THE DESIGN AND IMPLEMENTATION OF A SYSTEM TO BE USED IN THE AUDITING PROCESS

Lee, Dongwon

Table of Contents

抄　　錄

本 硏究의 目的은 企業體內의 自體統制의 完璧性을 監査하는데 使用될 system을 開發하는데에 있다. 이 system은 企業體內의 諸般活動과 文書處理에 對한 統制들을 內包하여 社內監査에 있어 必要한 情報를 얻을 수 있도록 한다. 本 硏究는 購買活動의 監査에 局限하고 있으나 提示된 槪念을 擴張하여 組織內의 全般活動에 까지 應用이 可能하다.

이 system을 開發하는 데에는 優先 組織內의 제반 통제들을 定義하여야 하는 바, 이 統制들은 組織體의 目的을 達成하는데에 必要한 諸般活動에 關聯되어야 하고 外部의 指針과 會計實務原則에 合當해야만 한다. 그러나 組織體의 特性에 따라 統制情報를 內包하고 있는 Knowledge Base의 內容이 多少 달라지게 마련이다.

ABSTRACT

This paper intends to develop a system that can be used by an internal auditor in his evaluation of the completeness of the internal controls within the organization. This system will provide a repository of necessary controls that the auditor will be able to address and receive information concerning the various activities and documents within the organization. The scope of this research is limited to the examination of the purchasing cycle of an organization. However, it would be very well possible to expand the concepts set forth to include a whole organization.

In order to accomplish this objective, it is necessary to define the controls relevant to the activities required to achieve the goal of the organization, to the external guidelines which affect the activities, and to acceptable accounting practices.

After these controls are defined, it will be necessary to store them in a knowledge base that will allow them to be retrieved as required. This knowledge base should allow for the controls to be displayed for each component of the activities under investigation and should be composed of contingent factors which will be influenced by the organizational objectives as set forth by the management. To allow for this constraint, it is not possible to generate a knowledge base that would be common to all organizations.

I. INTRODUCTION

To insure that internal controls are being properly maintained, it is necessary for an organization to provide a means of auditing the internal activities taking place within the organization. This is often done by a part of the organization that is collectively known as the internal auditior. This auditor, having the primary function of evaluating controls, verifying their implementation, and providing management with his conclusions, may also present to management constructive recommendations with his conclusions if he is properly qualified.[1]

The junior auditor may find that he is lacking the necessary experience and training to enable him to fulfill this primary function. This is not to imply that the junior auditor is not properly qualified, but that it is the result of him not having the experience possessed by a senior auditor. Often, a senior auditor will use his experience and rely more on his ability to sense problems than on other fact gathering techniques.[2]

I.I DATA BASE AS A REROSITORY OF INTERNAL CONTROL

One solution to this dilemma, as proposed by Perry[3], is to use a data base to create a repository of internal control knowledge. By identifying and storing the experiences of senior auditors in a data base, it would be possible to consolidate, analyze, restructure, and use them as input to the audit process.

The data base, acting as the experienced auditor's memory, would remember situations, conditions, and events that occur in the organization. This remembrance could then be used as reference or in the identification and classification of a reoccurring problem.

The data stored in the data base would contain historical information from previous audits along with suggestions and recommendations proposed for future audits. It would be necessary to develop a schema that would allow the stored information to be manipulated, analyzed and retrieved. The auditors would then use this information in various ways as they examine the activities of the organization, evaluate their findings, and plan future audits.

---

1) Mair, W. C., Wood, D. R., and Davis, K. W., *Computer Control and Audit,* Q. E. D. Information Sciences, The Institute of Internal Auditors, Inc., Altomont Springs, Florida, 1978.
2) Perry, W., "Creating and Using an Audit Data Base", *EDP Auditing,* Auerbach Publishers Inc., June/July 1982.
3) Perry, W., idid.

I.2 TICOM-II

Another system that has been developed to assist the auditor in his evaluation process is described by Bailey, et al.[4]

This system, known as TICOM-II, examines the adequacy of internal controls, and provides a cost/benefit analysis of these controls. TICOM-II utilizes auditing criteria and a concept approach to model the firm's internal behavior. This concept is known as Office Information Systems theory.

The first of four components of TICOM-II is a modeling language(Internal Control Description Language: ICDL) that was designed to support descriptive specifications of accounting control systems.

Using ICDL, TICOM-II maps the system into an internal representation. This representation, the second component of TICOM-II, allows for querying processing and simplifies system analysis.

The third component of TICOM-II is the analysis of the internal represention. This analysis technique allows the auditor to divide the system under investigation into subsystems. This is allowable since the technique stems from graph theory and traditional accounting methods. By dividing the system into subsystems, TICOM-II is able to produce a subgraph from the original ICDL model.

The query processing system, the last component of TICOM-II, allows for linkage of stored items to instructions involving those items. These instructions are then processed to produce a simplified subsystem that is very similar to the original system. As a result of this, the conditions that are required by the query are generated and can be examined closer.

I.3 ENTITY RELATIONSHIP EVALUATOR

Lieberman[5] also provides a computerized system to be used by the auditor.

This proposed system, using an entity-relationship data base, is directed to the study and evaluation of the internal controls of the organization. Lieberman's system provides assistance to the auditor by allowing him to test the plan of internal control possessed by an organization. In order to provide this assistance, the organization's plan or internal control must first be stored in

4) Bailey, Jr. A. D., Gerlach, J., McAfee, R. P., Whinston, A. B., "Internal Accounting Controls in the Office of the Future", *IEEE*, May 1981, pp. 59-69.
5) Lieberman, A. Z., "Mechanized Analysis of Entity-Relationship Design Databases", *Entity-Relationship Approach to Systems Analysis and Design*. North-Holland Publishing Co., 1980.

a "computer-accessible form."

In preparing the plan for storage, the auditor must define, for the computer system, the types of entities, relationships, and attributes that are to be used. After these have been defined, the auditor may then use the traditional auditing method of flowcharting to pictorially describe the process. This provides a way of checking the completeness and correctness of the description of the system and allows the auditor to observe the entire system and to see the direct interaction among the flow of processes and information that is generated. After the flowchart is determined to be complete, it is then necessary to translate it into a data base using an entity relationship language.

Upon the achievement of a "satisfactory construction," the auditor may then evaluate the plan of internal control. This evaluation is performed by testing the modeled system with "a set of audit rules" that, as stated by Lieberman, "must be a creation of the auditor, because the rules represent the auditor's personal evaluation criteria."

The result of these tests should provide insight into the possible weaknesses of the internal controls. If weaknesses are indicated by the evaluation, the auditor must determine if the controls are actually weak or if an incorrect rule, or incomplete model has been employed. For this reason, Lieberman states that "the auditor cannot be replaced."

I.4 DATA DICTIONARY/DIRECTORY SYSTEM

Along with Lieberman's ERE, there is another software system that provides valuable audit information to the auditor. This system[6], known as data dictionary/directory system(DD/DS), is being used today to control and permit access to multiple user data base supported information systems. Not only is this software system very useful to the data base administrator, but it is also useful to the auditor who is concerned with the integrity and completeness of the data stored in the data base. The DD/DS has three functions which are available as an aid to the auditor.

The first of these functions is to provide documentation for the processing system. In most traditional processing systems, the documentation is missing or is not adequate for the needs of the auditor. In either case. valuable time must be spent in preparing the information about the system. Using a DD/DS, documentation becomes an active and integral part of the data base environment. The documentation, as provided by the system, is available through the automation of metadata.

---

6) Plagman, B. K., "Control and Auditing Techniques Using a DD/DS", *EDP AUDITING,* Auerbach Publishers Inc., June/July 1982.

Using a DD/DS, the metadata is stored in a centralized repository that is automatically updated.

The production of an audit trail at the type level is another feature of DD/DS that benefits the auditor. Using the type level information that is produced, an auditor is able to examine the impact of transactions to the system. This information is crucial to the auditor as he performs substantive tests. From these tests, the DD/DS produces information about how the transactions interact with the system and allows the auditor to properly evaluate the system. The DD/DS is also invaluable in that it provides a means of learning the impact incorrect updates have on the data base.

The third way in which DD/DS provides assistance to the auditor is in the area of control. Since the DD/DS generates all of the metadata that is to be used by the programs, it is necessary to address the DD/DS before compiling the program. With this restriction, it is possible to achieve control by changing data descriptions. If a change has been made, it will be necessary to enter them through the DD/DS, before the application programs can operate. This restricts access to the programs and promotes control.

With these three useful features, the DD/DS provides the auditor with much information that can be used in his audit. However, the DD/DS does not help the auditor evaluate the system for internal controls, but only provides insight into the system and access to some useful information.

The systems, as presented in this review, provide valuable tools to the auditor. They can assist the auditor in his evaluation of an organization's accounting activities. However, each of these systems perform only a part of the audit process. The system proposed in this paper attempted to combine features of these systems.

The proposed system is an integration of a repository of internal controls that should be observed in the sequence of activities. In order to examine the purchasing, it was necessary to divide the purchasing cycle into activity blocks.

By establishing the necessary controls for each block, an auditor can request information that should be present if a particular activity is being performed. With this information at hand, the auditor can then perform compliance tests within the organization. It is also desirable to examine the sequence of activities taking place. The knowledge base has stored a sequence of activities that should be observed. The auditor can then test to see if the sequence of activities, as being performed by the organization, follow the defined sequence. Upon the completion of these tests, the auditor can then make an evaluation and judgement of the internal control structure.

2. COMPONENTS OF INTERNAL CONTROL KNOWLEDGE BASE

The internal control knowledge base (ICKB) is viewed as a repository of knowledge that can be extracted from an experienced auditor. The components of the knowledge base are captured for various processes which are carried out as partial accopmlishments for the organizational goal. A process initiated by a certain department is performed within the department alone or can be spanned across other departments depending upon the nature of activities.

For example, in a purchasing accounting cycle, purchase requisitions are initiated by a requesting department. The purchasing department receives the purchase requisition and performs purchasing function by preparing a purchase order and disseminates the forms to the vendor, receiving department, accounting office and originating department for acknowledgment. The receiving department handles the goods received from the vendor, distributes the goods, and disseminates the receiving report to related departments. Accounting office is involved in comparing documents and preparing the payment voucher.

The process goes through each workstation of the cycle where specific activities are performed according to the internal controls. Data abstraction is obtained in terms of contents carried on the forms which are routed from one workstation to another.[7]

2.I WORKSTATION

A workstation is an organizational unit where forms are received, manipulated and disseminated. Each station has a set of intrays where received forms are placed. An individual intray is provided for each type of form received. Similarly, a distinct outtray is provided for each type of form that is to be routed. Each workstation has a distinct set of trays which can be distinguishable from those of other workstations.

2.2 FORMS FLOW

A form is a vehicle to carry data values from one place to another. Each form contains a set of attributes for instantiated data values. If an organization is going to use forms in its day-to-day processing, it is most advantageous for the work flow to be organized. For if the activities of an

---

7) Sen, A. and Kerschberg, L., "Enterpries Modeling for Database Specification and Design", University of South Carolina, College of Business, Columbia, S. C., pp. 20-28.

organization follow a fixed and definite routine, "most of them can be handled with forms."[8] To be most beneficial, work activities should be arranged in a logical sequence to reduce the "unnecessary duplication of handling, writing, and backtracking, in connection with the forms."[9]

2.3 DATA

Data abstractions are obtained by identifying attributes and instances of a form. Each attribute has a specific name, domain, and range. The aggregation of attributes are represented in a table format for a specific form from which data is abstracted.

2.4 ACTIVITIES

Activities are directed toward achieving the organizational goals and are composed of the performance of each function taking place at a workstation. These functions are performed with the intention of accomplishing an assigned task that is regulated by the controls expressed by management. Activities result in a change in the snapshot of an organizational picture and in the generation of forms.[10]

2.5 CONTROLS

Controls are used to promote the reduction of exposures. This reduction is accomplished by "exercising a restraining or directive influence over the activities of an object, organism, or system."[11]

Internal controls are a necessary part of every organization. They are used to protect organizational assets and to promote the processing of data correctly. The presence of controls encourages and influences people as they perform their duties. Controls provide a means of detecting errors or misrepresentations before they are allowed to contaminate a system while at the same time providing a means of protecting a firm's assets.[12]

Control aspects are imbedded, in a sequence of activities, to define the scope and procedures of each activity. This is done to comply with organizational regulations and auditing rules.

---

8) Kane, Jr. R. L., *CPA Handbook,* Volume 2, American Institute of Accountants, New York, N. Y., 1956.
9) Ibid.
10) Plagman, B. K., op. cit.
11) Kohler, E. L., *A Dictionary for Accountants,* Prentice Hall, Inc., Englewood Cliffs, N. J., 1975.
12) Cushing, B. E., *Accounting Information Systems and Business Organizations,* Addison Wesley Publishing Co., 1982.

3. METHOD OF DESIGN

In developing the repository of controls for the procurement process it was necessary to define the categories of controls that should be observed.

The following control principles and objectives have been chosen for that purpose.

3.1 CONTROL OF DOCUMENTS

Control of documents is the limiting of access to documents used in the processing of a transaction. By restricting and controlling the use of documents, the level of accountability is increased. One way to enhance the accountability of documents is to have the documents prenumbered. This allows one to maintain a log of all documents used. By accounting for all documents, including those spoiled and void, the level of risk is reduced. It is also important to limit access to those documents which initiate transactions affecting organizational assets. Only authorized personnel should have access to these documents and the usage of these documents should be well documented, thus providing a detailed audit trail.[13]

3.2 VERIFICATION

Verification is the review or check, by one or more persons, on work performed by others. This review may be done by someone within the organization or by someone outside of the firm. An example of verification being performed within a firm would be the review of documents supporting the issuance of a check for a payment voucher. This verification should be done to insure the accuracy of all data items in support of the activity. The review of checks received, to verify that remittance is being performed properly, would be an example of verification being performed outside of the firm.[14]

3.3 AUTHORIZATION

Authorization, the assurance that transactions are properly authorized by employees acting

---

13) Nash, J. F. and Roberts, M. B., *Accounting Information Systems,* Macmillan Publishing Co., New York, 1984, Chapter 19.
14) Bailey, Jr. A. D., Gerlach, J., Mcafee, R. P. and Whinston, A. B., "Internal Accounting Controls in the Office of the Future", *IEEE,* May 1981, pp. 59-69.

within the scope of the authority delegated to them, may be either general or specific. A general authorization, set forth by management, is a standard condition followed when processing a transaction. For example, when a purchase requisition is received by the purchasing department from a user department, a purchase order is either prepared, or the requisition is returned to its originator. The action chosen is dependent upon a certain set of criteria established by management. A specific authorization pertains to one specific instance. For example, a purchase order for a new computer system requires the signature of the V.P. of Finance since the amount is over a predefined limit.

Authorizations affect the assets of an organization. It is important that authority be delegated to the proper members of the organization and that activities resulting from authorizations have a way of being checked and verified. Authorization may also be the result of the completion of documents if successive events follow. Thus, a check is prepared to a pay a vendor after a payment voucher has been completed and verified. Again, it is necessary that these "automatic" authorizations have a way of being checked in order to protect organizational assets.[15]

3.4 SEPARATION OF DUTIES

Separation of duties, also known as organizational independence, is a clear logical division of duties and responsibilities. In order to achieve a proper level of independence, it is necessary to distinguish and separate the functions within an organization. There are three general categories in which these functions may be divided. One, the custodial function, involves the custody of assets, such as maintaining the stockroom and releasing inventory for use within the organization. Another function involves the recording functions, such as the preparation of receiving reports or purchase orders. And the third function deals with the operating functions within the organization. An example of an operating function would be the authorization and exeution of transactions.

Following the principle of separating the duties within an organization, a single employee would not be allowed to process a transaction from its origin to its destination. By utilizing organizational independence, an employee could not commit a fraudulent act and also falsify the records to conceal his activities. This is achieved by not allowing an employee to have access to assets, nor to the records of those assets.[16]

---

15) Cushing, B. E., op. cit.
   Nash, J. F. and Roberts, M. B., op. cit.
16) Cushing, B. E., op. cit.
   Mair, W. C., Wood, D. R., and Davis K. W., op. cit.
   Nash, J. F. and Roberts, M. B., op. cit.

4. IMPLEMENTATION

This system was implemented using Waterloo PROLOG under a VM/CMS environment. As indicated previously, this system deals with the purchasing cycle typical in many medium-to-large organizations. However, with minor modifications it can be used for other business applications. Basically, this system can be looked upon as a tool that can be used to assist the auditor in his evaluation of the purchasing cycle in a specific organization. This can be done to ensure that the sequence of activities and internal controls conform to the organizational and general auditing objectives.

To be able to do this, the overall system has to have two major components. The first component consists of the internal representation (knowledge base) describing the standard sequence of activities and documents and the proper controls associated with these activities and documents. The second major component consists of the software (written in PROLOG) that can be used with the internal representation to provide the auditor with answers and/or reports concerning the purchasing cycle under investigation. Specifically, this system will take as an input, the activities and documents in a specific purchasing system and compare them with the knowledge base; provide feedback in terms of whether the sequence of blocks in the system conform to that in the knowledge base.

4.1 THE INTERNAL REPRESENTATION

Basically, there are three categories of knowledge that an auditor would like to use in evaluating a certain business application. The first category is the type of activities and documents that constitute the business cycle. Additionally, the sequence of these activities and documents is of a great importance and is the second category. The final category is the availability of the internal controls associated with each activity and the documents used in the cycle. Consequently, the internal representation part of the system must capture these types of information and store them in such a way that they can be referenced and used by the auditor.

The first two types of knowledge are usually represented using flowcharts. Therefore the first step in designing the system was to develop a flowchart that can be used as a standard for the purchasing function (appendix figure 1). The next step was to transform this flowchart into an internal representation that could be utilized in a computerized evaluation session using PROLOG as a representation tool. Each activity and each document in the purchasing cycle has been represented by an axion. All of these axioms have the same predicate (i.e., "seq(start)", "seq(appr)",

etc., where "seq" is the predicate name and "appr" is an abbreviation for the activity of preparing requisition). Representing the blocks in the flowchart by axioms having the same predicate will maintain the sequence of the activities (i.e., when the predicate "seq" is used to retrieve the block's contents, PROLOG will always start with the first occurrence of an axiom having the predicate "seq" and then moves to the next one in the same order as the axioms have been stored initially). Consequently, one file which has been named PCYCLE in the appendix will provide the auditor with the first two types of knowledge (i.e., types and sequence of blocks). The block names in this file are abbreviated with the beginning character being either an "a" to identify the block as an activity or a "d" to identify the block as a document. The third category of knowledge (i.e., internal controls) has been stored in a series of files. Each file contains the control descriptions for a unique block and has a unique file name corresponding to the block. The files representing the document blocks contain lists and narratives describing the data carried on these documents and the appropriate controls.

4.2 SOFTWARE AND QUERIES SUPPORT MODULES

The second major component of the overall system is the software modules that will use the internal representation to evaluate the auditor's input (i.e., flowchart) and provide answers to queries pertaining to the purchasing cycle.

There are three PROLOG modules that will be used separately but in a certain logical order.

The first step in the process (i.e., a typical session) is to represent the auditor's flowchart in a form acceptable to PROLOG (e.g., each block is transformed into an axiom written in lower case letters and ending with a period). Once the input is prepared, the first software module, named SEEIF, will be used to test the accuracy or relevancy of the input. In other words, SEEIF will check to see if the blocks in the input have the same standard names used in the knowledge base. Basically, SEEIF will try to match each block in the input with one of the axioms in the knowledge base, and will give a message indicating whether the search has been successful or not.

Once this is done, the next step is to check whether the sequence of blocks in the input conforms to that in the knowledge base. This is the purpose of the second software module which is named SEQUEN. Any block that is out of sequence is flagged and the blocks that has to precede it are displayed on the screen. If this occurs, the auditor will have to reorder the blocks in his input and proceed to the next step.

The final step in the process is the retrieval of control specifications for each block of input.

The system provides two alternatives for accomplishing this, and both alternatives use the third software module. This module is named PROJECT.

The first alternative, which is illustrated in the appendix (figure 2), is to provide the control descriptions for all of the blocks in the input in the same sequence that they appear. The other alternative is interactive in the sense that it prompts the auditor to enter a block at a time and in any sequence that he desires. Each time the auditor enters the name of a block, the controls associated with that block are displayed on the screen. Considerable amounts of time, that would otherwise be spent in traversing the flowchart, can be saved using this alternative. This is especially true when the auditor is selective in choosing the blocks which contain the controls he would like to see. When using this alternative and when the auditor enters a block name that does not exist in the knowledge base, the Waterloo PROLOG implementation will automatically return a massage indicating that the name entered does not exist.

A sample session in PROLOG outlining the above steps and their results is included in the appendix along with the software modules and files used in the system. Although the order of the above steps seems to be the logical one for auditing a business cycle, the software modules are independent of each other. Consequently, any of the steps or modules can be performed or executed independently.

5. __SUMMARY__

Designing an expert system for auditing the behavior of any business application is not a trivial task. In order for it to be effective and complete, the system has to incorporate the organizational and auditing objectives in addition to a detailed knowledge or description of the specific business application. Several research efforts have been conducted to design and implement systems that can assist the auditor in his evaluation processes. None of these efforts, however, have come up with the necessary complete system. For instance, some of these research efforts have concentrated on the sequence (or precedence) of activities while others were concerned with the internal controls associated with the activities.

This paper proposes a design that combines knowledge pertaining to the nature of activities and documents in a typical purchasing cycle, the sequence (or precedence) of these activities and documents, and the controls that should be imposed on them. This design was then implemented using PROLOG, thereby proving its applicability. Although the system proposed was concerned with the purchasing application, it can, with minor modifications, be used for other business applications as well and this can be the subject of further research in the area.

BIBLIOGRAPHY

1 . Bailey, Jr., A. D., Gerlach, J., MCA fee, R.P., Whinston, A. B., "Internal Accounting controls in the Office of the Future," *IEEE*, May 1981, pp. 59-69.

2 . Cushing, B. E., *Accounting Information Systems and Business Orqanizations,*Addison-Wesley Publishing Company, 1982.

3 . Kane, Jr., R. L., *CPA Handbook, Volume 2,* American Institute of Accountants, New York, N. Y., 1956.

4 . Kohler, E. L., *A Dictionary for Accountants,* Prentice-Hall, Inc., Englewood Cliffs, N. J., 1975.

5 . Lieberman, A. Z., "Mechanized Analysis of Entity-Relationship Design Databases," *Entity-Relationship Approach to Systems Analysis and Design,* North Holland publishing Company, 1980.

6 . Mair, W.C., Wood, D. R., and Davis, K. W., *Computer Control and Audit,* Q. E. D. Information Sciences, The Springs, Florida, 1978.

7 . Nash, J. F. and Roberts, M. B., *Accounting Information Systems,* Macmillan Publishing Company, New York, 1984, chapter 19.

8 . Perry, W., "Creating and Using an Audit Data Base," *EDP Auditing,* Auerbach Publishers Inc., June/July 1982.

9 . Plagman, B. K., "Control and Auditing Techniques Using a DD/DS," *EDP Auditing,* Auerbach Publishers Inc., June/July 1982.

10. Sen, A., and Kerschberg, L., "Enterprise Modeling for Database Specification and Design," Unversity of South Carolina, College of Business, Columbia, S. C., pp. 20-28.
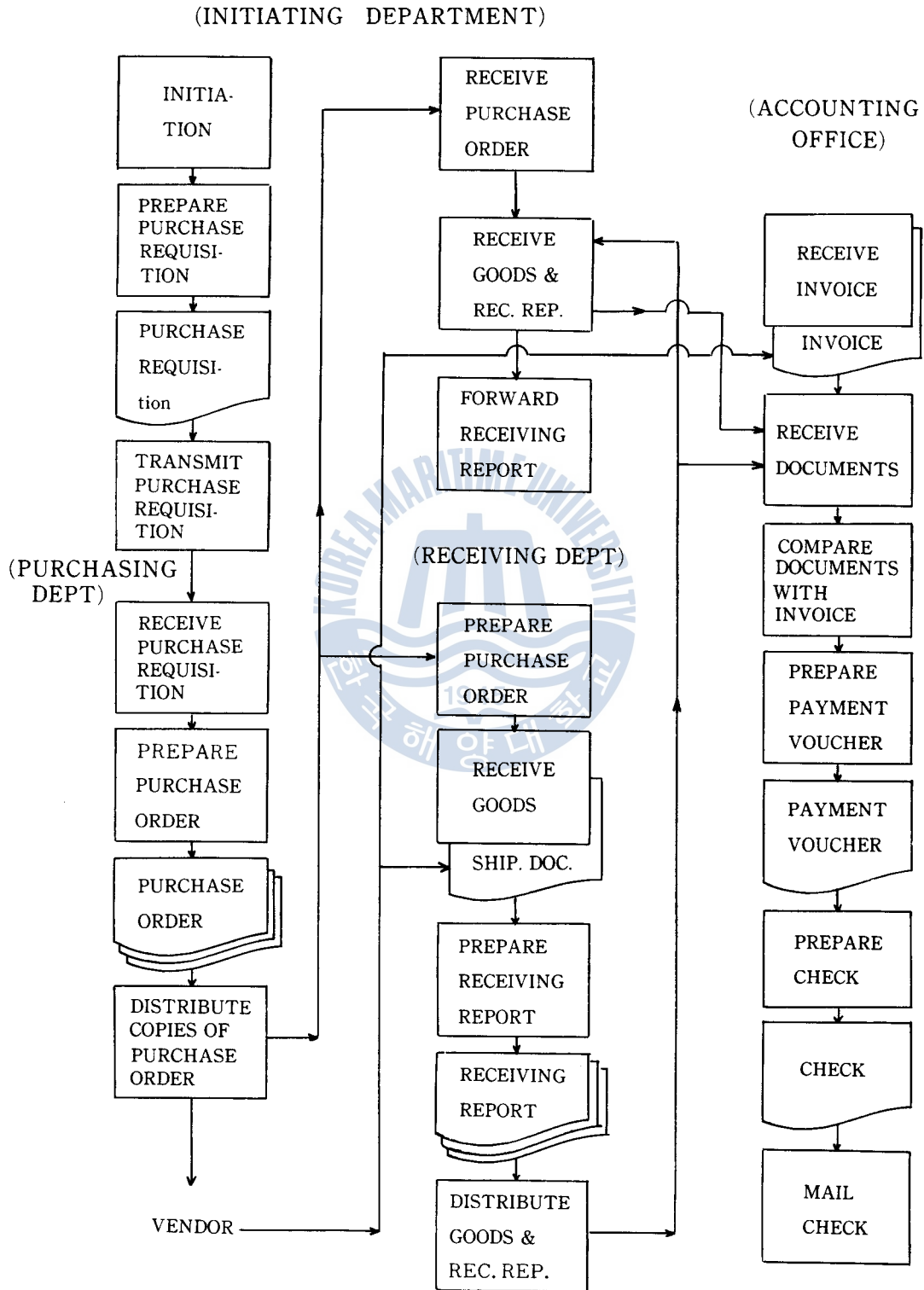
(INITIATING DEPARTMENT)

figure 1. System Flow Chart of the Purchasing Cycle

The Purchasing Cycle in Block Diagram Form

BLOCK	BLOCK NAME	BLOCK CONTENT
1	start	Initialization
2	appr	Prepare Purchase Requisition
3	dpr	Purchase Requisition (Document)
4	atpr	Transmit Purchase Requisition
5	arpr	Receive Purchase Requisition
6	appo	Prepare Purchase Order
7	dpo	Purchasae Order (Document)
8	adpo	Distribute copies of P.O.
9	arpo	Receive Purchase Order (Receiving Dept.)
10	arpor	Receive Purchase order (Requesting Dept.)
11	arg	Receive goods from vendor
12	aprr	Prepare Receiving Report
13	drr	Receiving Report (Document)
14	adg	Distribute Goods & Receiving Report
15	argr	Receive Goods & Receiving Report
16	afrr	forward Receiving Report
17	ari	Receive Invoice
18	ardap	Receive Documents (A/P Dept.)
19	amdi	Match Documents and Invoice
20	appv	Prepare Payment Voucher
21	dapv	Payment Voucher (Document)
22	apc	Prepare Check
23	dapc	Check (Document)
24	amc	Mail Check

NOTE : beginning character "a" stands for "Activity" while "d" stands for "Document".

The Purchasing Cycle In the Knowledge Base (PCYCLE)

seq(start).	seq(dpr).
seq(appr).	seq(atpr).

seq(adg).   seq(arpr).

seq(argr).   seq(appo).

seq(afrr).   seq(dpo).

seq(ari).   seq(adpo).

seq(ardap).   seq(arpo).

seq(amdi).   seq(arpor).

seq(appv).   seq(arg).

seq(dapv).   seq(aprr).

seq(apc).   seq(drr).

FLOWCHART REPRESENTATION		INTERNAL CONTROL KNOWLEDGE BASE

INTEGRATION MODULE

INTEGRATED FLOWCHART

-Data Deseriptions

-Control Features

-Process Flow

-Implied Forms Flow
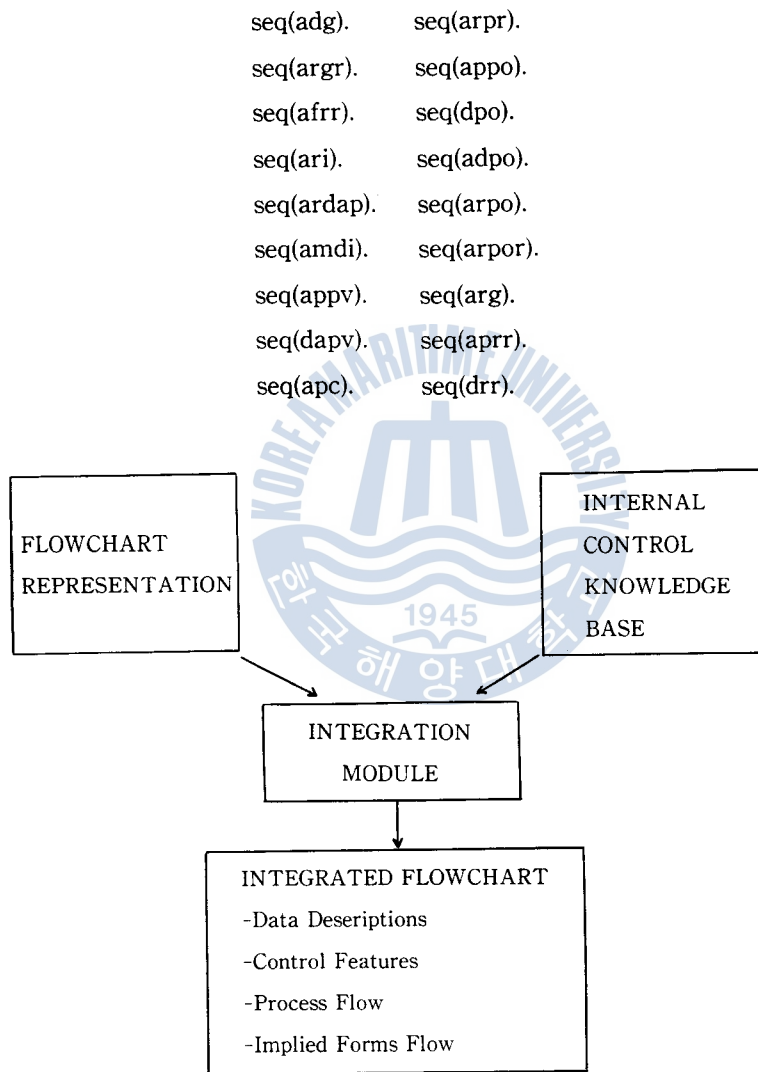
Figure 2. Processes involved in the Program : PROJECT

PROGRAM LISTING

FILE NAME : SEEIF PROLOG

∗ ∗ The program is applied to list the flowchart entered by the auditor and check if the blocks are relevant to this purchasing cycle and accurrately spelled. ∗ ∗

∗ 〈−write('ENTER YOUR FLOWCHART').

∗ exist〈−1 (repeat(X, 1, 30) ε read(XX, flowch, X) ε tran(XX) ε fail, fileclose(flowch)).

∗ tran(XX) 〈−1 (tran1 (XX, KK), chec(XX, KK) ε delaxall(aaa)).

∗ tran1(XX, KK) 〈−repeat(Q, 1, 24) ε ax(seq(Z), seq(ZZ), Q) ε | (equal(XX, ZZ) ε addax(aaa(3)) ε addax(aaa(5))ε
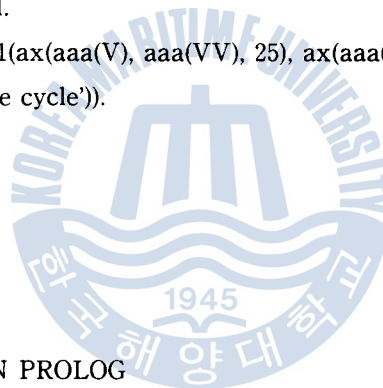
∗ write('the block '.XX.' does exist in our knowlege base')ε fail, neq(XX, ZZ) ε

∗ addax(aaa(Q)))ε fail.

∗ chec(XX,KK)   〈−1(ax(aaa(V), aaa(VV), 25), ax(aaa(V), aaa(VV), 24) ε write('∗ ∗', XX',

∗ does not exist in the cycle')).

∗ equal(X,X).

∗ neq(X,Y).

FILE NAME : SEQUEN PROLOG

∗ ∗ This program is written to check the sequence of the flowchart entered by the auditor according to the purchasing cycle stored in our knowlege base. ∗ ∗

∗ 〈−write('THIS IS YOUR FLOWCHART').

∗ 〈−readch(Y, flowch) ε writech(Y) ε fail.

∗ audit〈−repeat(X, 1,30) ε read(Y, flowch, X)

∗ ε ax(seq(Z), seq(XX), X)

∗ ε | (eq(Y, XX) ε write(X. Y. 'is in the correct order')

∗ ε fail, ne(Y, XX)ε

∗ write('∗ ∗ WARNING ∗ ∗ Before you have ', Y. ' you must enter the following')

∗ ε repeat(N, 1,24) ε ax(seq(W), seq(NN), N)

∗ ε eq(Y, NN) ε diff(N, 1, KLM)

∗ ε repeat(M, 1, KLM) ε ax(seq(K), seq(MM), M)

∗ ε write('   ', M. MM) ε failε fail) ε fail.

FILE NAME : PROJECT PROLOG

* * This program is used to retrieve the internal controls using either alternative 1 which will list the controls for all the blocks in the input (when action (flowchart) is entered) or alternative 2 which will list the controls for the specific block that is entered interactively (when action (screen) is used). * *

    * action(screen) ⟨−newline ε writech('ENTER BLOCK : ')

    * ε newline ε read(Block) ε consult(Block).

    * action(flowchart) ⟨−repeat(X, 1,5) ε read(Block, flowch)

    * ε newline ε newline ε consult(Block)ε fail.

    * action(flowf) ⟨−repeat(X,1,5) ε read(Block, flowf)ε consult(Block)ε fail.

The following is a listing of a PROLOG module that will be called from the program PROJECT to display the internal controls associated with the blocks in the auditor's input (i.e., flowchart).

〈－writech('ACTIVITY IS START') & newline & newline.

〈－writech('There are no controls since this is only a') & writech(' beginning point to be used as reference.') & newline.

〈－writech('ACTIVITY IS PREPARE PURCHASE REQUISITION') & newline & newline.

〈－writech('The following controls should be present:') & newline & newline.

〈－tab(3) & writech('－the purchasing requisition should be') & newline & tab(3) & writech(' prepared only by an authorized clerk in the') & newline & tab(3) & writech(' requesting department.') & newline.

〈－tab(3) & writech('－the purchase requisition should be authorized by') & newline & tab(3) & writech(' the manager of the requesting department.') & newline.

〈－tab(3) & writech('－the purchase requisition number should be the') & writech(' next in the pre-printed sequence.') & newline.

newline & newline & newline.

〈－writech('THE PURCHASE REQUISITION SHOULD CONTAIN:') & newline & newline.

〈－tab(3) & writech('－form number and description') & newline.

〈－tab(3) & writech('－user department name and number') & newline.

〈－tab(3) & writech('－purchase requisition number') & newline.

〈－tab(3) & writech('－authorization') & newline.

〈－tab(3) & writech('－date of issue') & newline.

〈－tab(3) & writech('－required delivery date') & newline & newline.

〈－writech('Each line item on the purchase requisition should') & writech(' contain') & newline & newline.

〈－tab(3) & writech('－internal item code or number') & newline.

〈－tab(3) & writech('－quantity requested') & newline.

〈－tab(3) & writech('－unit of measure') & newline.

〈－tab(3) & writech('－line-item description') & newline.

〈－writech('ACTIVITY IS TRANSMIT PURCHASE PEQUISITION') & newline & newline.

〈－writech('The following controls should be present:') & newline & newline.

〈─writech('─the properly prepared purchase requisition should be') & writech(' sent to the purchasing department via a reliable carrier.') & newline.

〈─writech('─a log giving the date and purchase requisition number') & writech(' should be maintained for each requisition released.') & newline.

〈─writech('ACTIVITY IS RECEIVE PURCHASE REQUISITION') & newline & newline.

〈─writech('The following controls should be present:') & newline.

〈─tab(3) & writech('─authorized clerk in purchasing should receive the') & newline & tab(3) & writech(' purchase requisition.') & newline.

〈─tab(3) & writech('─purchasing clerk should review') & writech(' the purchase requisition.') & newline.

〈─newline & newline.

〈─writech('ACTIVITY IS PREPARE PURCHASE ORDER.') & newline & newline.

〈─writech('The following controls should be present:') & newline & newline.

〈─tab(3) & writech('─the purchase order should be prepared only by─')

〈─tab(3) & writech('─the completed purchase order should be reviewed') & newline.

〈─tab(3) & writech('─and approved by the proper buyer.') & newline.

〈─tab(3) & writech('─the purchase orders should be used in the proper') & newline.

〈─tab(3) & writech(' sequence.') & newline.

〈─tab(3) & writech('─a copy of the purchase order with the purchase') & newline &

〈─tab(3) & writech(' requisition') &

〈─tab(3) & writech(' attached, should be filed in the purchasing') &

〈─tab(3) & writech(' department for reference.') & newline.

〈─tab(3) & writech('─access to unissued purchase orders should be limited') & newline & writech(' to only authorized persons.') & newline.

newline & newline & newline.

〈—writech('—the properly prepared purchase requisition should be') & writech(' sent to the purchasing department via a reliable carrier.') & newline.

〈—writech('—a log giving the date and purchase requisition number') & writech(' should be maintained for each requisition released.') & newline.

〈—writech('ACTIVITY IS RECEIVE PURCHASE REQUISITION') & newline & newline.

〈—writech('The following controls should be present:') & newline.

〈—tab(3) & writech('—authorized clerk in purchasing should receive the') & newline & tab(3) & writech(' purchase requisition.') & newline.

〈—tab(3) & writech('—purchasing clerk should review') & writech(' the purchase requisition.') & newline.

〈—newline & newline.

〈—writech('ACTIVITY IS PREPARE PURCHASE ORDER.') & newline & newline.

〈—writech('The following controls should be present:') & newline & newline.

〈—tab(3) & writech('—the purchase order should be prepared only by—')

〈—tab(3) & writech('—the completed purchase order should be reviewed') & newline.

〈—tab(3) & writech('—and approved by the proper buyer.') & newline.

〈—tab(3) & writech('—the purchase orders should be used in the proper') & newline.

〈—tab(3) & writech(' sequence.') & newline.

〈—tab(3) & writech('—a copy of the purchase order with the purchase') & newline &

〈—tab(3) & writech(' requisition') &

〈—tab(3) & writech(' attached, should be filed in the purchasing') &

〈—tab(3) & writech(' department for reference.') & newline.

〈—tab(3) & writech('—access to unissued purchase orders should be limited') & newline & writech(' to only authorized persons.') & newline.

newline & newline & newline.

〈—writech('THE PURCHASE ORDER SHOULD CONTAIN:') & newline & newline.

〈—tab(3) & writech('—form number and description') & newline.

〈—tab(3) & writech('—name and address of initiating organization') & newline.

〈—tab(3) & writech('—name and address of vendor or supplier') & newline.

〈—tab(3) & writech('—vendor number') & newline.

〈—tab(3) & writech('—purchase order number') & newline.

〈−tab(3) & writech('−purchase requisition number') & newline.

〈−tab(3) & writech('−authorization') & newline.

〈−tab(3) & writech('−date of issue') & newline.

〈−tab(3) & writech('−required delivery schedule') & newline.

〈−tab(3) & writech('−terms and conditions') & newline.

〈−tab(3) & writech('−shipping instructions') & newline.

〈−tab(3) & writech('−insurance required') & newline.

〈−tab(3) & writech('−F.O.B. point') & newline.

〈−tab(3) & writech('−responsibility for damage') & newline.

〈−tab(3) & writech('−billing instructions') & newline & newline.

〈−tab(3) & writech('Each line item on the purchase order should contain:') & newline & newline & newline.

〈−tab(3) & writech('−item code or number') & newline.

〈−tab(3) & writech('−description') & newline.

〈−tab(3) & writech('−quantity ordered') & newline.

〈−tab(3) & writech('−unit of measure') & newline.

〈−tab(3) & writech('−unit price') & newline.

〈−tab(3) & writech('−extended price') & newline.

〈−tab(3) & writech('−line-item description') & newline.

〈−tab(3) & writech('−net extended price') & newline & newline.

〈−writech('Purchase order froms should be prenumbered for') & writech(' control purposes.') & newline.

〈−writech('Purchase orders are normally prepared') & writech(' with multiple copies').

〈−writech(' for distribution to the vendor, to the') & writech(' Receiving Department,') & newline.

〈−writech(' to the warehouse or stockroom, and to Accounting.') & writech('One copy').

〈−writech(' is retained for the Purchasing Department's records.').

〈−writech('The vendor may be sent two copies, a retention copy') & writech(' and an').

〈−writech(' acknowledgement copy to be returned to the Purchasing').

〈−writech(' Department to indicate the acceptance of the order.') & newline.

〈−writech('ACTIVITY IS DISTRIBUTE COPIES OF PURCHASE ORDER') & newline & newline & newline.

〈−tab(3) & writech('−The following controls should be present:') & newline.

〈−tab(3) & writech('−copies should be distributed by') & writech(' reliable carrier') & newline.

〈─tab(3) & writech('─copy of purchase order should be sent to vendor for') & newline & tab(4) & writech(' authourization to ship.') & newline.

〈─tab(3) & writech('─copy of purchase order should be sent to the') & newline & tab(4) & writech (' ordering department for notification that order') & newline & tab(3) & writech(' has been made.') & newline.

〈─tab(3) & writech('─copy of purchase order should be sent to the') & newline & writech(' receiving department for authorization to accept the') & newline & tab(3) & writech(' goods when they arrive.) & newline.

〈─writech('ACTIVITY IS RECEIVE PURCHASE ORDER IN RECEIVING') & newline & newline.

〈─writech('The following controls should be present:') & newline & newline.

〈─tab(3) & writech('─the purchase order should be received') & newline & tab(4) & writech (' by authorized clerk.') & newline.

〈─tab(3) & writech('─the purchase order should be filed by authorized') & newline & tab(4) & writech(' clerk in area of limited access.') & newline.

〈─writech('ACTIVITY IS RECEIVE PURCHASE ORDER IN REQUESTING DEPT') & newline & newline.

〈─writech('The following controls should be present:') & newline & newline.

〈─tab(3) & writech('─the copy of the purchase order should be received') & writech(' only by an authorized clerk.') & newline.

〈─tab(3) & writech('─the copy of the purchase order should be reviewed') & newline (' to insure that the goods ordered where as requested.') & newline.

〈─tab(3) & writech('─the clerk who compares the purchase order and') & writech(' requisition should not be the same person.') & newline.

〈─tab(3) & writech('─the purchase order and purchase requisition should') & writech(' filed in an area of limited access.') & newline.

〈─writech('ACTIVITY IS RECEIVE GOODS FROM VENDOR') & newline & newline.

〈─writech('The following controls should be present:') & newline & newline.

〈─tab(3) & writech('─goods should be received in the receiving') & newline & tab(3) & writech(' department during normal working hours.') & newline.

〈─tab(3) & writech('─authorized clerk should sign for goods') & newline & tab(4) & writech(' when they are received.') & newline.

〈─tab(3) & writech('─purchase order should be matched with the goods.') & newline.

〈−writech('ACTIVITY IS PREPARE RECEIVING REPORT') & newline & newline.

〈−writech('The following controls should be present:') & newline & newline.

〈−tab(3) & writech('−goods should be examined and compared with the') & newline & tab(4) & writech(' purchase order to insure that their identity,') & newline & tab(4) & writech(' and quality conform to the expectations') & tab(1) & writech (' of the purchase order.') & newline.

〈−tab(3) & writech('−a receiving report should be prepared by an') & newline & tab(4) & writech(' authorized clerk detailing this information.') & newline.

〈−tab(3) & writech('−if the goods are not in conformity with applicable') & newline & writech(' specifications or were not ordered, they should) & newline & tab(4) & writech(' be returned to the vendor.) & newline.

〈−tab(3) & writech('−a log should be maintained datailing prepared') & newline & tab(3) & writech(' receiving reports.') & newline.

〈−tab(3) & writech('−receiving reports should be prepared in the proper) & newline & tab(4) & writech(' numerical sequence.') & newline.

〈−writech('DOCUMENT IS RECEIVING REPORT') & newline & newline.

〈−writech('Document should contain:') & newline.

〈−tab(3) & writech('−form number and description') & newline.

〈−tab(3) & writech('−name and address of vendor or supplier') & newline.

〈−tab(3) & writech('−vendor number') & newline.

〈−tab(3) & writech('−date received') & newline.

〈−tab(3) & writech('−shipped via') & newline.

〈−tab(3) & writech('−transportation charges') & newline.

〈−tab(3) & writech('−freight bill number and date') & newline.

〈−tab(3) & writech('−purchase order number') & newline.

〈−tab(3) & writech('−requisition number') & newline.

〈−tab(3) & writech('−department of destination') & newline.

〈−tab(3) & writech('−received by/checked by') & newline.

〈−tab(3) & writech('−accepted/rejected') & newline.

〈−tab(3) & writech('−partial or complete') & newline.

〈−tab(3) & writech('−comments') & newline & newline.

〈−writech('Each line item, on the receiving report should contain.') & newline & newline.

〈−tab(3) & writech('−item number') & newline.

〈−tab(3) & writech('−description') & newline.

〈—tab(3) & writech('—quantity received') & newline.

〈—tab(3) & writech('—unit of measure') & newline.

〈—tab(3) & writech('—condition') & newline.

〈—writech('Receiving reports should be prenumbered.') &

writech('The distribution of the report should') &

writech (' include the Purchasing Department, the warehouse or') &

writech(' stockroom, the originating department, and Accounting. An') &

Writech(' additional copy may be sent to Quality Control, and one copy') &

Writech(' is retained by the Receiving Department.') & newline.

〈—writech('ACTIVITY IS DISTRIBUTE GOODS TO REQUESTING DEPARTMENT') & newline & newline.

〈—writech('The following controls should be present:') & newline & newline.

〈—tab(3) & writech('—a copy of the receiving report should') & newline & tab(3) & writech(' be sent with the goods to') & newline & tab(3) & writech(' the requesting department.') & newline.

〈—tab(3) & writech('—a copy of the receiving report, purchase') & newline & tab(3) & writech(' order and other supporting documents should be') & newline & tab(3) & writech(' sent to the accounts payable dept.') & newline.

〈—writech('ACTIVITY IS RECEIVE GOODS IN REQUESTING DEPARIMENT') & newline & newline.

〈—writech('The following controls should be present:') & newline & newline.

〈—tab(3) & writech('—authorized clerk should receive the goods') & newline.

〈—tab(3) & writech('—clerk should sign receiving report after') & writech(' verifying that goods') & newling & tab(3) & writech(' received are as listed on receiving report.') & newline.

〈—writech('ACTIVITY IS FORWARD RECEIVING REPORT') & newline & newline.

〈—writech('The following control should be present:') & newline & newline.

〈—tab(3) & writech('—receiving report should be sent to') & writech(' accounts payable to') & newline & tab(3) & writech(' verify that goods were received.') & newline.

〈—writech('ACTIVITY IS RECEIVE INVOICE') & newline & newline.

〈—writech('The following controls should be present:') & newline & newline.

〈—tab(3) & writech('—the invoice should be received by') & writech(' authorized clerk in') &

newline & tab(3) & writech(' accounts payable department.') & newline.

〈—tab(3) & writech('—invoice should be opened and "date" ') & writech(' stamped promptly.) & newline.

〈—tab(3) & writech('—extensions should be verified.') & newline.

〈—tab(3) & writech('—invoice should be filed in proper place') & writech(' with limited access.') & newline.

〈—writech('ACTIVITY IS RECEIVE DOCUMENTS IN ACCOUNTS PAYABLE') & newline & newline.

writech('The following controls should be present:') & newline & newline.

〈—tab(3) & writech('—authorized clerk should receive supporting') & newline & tab(4) & writech('documents from various departments.') & newline.

〈—tab(3) & writech('—documents should be reviewed and filed in') & newline & tab(4) & writech('proper plate which has limited access.') & newline.

〈—writech('ACTIVITY IS MATCH DOCUMENTS WITH INVOICE') & newline & newline.

〈—writech('The following controls should be present:') & newline & newline.

〈—tab(3) & writech('—all supporting documents should be matched') & newline & tab(3) & Writech(' by authorized clerk.') & newline.

〈—tab(3) & writech('—verification should be made to insure that') & newline & tab(4) & writech(' goods received, ordered, and invoiced agree.') & newline.

〈—tab(3) & writech('—if they agree, authorized clerk should') & newline & tab(4) & writech(' approve for payment.') & newline.

〈—tab(3) & writech('—cancel all supporting documents.') & newline.

〈—tab(3) & writech('—cancelled documents should be filed in') & newline & tab(4) & writech(' proper place for reference.') & newline.

〈—writech('ACTIVITY IS PREPARE INVOICE') & newline & newline.

〈—writech('The following controls should be present:') & newline & newline.

〈—tab(3) & writech('—payment voucher should be prepared') & newline & tab(4) & writech(' by authorized clerk.') & newline.

〈—tab(3) & writech('—payment voucher should be used in') & newline & tab(4) & writech(' the proper sequence.') & newline.

〈—tab(3) & writech('—a log should be maintained to allow for') & newline &

tab(4) & writech(' accountability of all vouchers.') & newline.

<—tab(3) & writech('—all supporting documents should be reviewed.') & newline.

<—tab(3) & writech('—voucher should be signed by authorized clerk.') & newline.

<—writech('DOCUMENT IS ACCOUNTS PAYABLE VOUCHER') & newline & newline.

<—writech('Each form should include:') & newline & newline.

<—tab(3) & writech('—form number and description') & newline.

<—tab(3) & writech('—name and address of vendor') & newline.

<—tab(3) & writech('—date of issue') & newline.

<—tab(3) & writech('—vendor invoice number') & newline.

<—tab(3) & writech('—invoice date') & newline.

<—tab(3) & writech('—due date') & newline.

<—tab(3) & writech('—terms') & newline.

<—tab(3) & writech('—purchase order number') & newline.

<—tab(3) & writech('—requisition number') & newline.

<—tab(3) & writech('—purchase order and receiving report attached(Y/N)') & newline.

<—tab(3) & writech('—account number') & newline.

<—tab(3) & writech('—gross amount of purchases') & newline.

<—tab(3) & writech('—net amount of purchases') & newline.

<—tab(3) & writech('—freight and handling charges') & newline.

<—tab(3) & writech('—sales tax(if applicable)') & newline.

<—tab(3) & writech('—total amount due') & newline.

<—tab(3) & writech('—discount') & newline.

<—tab(3) & writech('—net amount due') & newline.

<—tab(3) & writech('—authorization for payment') & newline.

<—writech('Accounts payable vouchers should be prenumbered.') &

writech(' Vouchers are') &

writech(' sometimes prepared as a single copy : in other cases, a second') &

writech(' copy is made to support the check sent to the vendor.') & newline.

<—writech('ACTIVITY IS PREPARE CHECK') & newline & newline.

<—writech('The following controls should be present:') & newline & newline.

<—tab(3) & writech('—documents supporting check should be reviewed.') & newline.

<—tab(3) & writech('—check should be prepared by authorized person.') & newline.

〈−tab(3) & writech('−checks should be used in proper sequence.') & newline.

〈−tab(3) & writech('−a log should be maintained to allow for') & newline.

tab(4) & writech(' accountability of checks.') & newline.

〈−tab(3) & writech('−checks should be signed by authorized') & newline & tab(4) & writech(' person or persons.') & newline.

〈−tab(3) & writech('−checks should be stored in area of') & newline & tab(4) & writech(' limited access.') & newline.

〈−writech('DOCUMENT IS ACCOUNTS PAYABLE CHECK') & newline & newline.

writech('The document should contain:') & newline.

〈−tab(3) & writech('−date of check') & newline.

〈−tab(3) & writech('−name of vendor') & newline.

〈−tab(3) & writech('−address of vendor') & newline.

〈−tab(3) & writech('−vendor number') & newline.

〈−tab(3) & writech('−the gross amount') & newline.

〈−tab(3) & writech('−reference to the invoice(s) by number and amount') & newline.

〈−tab(3) & writech('−reference to the payment voucher') & newline.

〈−tab(3) & writech('−bank information') & newline & newline.

〈−writech('Checks should be prenumbered and stored in an area') & writech(' of limited access.').

〈−writech('A log should be maintained for each check used, including') & writech(' those void or spoiled.').

〈−writech('This log should include the date issued,') & writech (' who issued the check, to whom it was issued, the amount of') & writech(' issuance, and a place to show when the check has cancelled.') & newline.

〈−writech('ACTIVITY IS MAIL CHECKS') & newline & newline.

〈−writech('The following controls should be present:') & newline & newline.

〈−tab(3) & writech('−checks should be distributed promptly') & newline & tab(3) & writech(' after being prepared') & newline.

〈−tab(3) & writech('−checks should be distributed by a reliable') & newline   & tab(3) writech(' carrier') & newline.

SAMPLE SESSION

. prolog seeif pcycle

B (200) R/D

Welcome To Waterloo Prolog 1.3

ENTER YOUR FLOWCHART.

consult(SEEIF)〈−

consult(PCYCLE)〈−

consult(flowch).

consult(flowch)〈−

exist.

the block. start. does exist in our knowlege base.

the block. appr. does exist in our knowlege base.

the block. dpr. does exist in our knowlege base.

the block. atpr. does exist in our knowlege base.

the block. appo. does exist in our knowlege base.

the block. dpo. does exist in our knowlege base.

the block. adpo. does exist in our knowlege base.

the block. arpo. does exist in our knowlege base.

the block. afrr. does exist in our knowlege base.

the block. apc. does exist in our knowlege base.

the block. dapc. does exist in our knowlege base.

the block. amc. does exist in our knowlege base.

exist〈−

. stop.

. prolog sequen

C (201) R/O

Welcome To Water 100 Prolog 1.3

THIS IS YOUR FLOWCHART.

start.

appr.

dpr.

atpr.

appo.

dpo.

adpo.

arpo.

afrr.

apc.

dapc.

amc.

consult(SEQUEN)<—

. consult(flowch).

consult(flowch)<—

. consult(pcycle).

consult(pcycle)<—

. audit.

1. start. is in the correct order.

2. appr. is in the correct order.

3. dpr. is in the correct order.

4. atpr. is in the correct order.

＊＊WARNING＊＊ Before you have .appo. you must enter the following.

. 1. start.

. 2. appr.

. 3. dpr.

. 4. atpr.

. 5. arpr.

＊＊WARNING＊＊ Before you have .dpo. you must enter the following.

. 1. start.

. 2. appr.

. 3. dpr.

. 4. atpr.

. 5. arpr.

. 6. appo.

＊＊WARNING＊＊ Before you have .adpo. you must enter the following.

. 1. start.

. 2. appr.

.3. dpr.

.4. atpr.

.5. arpr.

.6. appo.

.7. dpo.

∗∗WARNING∗∗ Before you have .arpo. you must enter the following.

.1. start.

.2. appr.

.3. dpr.

.4. atpr.

.5. arpr.

.6. appo.

.7. dpo.

.8. adpo.

∗∗WARNING∗∗ Before you have .afrr. you must enter the following.

.1 . start.

.2 . appr.

.3 . dpr.

.4 . atpr.

.5 . arpr.

.6 . appo.

.7 . dpo.

.8 . adpo.

.9 . arpo.

.10. arpor.

.11. arg.

.12. aprr.

.13. drr.

.14. adg.

.15. argr.

∗∗WARNING∗∗ Before you have .apc. you must enter the following.

.1 . start.

. 2 . appr.

. 3 . dpr.

. 4 . atpr.

. 5 . arpr.

. 6 . appo.

. 7 . dpo.

. 8 . adpo.

. 9 . arpo.

.10. arpor.

.11. aprr.

.12. drr.

.13. adg.

.14. argr.

.15. afrr.

.16. ari.

.17. ardap.

.18. amdi.

.19. appv.

.20. dapv.

.21. apc.

＊＊WARNING＊＊ Before you have .dapc. you must enter the following.

. 1 . start.

. 2 . appr.

. 3 . dpr.

. 4 . atpr.

. 5 . arpr.

. 6 . appo.

. 7 . dpo.

. 8 . adpo.

. 9 . arpo.

.10. arpor.

.11. aprr.

.12. drr.

.13. adg.

.14. argr.

.15. afrr.

.16. ari.

.17. ardap.

.18. amdi.

.19. appv.

.20. dapv.

.21. dapv.

.22. apc.

WARNING Before you have .amc. you must enter the following.

. 1 . start.

. 2 . appr.

. 3 . dpr.

. 4 . atpr.

. 5 . arpr.

. 6 . appo.

. 7 . dpo.

. 8 . adpo.

. 9 . arpo.

.10. arpor.

.11. aprr.

.12. drr.

.13. adg.

.14. argr.

.15. afrr.

.16. ari.

.17. ardap.

.18. amdi.

.19. appv.

.20. dapv.

.21. apc.

.22. dapc.

. 23. dapc.

?

. stop.

. prolog project

E (202) R/O

Welcome To Waterloo Prolog 1.3

consult (PROJECT)〈ー

. action(flowchart).

ACTIVITY IS START

There are no controls since this is only a beginning point to be used as reference.

ACTIVITY IS RPEPARE PURCHASE REQUISITION

The following controls should be present:
— the purchasing requisition should be

prepared only by an authorized clerk in the requesting department.
— the purchase requisition should be authorized by the manager of the requesting department.
— the purchase requisition number should be the next in the pre-printed sequence.

THE PURCHASE REQUISITION SHOULD CONTAIN:

— form number and description
— user department name and number
— purchase requisition number
— authorization
— date of issue
— required delivery date

Each line item on the purchase requisition should contain

— internal item code or number
— quantity requested
— unit of measure
— line-item description

ACTIVITY IS TRANSMIT PURCHASE REQUISITION

The following controls should be present:

- the properly prepared purchase requisition should be sent to the purthasing department via a reliable carrier.
- a log giving the date and purchase requisition number should be maintained for each requisition released.

ACTIVITY IS PREPARE PURCHASE ORDER.

The following controls should be present:

- the purchase order should be prepared only by an authorized purchasing clerk.
- the completed purchase order should be reviewed and approved by the proper buyer.
- the purchase orders should be used in the proper sequence.
- a copy of the purchase order with the purchase
- access to unissued purchase orders should be limited to only authorized persons.

THE PURCHASE ORDER SHOULD CONTAIN:

- form number and description
- name and address of initiating organization
- name and address of vendor or supplier
- vendor number
- purchase requisition number
- authorization
- date of issue
- required delivery schedule
- terms and conditions
- shipping instructions
- insurance required
- F.O.B. point
- responsibility for damage
- billing instructions

Each line item on the purchase order should contain:

- item code or number

— description
— quantity ordered
— unit of measure
— extended price
— line-item description
— net extended price

Purchase order forms should be prenumbered for control purposes.

Purchase orders are normally prepared with multiple copies for distribution to the vendor, to the Receiving Department, to the warehouse or stockroom, and to Accounting. One copy is retained for the Purchasing Department's records.

The vendor may be sent two copies, a retention copy and an Acknowledgement copy to be returned to the Purchasing Department to indicate the acceptance of the order.

ACTIVITY IS DISTRIBUTE COPISE OF PURCHASE ORDER

The following controls should be present:
— copies should be distributed by reliable carrier
— copy of purchase order should be sent to vendor for authorization to ship.
— copy of purchase order should be sent to the ordering department for notification that order has been made.
— copy of purchase order should be sent to the receiving department for authorization to accept the goods when they arrive.

ACTIVITY IS RECEIVE PURCHASE ORDER IN RECEIVING

The following controls should be present:
— the purchase order should be received by authorized clerk.
— the purchase order should be filed by authorized clerk in area of limited access.

ACTIVITY IS FORWARD RECEIVING REPORT

The following control should be present:
— receiving report should be sent to accounts payable to verify that goods were received.

ACTIVITY IS PREPARE CHECK

The following controls should be present:
- documents supporting check should be reviewed.
- check should be prepared by authorized person.
- checks should be used in proper sequence.
- a log should be maintained to allow for accountability of checks.
- checks should be signed by authorized person or persons.
- checks should be stored in area of limited access.

DOCUMENT IS ACCOUNTS PAYABLE CHECK

- date of check
- name of vendor
- address of vendor
- vendor number
- the gross amount
- reference to the invoice(s) by number and amount
- reference to the payment voucher
- bank information
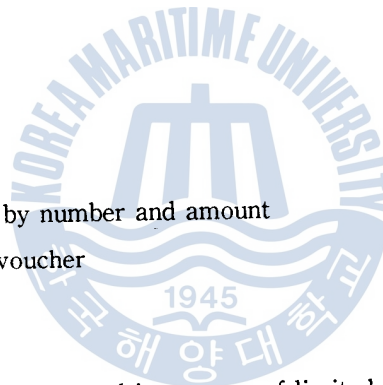
Checks should be prenumbered and stored in an area of limited access.

A log should be maintained for each check used, including those void or spoiled. This log should include the date issued,   who issued the check, to whom it was issued, the amount of issuance, and a place to show when the check has cancelled.

ACTIVITY IS MAIL CHECKS

The following controls should be present:
- checks should be distributed promptly after being prepared
- checks should be distributed by a reliable carrier

?

.action(screen).

ENTER BLOCK:

.ari.

ACTIVITY IS RECEIVE INVOICE

The following controls should be present:

- — the invoice should be received by authorized clerk in accounts payable department.
- — invoice should be opened and "date" stamped promptly.
- — extensions should be verified.
- — invoice should be filed in proper place with limited access.

action(screen)〈—

.action(screen).

ENTER BLOCK:

.soso.

SOSO   DOES NOT EXIST

action(screen)〈—

.action(screen).

ENTER BLOCK:

.arg.

ACTIVITY IS RECEIVE GOODS FROM VENDOR

The following controls should be present:

- — goods should be received in the receiving
- — authorized clerk should sign for goods when they are received.
- — purchase order should be matched with the goods.

action(screen)〈—

.stop.