ENTER Y OR N
### <4> ### :: NO DELETION MADE
### <5> ### :: DELETE MORE ? (Y OR N)
SCREEN 1.1.3.3
MODIFY GENERAL LEDGER RECORD

ENTERED ACCOUNT NUMBER : ______

ENTERED ACCOUNT NAME : ++++++++++++++++++++++

ACCOUNT TYPE :

ACCOUNT VALUE :

############################ <4>,<5> ######

############################ <1>,<2>,<3> #######

PURPOSE:

This screen is shown through the selection '3' of SCREEN - 1.1.3. The user will be asked to enter the account number which will be modified. Only the account name can be changed. Thus, after the desired account number is entered correctly, the system will display the information about that account and ask for the new account name for that account.

ERRORS RECOVERY:

The modified account number must exist on the current system, otherwise the system will display the error message and request for reentering. The system will only accepts the account name which is less than thirty letters.

DISPLAY MESSAGES AND PROMPTS ::

### <1> ### : ERROR MESSAGE::INVALID RESPONSE. PLEASE REENTER
### <2> ### : ACCOUNT ERROR:: WANT TRY AGAIN? ENTER Y OR N:
### <3> ### : ACTION CONFIRM :: IS THIS RECORD TO BE MODIFY? ENTER Y OR N
### <4> ### : NO MODIFY MADE
### <5> ### : MODIFY MORE ? (Y OR N)
### APPENDIX D-1

#### CHART OF ACCOUNTS

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>100000</td>
<td>ASSETS</td>
</tr>
<tr>
<td>110000</td>
<td>CURRENT ASSETS</td>
</tr>
<tr>
<td>111000</td>
<td>CASH</td>
</tr>
<tr>
<td>111100</td>
<td>REVENUE BANK</td>
</tr>
<tr>
<td>111150</td>
<td>PETTY CASH</td>
</tr>
<tr>
<td>111990</td>
<td>TOTAL</td>
</tr>
<tr>
<td>114000</td>
<td>ACCOUNT RECEIVABLE</td>
</tr>
<tr>
<td>114100</td>
<td>TRADE ACCOUNTS RECEIVABLE</td>
</tr>
<tr>
<td>114120</td>
<td>FINISHED GOOD SALES RECEIVABLE</td>
</tr>
<tr>
<td>114140</td>
<td>CONSULTING FEES RECEIVABLE</td>
</tr>
<tr>
<td>114190</td>
<td>ALLOW FOR DOUBTFUL ACCOUNTS</td>
</tr>
<tr>
<td>114191</td>
<td>TOTAL</td>
</tr>
<tr>
<td>114290</td>
<td>EMPLOYEE RECEIVABLE</td>
</tr>
<tr>
<td>114295</td>
<td>OTHER ACCOUNTS RECEIVABLE</td>
</tr>
<tr>
<td>114400</td>
<td>TOTAL</td>
</tr>
<tr>
<td>114600</td>
<td>INVENTORY-FINISHED GOODS</td>
</tr>
<tr>
<td>114610</td>
<td>FINISHED GOODS</td>
</tr>
<tr>
<td>114620</td>
<td>RAW MATERIALS</td>
</tr>
<tr>
<td>114630</td>
<td>OTHER INVENTORY</td>
</tr>
<tr>
<td>114640</td>
<td>TOTAL</td>
</tr>
<tr>
<td>118000</td>
<td>PREPAID EXPENSE</td>
</tr>
<tr>
<td>118100</td>
<td>PREPAID INSURANCE</td>
</tr>
<tr>
<td>118200</td>
<td>PREPAID TAXES</td>
</tr>
<tr>
<td>118300</td>
<td>OTHER PREPAID EXPENSES</td>
</tr>
<tr>
<td>118400</td>
<td>TOTAL</td>
</tr>
<tr>
<td>119000</td>
<td>CONSULTING</td>
</tr>
<tr>
<td>119100</td>
<td>CONTRACT-IN-PROCESS</td>
</tr>
<tr>
<td>119200</td>
<td>DIRECT LABOR</td>
</tr>
<tr>
<td>119300</td>
<td>OVERHEAD</td>
</tr>
<tr>
<td>119350</td>
<td>OTHER DIRECT CHARGE</td>
</tr>
<tr>
<td>119352</td>
<td>DIRECT MATERIALS</td>
</tr>
<tr>
<td>119354</td>
<td>OUTSIDE SERVICE</td>
</tr>
<tr>
<td>119356</td>
<td>FREIGHT - IN</td>
</tr>
<tr>
<td>119358</td>
<td>TRAVEL</td>
</tr>
<tr>
<td>119600</td>
<td>TOTAL</td>
</tr>
<tr>
<td>119700</td>
<td>CREDITS</td>
</tr>
<tr>
<td>119800</td>
<td>TOTAL</td>
</tr>
<tr>
<td>119900</td>
<td>TOTAL CURRENT ASSETS</td>
</tr>
<tr>
<td>130000</td>
<td>DEFERRED PRODUCTION COSTS</td>
</tr>
<tr>
<td>130100</td>
<td>DIRECT LABOR</td>
</tr>
<tr>
<td>130200</td>
<td>OVERHEAD</td>
</tr>
<tr>
<td>130400</td>
<td>OTHER DIRECT CHARGES</td>
</tr>
<tr>
<td>130410</td>
<td>DIRECT MATERIALS</td>
</tr>
<tr>
<td>130440</td>
<td>OUTSIDE SERVICES</td>
</tr>
<tr>
<td>130460</td>
<td>FREIGHT - IN</td>
</tr>
<tr>
<td>130490</td>
<td>TOTAL</td>
</tr>
<tr>
<td>130900</td>
<td>CREDITS</td>
</tr>
<tr>
<td>130990</td>
<td>TOTAL</td>
</tr>
<tr>
<td>150000</td>
<td>PROPER &amp; EQUIPMENT</td>
</tr>
</tbody>
</table>

```
<table>
<thead>
<tr>
<th>Item</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>151000COST</td>
<td>30000000000000000000000000</td>
</tr>
<tr>
<td>151200MACHINERY &amp; EQUIPMENT</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>151300FURNITURE &amp; FIXTURE</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>151400LEASEHOLD IMPROVEMENT</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>151500TOTAL</td>
<td>20000000000000000000000000</td>
</tr>
<tr>
<td>152000CONSTRUCTION-IN-PROCESS</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>153000ACCUMULATED DEPRECIATION</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>155000MACHINERY &amp; EQUIPMENT</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>156000FURNITURE AND FIXTURE</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>157000LEASEHOLD IMPROVEMENT</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>158000TOTAL</td>
<td>20000000000000000000000000</td>
</tr>
<tr>
<td>158500TOTAL PROPERTY &amp; EQUIPMENT</td>
<td>20000000000000000000000000</td>
</tr>
<tr>
<td>160000TOTAL ASSETS</td>
<td>20000000000000000000000000</td>
</tr>
<tr>
<td>200000LIABILITIES &amp;</td>
<td>10000000000000000000000000</td>
</tr>
<tr>
<td>200010STOCKHOLDER'S EQUITY</td>
<td>30000000000000000000000000</td>
</tr>
<tr>
<td>200020CURRENT LIABILITIES</td>
<td>30000000000000000000000000</td>
</tr>
<tr>
<td>210000NOTES PAYABLE</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>211000CURRENT MATUR ON L-T DEBT</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>212000SALES TAX PAYABLE</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>213000INCOME TAX PAYABLE</td>
<td>30000000000000000000000000</td>
</tr>
<tr>
<td>213100FEDERAL</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>213200STATE</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>214000TOTAL</td>
<td>20000000000000000000000000</td>
</tr>
<tr>
<td>215000PAYROLL TAXES PAYABLY</td>
<td>30000000000000000000000000</td>
</tr>
<tr>
<td>215100FEDERAL INCOME TAX WITHHOLD</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>215110FICA</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>215120FEDERAL UNEMPLOYMENT TAX</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>215130STATE INCOAXESAX WITHHOLD</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>215140SDI</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>215150STATE UNEMPLOYMENT TAX</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>216000TOTAL</td>
<td>20000000000000000000000000</td>
</tr>
<tr>
<td>217000ACCRUED LIABILITIES</td>
<td>30000000000000000000000000</td>
</tr>
<tr>
<td>218000PAYROLL</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>218100VACATION</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>218200PROPERTY TAXES</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>218300OTHER ACCRUED LIABILITIES</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>218900TOTAL</td>
<td>20000000000000000000000000</td>
</tr>
<tr>
<td>221000UNNEARNED INCOME</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>221100CUSTOMER OVERPAYMENT</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>221200UNIDENTIFIED CUSTOMER RECEIPTS</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>221400TOTAL CURRENT LIABILITIES</td>
<td>20000000000000000000000000</td>
</tr>
<tr>
<td>232000LONG TERM LIABILITIES</td>
<td>30000000000000000000000000</td>
</tr>
<tr>
<td>232200NOTES PAYABLE</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>232300DEFERRED INCOME TAXES PAYABLE</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>232400TOTAL</td>
<td>20000000000000000000000000</td>
</tr>
<tr>
<td>242000STOCKHOLDER'S EQUITY</td>
<td>30000000000000000000000000</td>
</tr>
<tr>
<td>242100CAPITAL STOCK</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>242200ADDITIONAL PAID-IN CAPITAL</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>243200RETAINAL EARNINGS</td>
<td>00000000000000000000000000</td>
</tr>
<tr>
<td>243300TOTAL</td>
<td>20000000000000000000000000</td>
</tr>
<tr>
<td>249990TOTAL LIAB &amp; STOCKHOLDER'S EQ</td>
<td>20000000000000000000000000</td>
</tr>
<tr>
<td>300000INCOME</td>
<td>10000000000000000000000000</td>
</tr>
<tr>
<td>305000INCOME DETAIL</td>
<td>30000000000000000000000000</td>
</tr>
<tr>
<td>Account Number</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>310000</td>
<td>Sale of Good</td>
</tr>
<tr>
<td>311000</td>
<td>Finished Goods</td>
</tr>
<tr>
<td>312200</td>
<td>Sales Returns &amp; Allowances</td>
</tr>
<tr>
<td>312300</td>
<td>Total</td>
</tr>
<tr>
<td>320000</td>
<td>Consulting Fees</td>
</tr>
<tr>
<td>330000</td>
<td>Royalties Fees</td>
</tr>
<tr>
<td>331000</td>
<td>Patent Royalties</td>
</tr>
<tr>
<td>332000</td>
<td>Consulting Royalties</td>
</tr>
<tr>
<td>333000</td>
<td>Total</td>
</tr>
<tr>
<td>340000</td>
<td>Other Income</td>
</tr>
<tr>
<td>341000</td>
<td>Collection of Bad Debts.</td>
</tr>
<tr>
<td>342000</td>
<td>Miscellaneous Income</td>
</tr>
<tr>
<td>349000</td>
<td>Total</td>
</tr>
<tr>
<td>350000</td>
<td>Total Income</td>
</tr>
<tr>
<td>400000</td>
<td>EXPENSES</td>
</tr>
<tr>
<td>405000</td>
<td>EXPENSES DETAIL</td>
</tr>
<tr>
<td>410000</td>
<td>COST OF INCOME</td>
</tr>
<tr>
<td>411000</td>
<td>COST OF SALE OF GOODS</td>
</tr>
<tr>
<td>412000</td>
<td>COST OF CONSULTING FEES</td>
</tr>
<tr>
<td>413000</td>
<td>ROYALTY PAYMENTS</td>
</tr>
<tr>
<td>414000</td>
<td>VARIANCE EXPENSE</td>
</tr>
<tr>
<td>415000</td>
<td>Total</td>
</tr>
<tr>
<td>420000</td>
<td>DEVELOPMENT PROJECT EXPENSES</td>
</tr>
<tr>
<td>421000</td>
<td>DIRECT LABOR</td>
</tr>
<tr>
<td>422000</td>
<td>OVERHEAD</td>
</tr>
<tr>
<td>423000</td>
<td>OTHER DIRECT CHARGE</td>
</tr>
<tr>
<td>424000</td>
<td>DIRECT MATERIALS</td>
</tr>
<tr>
<td>424100</td>
<td>OUTSIDE SERVICES</td>
</tr>
<tr>
<td>424200</td>
<td>FREIGHT-IN</td>
</tr>
<tr>
<td>424300</td>
<td>TRAVEL</td>
</tr>
<tr>
<td>424400</td>
<td>Total</td>
</tr>
<tr>
<td>424500</td>
<td>TOTAL DEVELOPMENT PROJ.EXPENSES</td>
</tr>
<tr>
<td>431000</td>
<td>SALARIES, WAGES, &amp; BENEFITS</td>
</tr>
<tr>
<td>432000</td>
<td>DIRECT LABOR</td>
</tr>
<tr>
<td>432200</td>
<td>INDIRECT LABOR</td>
</tr>
<tr>
<td>432300</td>
<td>OVERTIME PREMIUM</td>
</tr>
<tr>
<td>432400</td>
<td>HOLIDAY-VACATION-SICK</td>
</tr>
<tr>
<td>432500</td>
<td>JURY DUTY &amp; OTHER LEAVE</td>
</tr>
<tr>
<td>432600</td>
<td>PAYROLL TAXES</td>
</tr>
<tr>
<td>432700</td>
<td>BONUS</td>
</tr>
<tr>
<td>432800</td>
<td>OTHER FRINGES BENEFITS</td>
</tr>
<tr>
<td>433000</td>
<td>Total</td>
</tr>
<tr>
<td>440000</td>
<td>MATERIALS &amp; SUPPLIES</td>
</tr>
<tr>
<td>441000</td>
<td>COMPUTER MATERIALS AND SUPPLIES</td>
</tr>
<tr>
<td>441200</td>
<td>RAW MATERIALS &amp; SUPPLIES</td>
</tr>
<tr>
<td>441300</td>
<td>STATIONAL-MAT'LS AND SUPPLIES</td>
</tr>
<tr>
<td>441400</td>
<td>OTHER MATERIALS AND SUPPLIES</td>
</tr>
<tr>
<td>441500</td>
<td>TOTAL</td>
</tr>
<tr>
<td>442000</td>
<td>FACILITIES &amp; EQUIPMENT</td>
</tr>
<tr>
<td>442100</td>
<td>RENT-OFFICE SPACE</td>
</tr>
<tr>
<td>442200</td>
<td>EQUIPMENT LEASES</td>
</tr>
<tr>
<td>442300</td>
<td>DEPRECIATION</td>
</tr>
<tr>
<td>442400</td>
<td>EQUIPMENT MAINTAINCE</td>
</tr>
<tr>
<td>Code</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>------------------------------------</td>
</tr>
<tr>
<td>442500</td>
<td>JANITORIAL SERVICE</td>
</tr>
<tr>
<td>442600</td>
<td>UTILITIES</td>
</tr>
<tr>
<td>442700</td>
<td>AUTO EXPENS</td>
</tr>
<tr>
<td>442800</td>
<td>EQUIPMENT RENTAL</td>
</tr>
<tr>
<td>442900</td>
<td>TOTAL</td>
</tr>
<tr>
<td>443000</td>
<td>OUTSIDE SERVICES</td>
</tr>
<tr>
<td>443100</td>
<td>PROFESSIONAL SERVICES-LEGAL</td>
</tr>
<tr>
<td>443200</td>
<td>PROF. SERVICES-CONSULTING</td>
</tr>
<tr>
<td>443300</td>
<td>EMPLOYMENT SERVICES</td>
</tr>
<tr>
<td>443400</td>
<td>CREDIT &amp; COLLECTION SERVICE</td>
</tr>
<tr>
<td>443500</td>
<td>OTHER OUTSIDE SERVICES</td>
</tr>
<tr>
<td>443600</td>
<td>TOTAL</td>
</tr>
<tr>
<td>444000</td>
<td>ADVERTISING</td>
</tr>
<tr>
<td>444100</td>
<td>PERIODICALS</td>
</tr>
<tr>
<td>444200</td>
<td>TRADE SHOWS</td>
</tr>
<tr>
<td>444300</td>
<td>SHOWS - TRAVEL</td>
</tr>
<tr>
<td>444400</td>
<td>SHOWS - OTHER</td>
</tr>
<tr>
<td>450000</td>
<td>DIRECT MAILINGS - SHAMPLES</td>
</tr>
<tr>
<td>451000</td>
<td>BROUCHURE PRINTING AND MAILING</td>
</tr>
<tr>
<td>452000</td>
<td>OUTSIDE PREP. WORK ON ADS</td>
</tr>
<tr>
<td>453000</td>
<td>OTHER ADVERTISING</td>
</tr>
<tr>
<td>454000</td>
<td>TOTAL ADVERTISING</td>
</tr>
<tr>
<td>460000</td>
<td>ORDER &amp; SHIPPING EXPENSES</td>
</tr>
<tr>
<td>460100</td>
<td>SHIPPING EXPENSE - OUTGOING</td>
</tr>
<tr>
<td>460200</td>
<td>ORDER/SHIPPING FORMS PRINTING</td>
</tr>
<tr>
<td>460300</td>
<td>POSTAGE</td>
</tr>
<tr>
<td>460900</td>
<td>TOTAL</td>
</tr>
<tr>
<td>465000</td>
<td>OTHER EXPENSES</td>
</tr>
<tr>
<td>465010</td>
<td>TAXES-LICENSES-FEES</td>
</tr>
<tr>
<td>465020</td>
<td>INSURANCE</td>
</tr>
<tr>
<td>465030</td>
<td>TRAVEL &amp; EMPLOYEE BUSINESS EXP</td>
</tr>
<tr>
<td>465040</td>
<td>COMMUNICATIONS</td>
</tr>
<tr>
<td>465050</td>
<td>DOUBTFUL ACCOUNTS</td>
</tr>
<tr>
<td>465060</td>
<td>DUES AND SUBSCRIPTIONS</td>
</tr>
<tr>
<td>465070</td>
<td>OTHER FREIGHT - IN</td>
</tr>
<tr>
<td>465080</td>
<td>ALLOW'L CUSTOMER UPDERAPMENT</td>
</tr>
<tr>
<td>465090</td>
<td>MISCELLANEOUS</td>
</tr>
<tr>
<td>465900</td>
<td>TOTAL</td>
</tr>
<tr>
<td>470000</td>
<td>CRDITS</td>
</tr>
<tr>
<td>470100</td>
<td>DIRECT LABOR TRANSFERRED OUT</td>
</tr>
<tr>
<td>470200</td>
<td>OVERHEAD APPLIED</td>
</tr>
<tr>
<td>470300</td>
<td>EST. O'HEAD UNDER ABSORBED</td>
</tr>
<tr>
<td>470400</td>
<td>TOTAL</td>
</tr>
<tr>
<td>475000</td>
<td>NON-OPERATION (INCOME) EXPENSE</td>
</tr>
<tr>
<td>475020</td>
<td>LOSS ON PROP./EQUIP. SALE</td>
</tr>
<tr>
<td>475030</td>
<td>INTEREST (INCOME) EXPENSE</td>
</tr>
<tr>
<td>479000</td>
<td>TOTAL</td>
</tr>
<tr>
<td>480000</td>
<td>ESTIMATED INCOME TAXES</td>
</tr>
<tr>
<td>480100</td>
<td>FEDERAL</td>
</tr>
<tr>
<td>480200</td>
<td>STATE</td>
</tr>
<tr>
<td>489000</td>
<td>TOTAL</td>
</tr>
<tr>
<td>495000</td>
<td>TOTAL EXPENSES</td>
</tr>
</tbody>
</table>
EXAMPLE FILES NEEDED TO RUN THE PCL

Following files are needed in the directory for the using of PCL.

**********
LOGIN.CMD:
**********

copy cmd.cmd comand.cmd
<PCL-EXEC>EXEC.EXE

********
CMD.CMD:
********

refuse links
Set def decl /noconfirm
dcl pcl psl:<cs.grad.ccw>init.pcl
ter vt52
ter no pause end-of-page
def system:ps:<pcl-exec>exec.exe,system:
del ps:<cs.grad.ccw>comand.cmd
take

**********
INIT.PCL:
**********

command ledger;
begin
docommand original "ter wid 0";
docommand original "run ledger.exe";
docommand original "ter wid 80";
end;
SET UP THE FILES

Before using this system, to initialize this system for use, make sure the empty sequential files for account.fil (entered account file), previ.fil (previous entered account file) and tempac.tmp (temporary file) is created in the system directory. The following commands build those sequential files. Be sure the empty files' filename and extension are exact the same as mention above.

```
@EDIT (FILENAME.EXT) <CR>
00100 <ESC>
*EU <CR>
```

A sequential file of the chart of accounts is needed to build the genfil.inx (general ledger file), an index file, through the use of Indexed Sequential File Maintenance program (ISAM) on the DEC-2060 system. A sample of the chart of accounts which is used to build the index file is listed at appendix d-1. The following commands are a example of the using of the ISAM in the DEC-2060. In this case, the file name of the chart of accounts is genfil.seq.

```
@ISAM <CR>
*GENFIL.INX,GENFIL.IND=GENFIL.SEQ/B <CR>
Mode of input file: A <CR>
Mode of data file: A <CR>
Maximum record size: 67 <CR>
Key descriptor: UN1.6 <CR>
Records per input block: 0 <CR>
Total records per data block: 42 <CR>
Empty records per data block: 21 <CR>
Total entries per index block: 42 <CR>
Empty entries per index block: 21 <CR>
Percentage of data file to leave empty: 20 <CR>
Percentage of index file to leave empty: 20 <CR>
Maximum number of records file can became: 500 <cr>
```

For more detail in using ISAM, please check the help file in DEC-2060, simply typing in "help ISAM" when in monitor or asking the computer center consult.