DOCUMENT RESUME

ED 039 009 LI 001 936

AUTHOR	Black, Donald V.
TITLE	Library Information System Time-Sharing (LISTS) Project. Final Report.
INSTITUTION	System Development Corp., Santa Monica, Calif.
SPONS AGENCY	Council on Library Resources, Inc., Washington, D.C.
REPORT NO	TM-4547
PUB DATE	1 May 70
NOTE	216p.
EDRS PRICE	EDRS Price MF-$1.00 HC-$10.90
DESCRIPTORS	*Automation, Cataloging, *Computer Programs, Documentation, *Electronic Data Processing, Information Processing, Libraries, *Library Acquisition, *Library Technical Processes
IDENTIFIERS	Library Automation, *Library Information System Time Sharing, LISTS, On Line Systems

ABSTRACT

The Library Information System Time-Sharing (LISTS) experiment was based on three innovations in data processing technology: (1) the advent of computer time-sharing on third-generation machines, (2) the development of general-purpose file-management software and (3) the introduction of large, library-oriented data bases. The main body of the report contains: (1) purpose, (2) background and development of LISTS, (3) conduct of the LISTS experiment, (4) problems encountered, (5) results and (6) conclusions. The five appendices cover: (1) on-line circulation simulation, (2) LISTS system users manual, (3) LISTS system users manual for circulation control subsystem at Fullerton Junior College Library, (4) instructions for operating an on-line circulation system and (5) LISTS evaluation questionnaire. Based on this experiment, it appears that automation is an improvement over manual processing in some parts of the acquisitions process, for bibliographic searching of the Machine Readable Cataloging (MARC) records, for circulation control and for production of bookform catalogs. One of the most important results of this experiment was the understanding gained by the participants of what automation can offer for library applications. (NH)

TM-4547

TECHNICAL MEMORANDUM

(TM Series)

LIBRARY INFORMATION SYSTEM TIME-SHARING (LISTS) PROJECT

Final Report

Donald V. Black

May 1, 1970

SYSTEM DEVELOPMENT CORPORATION
2500 COLORADO AVE.
SANTA MONICA
CALIFORNIA
90406

ACKNOWLEDGMENT

The author wishes to acknowledge the help of many who worked on the LISTS project, in particular: former colleagues Jules Mersel, Donald Bethe, Mrs. Margaret Cabaniss, and Mrs. Barbara Blankenship; all of the librarians and other staff in the participating libraries; consultants Earl Farley and Melvin Voigt; colleagues Dr. Carlos Cuadra, Dr. Robert Katter, Donald Blankenship, Robert Burket, and John Luke; Mrs. Lillian Trevisan, whose efficient secretarial support was invaluable; and especially Karl Pearson who, more than any other person, contributed so much to make the experiment a success. The author also wishes to acknowledge the support of the Council on Library Resources and the helpfulness of the project monitor, Mr. Carl Spaulding.

TABLE OF CONTENTS

LIST OF EXHIBITS

SUMMARY

The LISTS experiment was based on three innovations in data processing technology: the advent of computer time-sharing on third-generation machines, the development of general-purpose file-management software, and the introduction of large, library-oriented data bases, especially the Library of Congress' MARC II service. Within that context, the experiment was conducted to determine whether library automation is clearly an improvement over manual processing, and whether the cost of library automation is tolerable, particularly for smaller libraries without access to computer support of their own. A prototype computer system operating on an IBM 360/67 was developed to take advantage of the three innovations, and service was provided to six Southern California libraries to carry out the LISTS experiment.

The LISTS system comprised a complex of time-shared and batch processing computer programs. The heart of the system was a set of programs that worked with the MARC II data. Various subsystems developed during the project supported typical library functions such as acquisitions, cataloging, serials control, circulation control, and certain aspects of reference work. The participating libraries had the use of the computer system from remote terminals during the operational phase of the experiment, a period of eight months.

Based on the experiment, it appears that automation is an improvement over manual processing in some parts of the acquisitions process, for bibliographic searching of the MARC records, for circulation control, and for production of bookform catalogs. Because of the loss of one participating library, it was not possible to assess the value of automation for serials control during this experiment.

LISTS was also able to provide some answers to the question of cost. Some aspects of on-line bibliographic searching, circulation control, and production of a bookform catalog appear to be reasonable in cost.

The experiment indicated the critical need to receive MARC data rapidly, and to have retrospective records included in MARC. The cataloging done at the Library of Congress is of little value until it has been received by those wanting to use it.

The most important result of the experiment, for the participants, was the understanding they gained of what automation is likely to offer for library applications, together with an appreciation of the considerable time and effort that must go into making a major computer-based system useful.

A general-purpose system such as TS/DMS is primarily useful for achieving an operational system in a fairly short period of time, but is not likely to be cost-effective for a specialized operation such as library processing. Large, centrally maintained files of bibliographic data are likely to be more efficient in the long run than separately maintained files, provided that there is a sufficiently large user group to support file correction and updating.

LISTS has given at least some indication that it is more efficient, in the long run, for libraries to join together to develop a system that all can use on a common computer than to develop a separate system for every library.

I. PURPOSE

TECHNOLOGICAL BASIS FOR THE LISTS CONCEPT

At the inception of the LISTS project in 1967, it was felt that three advances in data processing technology had raised the possibility that computer support could be extended to small or medium-sized libraries. These advances were:

1) The introduction of computer time-sharing;

2) The introduction of large, general-purpose file-handling programs capable of manipulating the types of data bases (or files) required for the internal management of library holdings; and

3) The introduction of large, library-oriented data bases or files of bibliographic data, especially the Library of Congress MARC II files.

The first of these, time-sharing, was not really new, particularly at SDC, but the introduction of the so-called "third-generation" computer equipment with special features to support time-sharing use was still new. Time-sharing seemed to be particularly important for library data processing for several reasons. It offered the prospect for relatively inexpensive computer use by spreading fixed expenses over a large number of users. It allowed maximum flexibility and responsiveness to user requirements for computer time whenever it was wanted. It provided as much or as little of the available computer and peripheral equipment resources needed to perform any data processing task; customers could be charged only for the resources actually used, yet have available the full power of a large computer for work (such as sorting) done most efficiently on a large machine. It provided users with a capability for interacting with the computer-based files more flexible than was possible with manual files. More than one person at a time could use the same file, and very powerful file searching strategies--controlled by the user--could be employed.

A single library could provide only a few minutes of work each day for a large computer. It would be exhorbitant, for a library to acquire such a computer for its own use. Time-sharing could make such a computer available to many libraries simultaneously, at a comparatively moderate cost to each. Libraries could communicate with the computer through relatively inexpensive terminal devices that utilize voice-grade telephone lines or the even cheaper lines of a Teletype network.

Before the introduction of general-purpose file management programs for computers, it would have been necessary for libraries to design and write their own special-purpose programs. The availability of general-purpose

programs provided a base upon which to build programs specific to libraries with much less labor and time than would have been required in earlier years. SDC's TS/DMS (Time-Shared Data Management System), capable of handling large files on the IBM 360/67 computer in an on-line, time-sharing mode, was scheduled to be completed at that time. TS/DMS is a comprehensive information processing system designed to serve the needs of a variety of facilities, users, and applications. With TS/DMS, SDC was integrating an array of advanced information processing tools and techniques into a single, multi-purpose system. TS/DMS comprises a time-sharing executive program, data management systems (ORBIT and CDMS), utilities, and a programmers' package. The executive permits a straightforward conversational exchange between the computer and a user, who does not need programming experience. It also has the capability to do off-line batch processing of a utility nature.

SDC has developed its own executive program for the time-sharing system on the 360/67 computer. The executive employs the "time-slicing" principle rather than the core partition principle for time-sharing. Each user receives, in turn, a small quantum of time during which his program has access to the full resources of the computer. Users communicate to the time-sharing executive a set of special command words. These commands include the login and logout requests, to enter or leave the system; the program load and go requests, to begin operation of a user's program; miscellaneous commands to show such things as storage files available, number of users, program operational status, etc.; and a command to send a message to the computer operator or another terminal.

The data management systems allow users to retrieve and manipulate data from large files rapidly in a direct, reasonably uncomplicated way. TS/DMS does not require that the user have sophisticated knowledge of computer technology. With TS/DMS the user can:

- describe large collections of data and input them, using only the real characteristics of the data--that is, name, type, and relationship to other elements--without concern for computer functions;
- modify data elements on-line or off-line;
- retrieve information by simply presenting queries to the computer or by requesting reports of various kinds;
- manage large data files using a reasonably convenient language.

Utility programs and a programmers' package provide the full advantages of on-line program production, debugging, and various types of utility use of the computer without tying up all of the resources of the computer.

Finally, the particular large library-oriented file of bibliographic data that was most important to LISTS was the Library of Congress MAchine Readable Cataloging (MARC) file, at that time still in the initial pilot-project stage. Studies had shown that the cost of keyboarding library catalog information could range from roughly $.30 to more than $1.00 per catalog entry. (The cost of keyboarding most textual information is about one cent per word.) The work of the Library of Congress in creating the MARC system promised to relieve many libraries of the burden of keyboarding catalog information, at least for current English-language monographs. It was hoped that the availability of this bibliographic data base for acquisitions, cataloging, and other purposes would make financially feasible many library data processing operations that could not be undertaken before.

PURPOSE OF THE LISTS EXPERIMENT

The purpose of the LISTS experiment was to determine whether or not on-line access to a large file of bibliographic data (such as represented by MARC), combined with other related computer-based bibliographic processing capabilities, offered a means for providing cost-effective automated services to small and medium-sized libraries. Such libraries, except in unusual circumstances, have heretofore been unable to afford the cost of developing or operating computer programs with sufficient power to take on a significant share of the library operations load or extend the range of library services.

AUTOMATION AS AN IMPROVEMENT OVER MANUAL PROCESSING

Automation has the potential to be an improvement over manual processing in two ways: (1) as a means of doing the same work at a lower cost, and (2) as a means of providing better or novel services to library patrons and staff.

In order to determine the actual value to libraries of the LISTS approach, answers to two questions were to be obtained from the experiment, if possible: (1) Is automation clearly an improvement over manual processing, regardless of cost? (2) Is the cost of automation tolerable, particularly for smaller libraries without access to computer support on their own? With LISTS, potential products are available from the computer that simply are not available in manual processes. For example, certain of the acquisitions reports that could be produced by the computer could not have been derived by manual means because the data are not readily available from typical manual files. (At least, the typical library does not maintain the files in such a way that most of the data could be obtained.) Nor is it possible, in typical libraries, to access catalog records by form of content, for example, "all encyclopedias" or "all conference proceedings." Such means of access into the MARC file are available using computer processing.* Extracting bibliographic

*The use of additional access points is discussed in a recent paper by Cooper: Cooper, William S. "The Potential Usefulness of Catalog Access Points Other Than Author, Title, and Subject." Journal of the American Society for Information Science, Vol. 21, no. 2, (Mar.-Apr. 1970), pp. 112-127.

data from the MARC file by simply identifying a record is obviously easier than keyboarding the record completely from scratch. While there seemed to be apparent advantages in LISTS over manual processes, it was not clear whether these functions would be attractive to librarians. LISTS was to explore that issue as thoroughly as possible.

COST-FEASIBILITY OF LISTS CONCEPT IN TARGET LIBRARIES

The problem of evaluating library automation on a cost basis comparative to manual methods has long been a vexing one because of the difficulties involved in isolating meaningful cost measurements for either traditional library procedures or automated procedures. This is especially true in light of the highly variable environment imposed by the substantial changes in objectives and practices which have taken place in the library and computer science field in the last decade. Nevertheless, head librarians can usually make a valid, if subjective, judgment as to what are reasonable costs for the various operations within their library. Thus, when presented with a cost for performing a library operation, such as book ordering, with computer assistance, these librarians ought to be able to express a reasonable opinion as to whether the cost is higher or lower than the current manual processing cost, or in any case whether the cost of the automated procedures seems reasonable. We did not believe that it was possible in a small-scale experiment to thoroughly test the difference for the libraries between the cost of using LISTS and the cost of manual processing. Only detailed study over a period of time, of either the existing system in the library or of the two systems running in parallel, would give meaningful figures. While that would be desirable, it was beyond the scope of this project. We could, however, determine the cost of the computer operations for the various LISTS processes and products, in terms of each subsystem--acquisitions, cataloging support, etc.--and in terms of each library participating in the experiment. These costs could then be evaluated by librarians in relation to current manual processing costs. Such evaluation should take into account the probability of lower costs for automation as the LISTS complex moved from a pilot system to an operational one, and higher costs for manual processing as salary scales inevitably rise.

II. BACKGROUND AND DEVELOPMENT OF LISTS

DEVELOPMENT OF LISTS

In the spring and early summer of 1967, the SDC staff began to plan for building LISTS based on the time-sharing system then being developed for the IBM 360/67 at SDC. During the summer of 1967, we examined each element of the SDC Technical Library's operations to see how we could implement computer support for them. We prepared a list (see Exhibit 1) of functions that we thought we could support with the time-sharing system. As we began gathering specifications for the various processes to be implemented on the computer, we began to make up a list of prospective participating local libraries, for we felt that an adequate test of our concepts could not be made without the use of the system in actual libraries during normal daily operations.

By December 1967, we had selected libraries to be approached for their participation. By the early spring of 1968, six libraries had agreed to participate, and one more was added in the summer of 1968. The final group of seven libraries that were to participate in the experiment consisted of:

- Beverly Hills Public Library
- Fullerton Junior College Library
- Pierce College Library
- San Marino Public Library
- System Development Corporation Technical Library
- University of California at Riverside Library
- University of Southern California Central Library.

During 1968, we designed the various programs of LISTS, taking advantage of the commercial data management system (CDMS) being built as part of the time-sharing system on SDC's IBM 360/67. However, at the end of the year it became apparent that CDMS was not sufficiently functional to meet the schedule for full support of LISTS. Accordingly, early in 1969 we redesigned LISTS to take advantage of another part of TS/DMS, an information storage and retrieval system that had recently been rewritten to operate on the 360/67 after several years operation on SDC's AN/FSQ-32 computer.

The operational phase of the LISTS experiment, under the sponsorship of the Council of Library Resources, was begun on March 1, 1969. The Library of Congress initiated its distribution service for MARC II tapes at the end of March 1969. By the end of April, the file searching program in LISTS was

EXHIBIT 1

LISTS PRODUCTS AND SERVICE AREAS
(Sheet 1)

I. Acquisitions of Monographs (New or Out-of-Print)

A. Production of orders to vendors or publishers from teletype input of either:

1. LC card number and quantity desired, or
2. normal author-title-publisher description

B. Creation and maintenance of an "in-process" file within the computer. Reports available optionally:

1. on demand by teletype with immediate response
2. on demand by teletype with off-line response
3. on a scheduled basis off-line

C. Creation and maintenance of out-of-print "wants" file. Reports as per (I.B) above.

D. Reports for:

1. Budget control

a. liens
b. special funds
c. forecasting of expenditures by various breakdowns, e.g., subjects, funds, vendors, etc.
d. expenditures-to-date by various breakdowns, as in (c) above

2. Prevention of unwanted duplication of orders

3. File access by several alphabetic sequences, e.g.:

a. author
b. title
c. subject
d. vendor
e. key-word-in-context

II. Cataloging Routines

A. Creation and maintenance of cataloging "in-process" file in common with acquisitions file (I.B). Reports available on same basis as for acquisitions (I.B, 1-3), above, i.e., by teletype or off-line, scheduled or on demand.

EXHIBIT 1

LISTS PRODUCTS AND SERVICE AREAS
(Sheet 2)

B. Production of cataloging aids, e.g.:

1. cataloging work sheets
2. temporary entries in card form or book form
3. authority file creation and maintenance
4. final catalog records in either card form or book form

III. Circulation Management

A. Loan file creation and maintenance with reports and notices produced for:

1. overdues
2. recalls
3. fines
4. use of the library collection

B. Borrower registration file created and maintained for such uses as:

1. use of the library collection
2. selective announcement service on newly received titles

IV. Serials Control

A. Serials acquisitions

1. Preparation of initial orders to vendors, as for monographs (I.A) above
2. Expiration warnings
3. Production of renewal lists

B. Serials records

1. Creation and maintenance of serial records file within computer. Reports available optionally:

a. on demand by teletype with immediate response
b. on demand by teletype with off-line response
c. on a scheduled basis off-line

2. Reports in the form of:

a. serial holdings in various possible alphabetic sequences, e.g., title, subject, classification number, language, etc.
b. serial "wants" or gaps in holdings
c. current receipts
d. items to be claimed

3. Bindery processing records, e.g.:

a. list of complete volumes ready to bind
b. bindery instructions for each volume or in list form
c. list of volumes at bindery

operational and sufficient MARC data had been accumulated that we could notify the participating libraries to order their Teletype terminals. As the terminals were installed, beginning in June, the experimental use of LISTS got underway.

Through the remainder of 1969, other LISTS programs, both for time-sharing and batch operation, were developed further and improved as we gained experience and received feed-back from the experimental libraries. A major revision of the file search program was released for use in September. The revised program incorporated a more powerful and truly user-oriented interface language. CDMS became operational in time to support circulation and serials control library functions. In fact, two-thirds of the way through the operating period LISTS had grown into a good approximation of the elusive "total library processing system."

OPERATIONAL DESCRIPTION OF LISTS

ACQUISITIONS SUPPORT

The acquisitions-support subsystem was built around the premise that, in many of the small to medium-sized libraries in the U.S., the majority of current acquisitions are English-language materials that are either new or reasonably current, and therefore correspond to the contents of the MARC file. The MARC tapes distributed weekly by the Library of Congress were processed by a conversion program that converted the original MARC record into a form usable within our time-sharing system. The converted records were stored in a catalog file for on-line use. The on-line MARC file did not contain the complete record as received from the Library of Congress. Exhibit 2 shows the file structure for bibliographic data extracted from the MARC tapes and stored in the time-sharing system.

In the acquisitions operation the librarian accessed the on-line catalog file from his terminal and could examine catalog entries in several ways. Entries could be retrieved from the file in their entirety, either on the terminal or on the line printer. In Exhibit 2, the elements that were access points in the file are indicated by X's. The basic procedure for accessing these records was to log in to the system, call a program named BIRCH (for BIbliographic SeaRCH), then, after identifying the desired catalog records, to use a second program called INFIG (for In-Process FIle Generator), inputting the additional information necessary to create a book order, such as name of the book fund, order number, and number of copies to be ordered. The system also allowed the creation of orders for materials not in MARC. To do this, the user would input from his terminal enough of the bibliographic description to enable a book vendor to identify and supply the desired item. If MARC-based records existed for a particular title, the user input only the LC card number to retrieve the bibliographic description from the file. The INFIG program captured that information and, in conjunction with the order information supplied by the user, created an output file from which multiple-part order forms were created using an off-line program under batch computer operation.

EXHIBIT 2

ELEMENTS OF THE ON-LINE MARC FILE

ELEMENT NUMBER	ELEMENT NAME	LENGTH	RETRIEVAL ELEMENT	QUALIFYING ELEMENT
01	LC CARD NUMBER	12	X	X
02	MARC DATE	6		X
03	LC CLASS NO.	8	X	X
04	DEWEY CLASS NO.	3	X	X
05	LC PRICE	20		
06	EDITION	20		
07	CONFERENCE INDICATOR	1	X	X
08	AUTHOR-CONFERENCE	64	X	X
09	SHORT TITLE	63		
10	TITLE	189		
11	SERIES NOTE	126		
12	PLACE OF PUBLICATION	20		
13	PUBLICATION DATE	4	X	X
14	LEVEL	1	X	X
15	PUBLISHER	63		
16	FORM OF CONTENTS	4		

LISTS provided a short form of operation for ordering books whose cataloging data was available in the MARC file. Short-form ordering required the user to input to BIRCH a list of Library of Congress catalog card numbers for the desired books; for those numbers matching the records in the MARC file, BIRCH output a computer-assigned accession number (and any other requested data such as author and title, handy for verifying that the card numbers were input correctly). The book ordering program would then be operated, with the accession number for each book input together with any non-standard order information, such as a special vendor, fund name, or requestor. For materials not covered in the existing MARC file, LISTS provided a long form of operation for ordering which allowed users to input the necessary bibliographic description. Only the simplest, most straightforward, technique could be provided during the LISTS experiment; the program asked the user to type in data, one record element per line. Exhibit 3 shows a record from the In-Process File.

If an in-process file were built for each library, programs would have been available to analyze acquisition operations statistically. Reports could have been produced showing, for example, the percentage of orders delivered in from 1 to 30 days, 31 to 60 days, 61 to 90 days, and over 90 days, all arranged in vendor order.

CATALOGING SUPPORT

Using the MARC data, LISTS could provide cataloging support in several modes; for example, the on-line retrieval program, BIRCH, could be used to print out bibliographic data from the MARC-based file, either on line or off line. These printouts could serve catalogers as copy for producing or correcting catalog cards, or as worksheets. Libraries willing to accept data from the MARC file with no changes could ask for catalog card sets having filing entries printed on the tops of the cards so that they were ready to file. Samples of catalog cards are shown in Exhibit 4. Catalog cards were produced from the original MARC tapes, not the on-line file. Therefore, subject headings and other data not in the on-line file were available.

Had we had subject headings in the on-line MARC-based file, catalogers could have used the on-line file as an authority list, with subject headings as access points. While the initial value of this technique during LISTS would have been small, the concept seems to have merit.

As part of cataloging support and, ultimately, for the benefit of library patrons, a catalog conversion bookform-catalog printing program was developed. The place of the bookform catalog within LISTS may not be immediately apparent. Its primary purpose was to carry the use of MARC records from the on-line MARC file through the acquisitions process and into the bookform catalog, and also into the batch circulation system. In order to be useful there had to be additional records in such a catalog other than those from MARC, because during the period of the LISTS experiment sufficient new works would not have been

EXHIBIT 3

AN ENTRY FROM AN IN-PROCESS FILE

LC CARD NUMBER	68-27449
ORDER DATE	270901
LC CLASS NUMBER	CS69
DEWEY CLASS NUMBER	929
LC PRICE	*BLANK*
EDITION	*BLANK*
LIBRARY CODES	X
AUTHOR-CONFERENCE	VIRKUS, FREDERICK ADAMS, MARQUIS, ALBERT NELSON, 1943.
SHORT TITLE	THE COMPENDIUM OF AMERICAN GENEALOGY.
TITLE	THE ABRIDGED COMPENDIUM OF AMERICAN GENEALOGY: FIRST FAMILIES OF AMERICA. A GENEALOGICAL ENCYCLOPEDIA OF THE UNITED STATES. EDITED BY FREDERICK A. VIRKUS UNDER DIRECTION OF ALBERT NELSON
SERIES NOTE	*BLANK*
PLACE OF PUBLICATION	BALTIMORE,
PUBLICATION DATE	1968 1925 C
LEVEL	*BLANK*
PUBLISHER	GENEALOGICAL PUB. CO., 1968 C1925-42
BOOK ORDER NUNBER	1235
ORDER STATUS	T
VENDOR-STAND	*BLANK*
VENDOR-NON-STANDARD	*BLANK*
REQUESTOR	*BLANK*
FUND	*BLANK*
NUMBER OF COPIES	*BLANK*
VOLUMES	*BLANK*
BINDING	*BLANK*
ARRIVAL DATES	*BLANK*
NET PRICE	*BLANK*
CALL NUMBER	*BLANK*
PURCHASEORDER NUMBER	*BLANK*
ORDER CODE	BBBC
VOUCHER NUMBER	*BLANK*
ITEMS RECEIVED	*BLANK*

EXHIBIT 4

CATALOG CARDS PRODUCED BY LISTS SYSTEM
FROM MARC II INPUT

```
MORGAN, WILLIAM L., 1927-
  THE CLINICAL APPROACH TO THE PATIENT
BY WILLIAM L. MORGAN, JR. AND GEORGE
L. ENGEL. ILLUSTRATED BY EVELYN LIPMAN
ENGEL.
  PHILADELPHIA, SAUNDERS, 1969.XI,
314 P. ILLUS. 24 CM.
  INCLUDES BIBLIOGRAPHIES.

1.MEDICAL INTERVIEWING.2.PHYSICAL
DIAGNOSIS.I.ENGEL, GEORGE LIBMAN, 1913-
  JOINT AUTHOR.

  RC65.M6              616      69-12887
  SDC
```

```
    MEDICAL INTERVIEWING.
MORGAN, WILLIAM L., 1927-
  THE CLINICAL APPROACH TO THE PATIENT
BY WILLIAM L. MORGAN, JR. AND GEORGE
L. ENGEL. ILLUSTRATED BY EVELYN LIPMAN
ENGEL.
  PHILADELPHIA, SAUNDERS, 1969.XI,
314 P. ILLUS. 24 CM.

  RC65.M6              616      69-12887
  SDC
```

```
    PHYSICAL DIAGNOSIS.
MORGAN, WILLIAM L., 1927-
  THE CLINICAL APPROACH TO THE PATIENT
BY WILLIAM L. MORGAN, JR. AND GEORGE
L. ENGEL. ILLUSTRATED BY EVELYN LIPMAN
ENGEL.
  PHILADELPHIA, SAUNDERS, 1969.XI,
314 P. ILLUS. 24 CM.

  RC65.M6              616      69-12887
  SDC
```

```
    ENGEL, GEORGE LIBMAN, 1913- JOINT
  AUTHOR.
  THE CLINICAL APPROACH TO THE PATIENT
BY WILLIAM L. MORGAN, JR. AND GEORGE
L. ENGEL. ILLUSTRATED BY EVELYN LIPMAN
ENGEL.
  PHILADELPHIA, SAUNDERS, 1969.XI,
314 P. ILLUS. 24 CM.

  RC65.M6              616      69-12887
  SDC
```

added to any of the libraries' collections to make a useful book catalog. One of the participating libraries, Fullerton Junior College, converted their catalog to machine-readable form so that LISTS could produce a bookform catalog for them.

CIRCULATION MANAGEMENT

LISTS included both on-line and off-line (batch processing) circulation control. The batch system was capable of producing a daily listing showing materials returned the previous day, listing the items currently on loan arranged by call number and giving short-form author and title, a borrower number, due date, the number of times the book has been renewed, and flagging items for which reserves or holds have been made. Overdue notices could be produced, as well as on-demand searches either for individual borrower records or for a class of borrowers: e.g., "all items out to the bindery," or "all items on reserve." In addition, a historical file was generated that, in time, could be used to yield statistics such as the number of patrons who have borrowed books, the average number of books borrowed by a patron, the average number of books in certain subject classes borrowed by patrons in certain borrowing categories, and so forth.

This system was implemented using an IBM 357 data-collection device, a machine-readable borrower identification badge, and a book card produced as a byproduct of the catalog conversion program.

An on-line circulation control system was developed for the SDC technical library. It provided for the input of data through a remote terminal to create a loan file from which several kinds of reports could be produced. On-demand searches for individual books by any element of the record, or searches by borrower, or overdue reports could be produced. A simulation of this on-line system is discussed in Appendix A.

SERIALS CONTROL

The serials control subsystem was based on the general-purpose data management system (CDMS). After serials records were converted into machine-readable form, the subsystem could create, maintain, and update a serials record file. It could, on demand, print listings (either on line or off line) in any of several formats specified at the terminal. It could retrieve records based on any element in the file. Exhibit 5 shows the elements of the serials control subsystem developed for the University of California at Riverside. Various products could be prepared from the files, such as claim notices when expected items fail to arrive, warnings of impending subscription expiration, statistics based on any elements in the file, wants lists based on keyboarded information on gaps in the library's holdings, etc. Records would be updated by information input from the terminal on materials that were not received when expected.

EXHIBIT 5

THE FILE SPECIFICATION FOR THE SERIALS CONTROL SYSTEM USING CDMS

```
 C1  ENTRY (NAME)

 C4  CALL NUMBER (NAME)

 C5  SORTING CALL NUMBER (NAME)

 C6  CLASS (NAME) FORMAT IS CXX

 C7  CLASS EXPANSION (NAME)

C27  COMMENT (NAME)

C28  PREDICTION CLASS (NAME) VALUES ARE I,P,U

 C2  FILING ENTRY (REPEATING GROUP)

 C3      ENTRY STRING (NAME IN 2)

 C8  PLACE (REPEATING GROUP)

 C9      LOCATION (NAME IN 8)
C10      COPY (RG IN 8)

C11          COPY NUMBER (NUM IN 10) VALUES ARE 1...99
C12          BACK FILE (NAME IN 10) VALUES ARE YES
C13          ACTIVE (NAME IN 10) VALUES ARE YES
C14          COMPLETE (NAME IN 10) VALUES ARE YES
C15          VOLUMES BOUND (NAME IN 10)
C16          VOLUMES UNBOUND (NAME IN 10)
C17          CURRENT ISSUE (NAME IN 10) VALUES ARE R,N,I
C18          MISSING ISSUES (NAME IN 10)
C19          SUBSCRIPTION (NUM IN 10) FORMAT IS 0000
C20          VENDOR OR SOURCE (NAME IN 10)

C21  CROSS REFERENCES (REPEATING GROUP)

C22      SEE ALSO (NAME IN 21)
C23      FROM REFERENCE (NAME IN 21)
C24      SEE (NAME IN 21)
C25      SUPERSEDED BY (NAME IN 21)
C26      SUPERSEDES (NAME IN 21)
```

A check-in list would be produced by the system, showing items to be received and also a listing of unpredictable items. (See Exhibits 6 and 7 for a sample report.)

REFERENCE SUPPORT

The MARC records lend themselves well to the production of special, subject-oriented bibliographies, or bibliographies of particular authors. The longer the MARC service is in existence, of course, the greater will be the likelihood that it will contain a significant number of entries for particular authors or subjects. At the end of the on-line operating period for LISTS, there were approximately 30,000 entries in the MARC file. It was possible to produce subject bibliographies using either the LC classification number or the three-digit Dewey Decimal Classification number.

Subject bibliographies, or listings of titles in particular locations, could also be produced from the serials in-process files to meet such needs as a location list or a recent-acquisitions bulletin.

These bibliographies, no matter on what basis they are selected from the file, could be produced on line at the terminal or off line at the computer center using a line printer. This off-line process was triggered from the terminal; off-line processing would be especially advantageous if the number of items to be included in a bibliography exceeded 20 or 25 records.

EXHBIIT 6

REPORT SPECIFICATION AND PROOF TO PRODUCE A LISTING OF SERIAL TITLES HAVING FOUR OR FEWER ISSUES REMAINING ON THE PRESENT SUBSCRIPTION

```
REENTER LINE OR CANCEL WITH A 'S'.
*REVIEW

      DATA BASE NAME IS RIVERSIDE SERIALS
      QUALIFY SUBSCRIPTION  WHERE C19 LQ 4
      SORT BY LOW VENDOR OR SOURCE
T2    T. SUBSCRIPTIONS DUE TO EXPIRE
T1    T. <REVIEW DATE:>, 'K MONTH
      MASK T1F2 = XXXXXXXXX
H1    H. <ENTRY>, VENDOR, ISSUES* REMAINING
      SPACE TO 38, 48 BEFORE F2, F3
C2    C. ENTRY, VENDOR OR SOURCE, SUBSCRIPTION
      MASK C2F1 = *XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
      SPACE TO 38, 50 BEFORE F2, F3
R2    R. VENDOR OR SOURCE = 'L <TOTAL>, COUNT ENTRY
      MASK R2F2 = 0000
NEXT



PROOF

      ....5....0....5....0....5....0....5....0....5....0....5.


T2                    SUBSCRIPTIONS DUE TO EXPIRE
T1                     REVIEW DATE:     XXXXXXXXX

H1    ENTRY                                          VENDOR     ISSUES
                                                                REMAINING

C2    *XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX               XXXXX        0000

R2    TOTAL     0000

      ....5....0....5....0....5....0....5....0....5....0....5.

NEXT
```

EXHIBIT 7

LISTING OF SERIAL TITLES HAVING FOUR OR FEWER ISSUES
REMAINING ON THE PRESENT SUBSCRIPTION

```
RUN
ENTER KEYBOARD DEFINITIONS.
'K 'MONTH' =:NOVEMBER
 REPORT GENERATION HAS BEGUN.    11/11/69   13:24
 DATA BASE NAME IS...    RIVERSIDE SERIALS

                SUBSCRIPTIONS DUE TO EXPIRE
                 REVIEW DATE:     NOVEMBER
```

ENTRY	VENDOR	ISSUES REMAINING
A.C.L.S. NEWSLETTER.	FAXON	4
A C S VOLUNTEER.	FAXON	3
A.D.A. WORLD.	FAXON	2
A.E.D.S. MONITOR.	FAXON	4
A.I.CH.E. JOURNAL.	FAXON	3

TOTAL 5

III. CONDUCT OF THE LISTS EXPERIMENT

PARTICIPATING LIBRARIES

INDIVIDUAL EMPHASES

When the participating libraries were first approached, each library was sent a letter listing potential areas of operation for the LISTS experiment. As we discussed the experiment with the libraries, it became clear that each library had in mind certain general areas in which its participation would be particularly useful or interesting. This list of potential operations is shown in Exhibit 1, in Section II. All of the libraries, with the exception of the SDC Technical Library, initially expressed interest in acquisitions; as the experiment progressed, certain changes were made, either because of changes in interests on the part of the libraries, or simply for the sake of expedience. It became clear that not every library could adequately test each of the areas in which it had originally expressed interest.

AREAS OF PARTICIPATION

Exhibit 8 lists the libraries, their holdings at the beginning of the LISTS experiment, the additions they anticipated during the fiscal year 1968-69, the number of staff, and the functions that were to be automated as part of LISTS. Exhibit 9 is a map showing the relative location of each library with respect to the computer at SDC.

The reader will note that the areas that were actually tested, as described in the following pages, differ somewhat from those shown in Exhibit 8. The Beverly Hills Public Library did test acquisitions, but was also interested in catalog cards. However, the format we were able to offer was not appealing to the librarian in charge of technical processing, so no catalog cards were provided. (This is discussed in more detail in Section V.) Fullerton Junior College actually experimented with all areas in which they had originally expressed interest. Pierce College experimented only with reference retrieval and acquisitions. San Marino Public Library, after experimenting briefly with reference retrieval from the MARC file, withdrew from the experiment. The SDC Technical Library staff experimented with reference retrieval as preparation for the on-line circulation control operation, but was unable to continue. The University of California at Riverside did not test any function, since there was difficulty in installing the remote terminal; they withdrew from the experiment. The University of Southern California did experiment with acquisitions and also (briefly) with some retrieval from the MARC file for cataloging purposes.

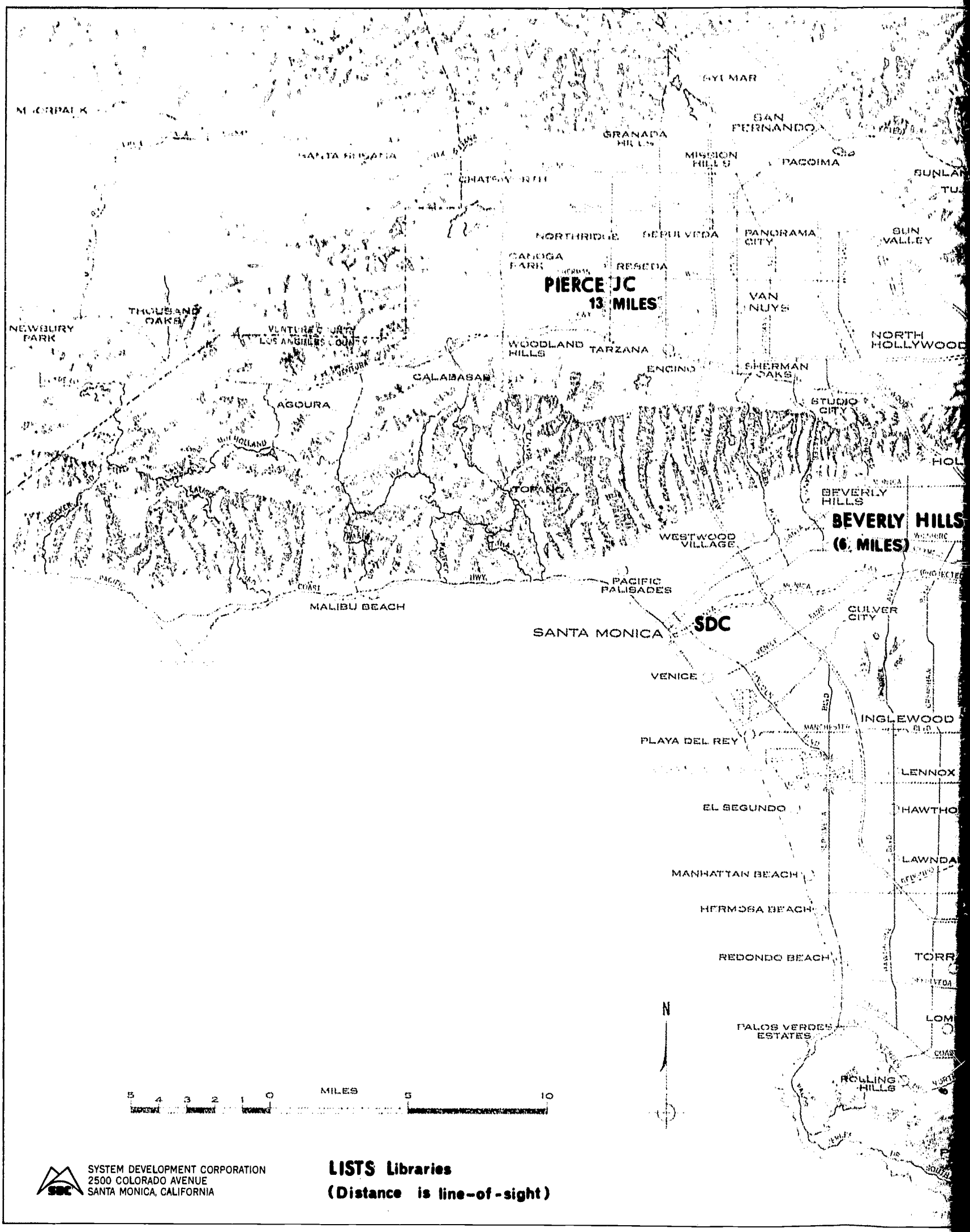
EXHIBIT 8

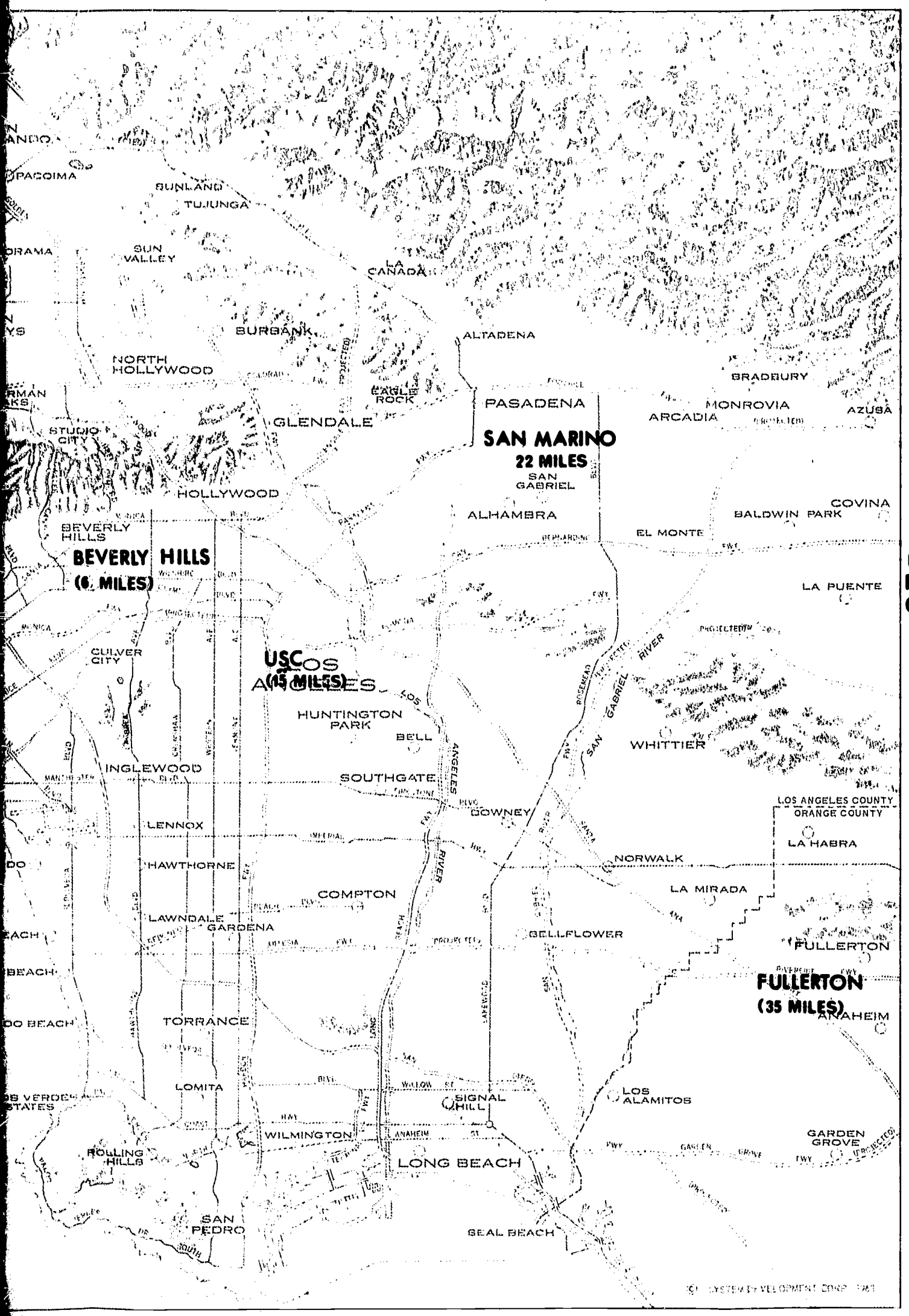
THE LISTS PARTICIPANTS

LIBRARY	PRESENT HOLDINGS		ANTICIPATED ADDITIONS DURING 1968-69 (volumes)	NUMBER OF STAFF	FUNCTIONS TO BE AUTOMATED
	VOLUMES	SUBSCRIPTIONS			
San Marino Public	60,000	250	4,000	12	Selective announcements; catalog of phono recordings; acquisitions
Beverly Hills Public	108,000	300	8,500	27	Acquisitions
Pierce Jr. College	60,000	700	5,000	--	Acquisitions; book catalog
Fullerton Jr. College	56,000	750	6,000	26	Book catalog; circulation control; acquisitions
University of California, Riverside	450,000	6,500	50,000	100	Acquisitions; serials control
University of Southern California	1,200,000	26,000	35,000	--	Acquisitions
SDC Technical	18,000 books and bound journals 86,000 technical documents	2,200	2,400 books 9,600 technical documents	24	Documents control; circulation control; serials control

EXHIBIT 9

LOCATION OF LISTS LIBRARIES RELATIVE TO SDC




PACOIMA
SUNLAND
TUJUNGA
SUN VALLEY
LA CANADA
BURBANK
ALTADENA
NORTH HOLLYWOOD
EAGLE ROCK
BRADBURY
PASADENA
MONROVIA
ARCADIA
AZUSA
GLENDALE
STUDIO CITY
SAN MARINO
22 MILES
SAN GABRIEL
HOLLYWOOD
ALHAMBRA
BALDWIN PARK
COVINA
BEVERLY HILLS
EL MONTE
BEVERLY HILLS
(6 MILES)
LA PUENTE
UC
RIVERSIDE
(70 MILES)
CULVER CITY
USC
LOS ANGELES
(15 MILES)
HUNTINGTON PARK
BELL
WHITTIER
INGLEWOOD
SOUTHGATE
DOWNEY
LOS ANGELES COUNTY
ORANGE COUNTY
LENNOX
LA HABRA
HAWTHORNE
NORWALK
COMPTON
LA MIRADA
LAWNDALE
GARDENA
BELLFLOWER
FULLERTON
FULLERTON
(35 MILES)
TORRANCE
ANAHEIM
LOMITA
SIGNAL HILL
LOS ALAMITOS
WILMINGTON
GARDEN GROVE
ROLLING HILLS
LONG BEACH
SAN PEDRO
SEAL BEACH

ORIENTATION OF LIBRARY PERSONNEL

The participating librarians were given numerous orientation sessions (ranging from a low of four, at USC, to sixteen, at Beverly Hills). On one occasion, all of the participating librarians were invited to SDC for a one half-day training session. The use of the terminal was designed to be as simple as possible, although the actual level of difficulty varied from subsystem to subsystem.

User manuals were produced and went through several editions during the course of the experiment. (The last edition of the Users Manual for the use of the on-line MARC file and for production of book orders is included in Appendix B; a manual for the use of the on-line circulation control system is included in Appendix D; and a manual for the batch process, circulation control system is included in Appendix C.)

Users were also furnished with more general manuals, provided by the SDC computer center for all users of the time-sharing service. That these were of any aid to the librarians is doubtful, since there was, perhaps, an overwhelming amount of material to be read, assimilated, and used. To inexperienced users of time-sharing services, these materials were, undoubtedly, formidable.

EXPERIMENTAL SERVICE

THE SDC COMPUTER AND TS/DMS

The original plan was to have a full six months of experimental operation for each library. It was expected that a nine-month operating period would be necessary in order to allow all libraries to have a full six months of operating experience. Because of the delays in obtaining terminals, the operating period was reduced to eight months.* At the beginning of the on-line operational period in June, the computer was available from 8:00 a.m. to 12:00 noon, Monday through Friday. Most of the functions of TS/DMS were available at that time, although the participating libraries were using relatively few of the total functions and programs available. Special programs within the overall general-purpose system had been created for using the MARC file and creating book orders. In September 1969 operating hours were increased on the computer to eight hours a day, from 8:30 a.m. to 4:30 p.m., Monday through Friday. However, the libraries were restricted to the period from 9:00 a.m. to 1:00 p.m., except for one library (Beverly Hills), where this particular period was not convenient. They used the system later in the afternoon than other libraries.

---

*See Exhibit 10 for the actual schedule of time-sharing operation.

EXHIBIT 10

ON-LINE OPERATION SCHEDULE OF LISTS LIBRARIES

NAME OF LIBRARY	JUN	JUL	AUG	SEP	OCT	NOV	DEC	JAN
Pierce Junior College								
Beverly Hills Public Library								
Fullerton Junior College								
San Marino Public Library								
SDC Technical Library								
University of So. Calif.								

COSTS OF SERVICES

At the beginning of the operational period, computer costs were based on a flat-rate charge for time-sharing of $31.00 per hour for some parts of the system, and a lesser rate for certain utility programs. Batch time on the 360/67 was $344.00 per hour, and on the 360/30 was $100.00 per hour. Costs for batch operation on the 360/67 were raised, effective July 1, 1969, to $443.00 per hour.

On-line use of the MARC file, creation of book orders, serials control, and on-line circulation control all made use of the time-sharing service at the above prices. Of course, each library was paying for telephone calls and had to pay a monthly charge for their terminal. The actual costs for the telephone calls varied from library to library depending on their distance from SDC. In many cases there was no charge at all due to the location of the library.*

Special forms were procured for the experiment because of the lack of satisfactory standard forms. This was the case with the book-order forms and the catalog card stock. Since these were special orders, the prices were higher than they would be under regular, ongoing procurement. (It would be possible to procure such special materials at much lower prices were operations to be carried on regularly so that quantity orders could be made rather than small unit orders.) The 8-part book order forms were $32.53 per thousand, and catalog card stock was $11.47 per thousand.

Personnel were necessary for pick-up and delivery, handling of forms, etc. During the experiment these were regular SDC employees, and their salary rates are probably higher than one would expect to pay for an operational system of this type.

The project manager acted as system manager during the time the libraries were actually "on the air" each day. Librarians in the participating libraries were encouraged to telephone anytime they experienced operational problems. This was frequent and the number of calls, while not recorded, was well over 100 during the six-month operating period. In most instances the project manager was the one who answered the phone when there was difficulty, but there was always someone available to give advice to the users. Obviously in an operational system, a much different personnel structure would be needed for such a system as LISTS. Operational personnel must be both knowledgeable and relatively inexpensive.

---

*There are some anomalies, due to the peculiarities of the Los Angeles telephone system. For example, Pierce College, at a straight-line distance of 13 miles from SDC, paid nothing per phone call, whereas Beverly Hills (6 miles) had to pay two message units for the first 3 minutes of every call.

DATA GATHERING

LISTS RECORD-KEEPING

The LISTS project originally anticipated several levels of data gathering as part of the computer system itself. The time-sharing executive system was to have had some statistic l data gathering capabilities. However, at the time of the experiment they had not been implemented. Had they been available, the terminal time and central processing unit time would have been printed out at the close of each terminal session. There was a satisfactory alternative, as described below.

The computer operations center, known as the Datacenter, used an accounting program based on the general-purpose data management system (CDMS). The accounting file for several months during the operational period of LISTS was made available to the LISTS staff, and QUERY and COMPOSE components of CDMS were used with this file to explore usage of the system by the libraries. Since log numbers are individualized, the system allows quite comprehensive evaluation of the data. Exhibit 11 shows the elements of the accounting file that were available. Statistical operators available within the system include count, sum, average, minimum, maximum, and standard deviation. Arithmetic capabilities include addition, subtraction, multiplication, and division.

Our primary retrieval program, BIRCH, was to have been able to record items of information such as the method of retrieval used, retrieval elements used, the number of catalog records retrieved, number of records printed on-line or off-line, dates of operation, time utilized from initiation to completion of each request, user comments, error messages output, etc. Here, too, the collection capabilities that were to have been available within the program were not because of the unexpected departure of the chief designer and programmer of the ORBIT system which the BIRCH program used. There was no way to analyze program use other than to review Teletype logs--a process that is slow and inaccurate, and required more manpower than was available.

Other programs within the overall complex of LISTS included elementary counting mechanisms, which were useful in evaluating or in managing the operation. For example, the program converting the MARC tapes to our on-line format kept track of how many records were processed, how many had been rejected, and the types of errors encountered. The file-generation program gave the unique index terms and the total number of postings against the indexed terms.* From the thousands of MARC records that were processed during the experiment, the number of unique terms in the file at its largest was 65,012, and the number of records posted against these terms was 138,951, an approximate 2 to 1 ratio. This was to be expected, since (1) the MARC records contained so many unique elements, such as LC card number, and (2) so few authors are represented by more than one work in the MARC file. The total number of records in the file was 24,215.

*See Exhibit 12

EXHIBIT 11

DESCRIPTION OF FILE ELEMENTS OF USER RECORD FILE USED TO COLLECT DATA ON USE OF TS/DMS

C1 LOGIN USER ID

C2 LOGIN ACCOUNT NUMBER

C201 LOCATION

C202 DIVISION

C203 TYPE

C204 NUMBER

C3 ORGANIZATION

C4 ADDRESS

C6 CUSTOMER NAME

C7 TITLE

C8 WORK ORDER NUMBER

C28 TAPE UTILIZATION MEASURED IN MINUTES

C29 OTHER DEVICE USUAGE MEASURED IN MINUTES

C27 DISK UTILIZATION FOR FIRST CYLINDER BILLING

C57 YEAR TO MONTH TOTAL DISK USUAGE IN MIN

C58 YEAR TO MONTH TOTAL TAPE USED IN MIN

C59 YR TO MO TOTAL OTHER DEVICE USED IN MIN

C14 TERMINAL DATA

C15 DATE OF LOGIN
C16 TIME OF LOGIN
C17 TIME OF LOGOUT
C18 TERMINAL TIME IN MINUTES
C19 CLASS 2 PROGRAM TIME IN MINUTES
C20 CLASS 3 PROGRAM TIME IN MINUTES
C51 YEAR TOTAL TERMINAL TIME
CONTINUE? (Y/N): Y

C52 YEAR TOTAL CLASS 2 PROGRAM
C53 YEAR TOTAL CLASS 3 PROGRAM

C21 PROGRAM DATA

C22 PROGRAM NAME
C23 TIME LOADED
C24 SIZE IN DRUM PAGES
C25 TIME QUIT
C26 CPU TIME USED IN SECONDS
C54 YEAR TOTAL DRUM STORAGE
C56 YEAR TOTAL CPU TIME

C30 FILE DATA

C31 UNIT FILE RESIDES
C32 VOLUME CONTAINING FILE
C33 FILE NAME
C34 FILE TIME IN MINUTES
C35 STORAGE UNITS
C36 SIZE IN CYLINDERS

EXHIBIT 12

TELETYPEWRITER LOG OF MARC ON-LINE
FILE UPDATE PROCESS

```
 25 INPUTS PROCESSED
 50 INPUTS PROCESSED
 75 INPUTS PROCESSED
100 INPUTS PROCESSED
125 INPUTS PROCESSED
150 INPUTS PROCESSED
175 INPUTS PROCESSED
200 INPUTS PROCESSED
225 INPUTS PROCESSED
250 INPUTS PROCESSED
275 INPUTS PROCESSED
300 INPUTS PROCESSED
325 INPUTS PROCESSED
350 INPUTS PROCESSED
375 INPUTS PROCESSED
400 INPUTS PROCESSED
425 INPUTS PROCESSED
450 INPUTS PROCESSED
475 INPUTS PROCESSED
500 INPUTS PROCESSED
525 INPUTS PROCESSED
550 INPUTS PROCESSED
575 INPUTS PROCESSED
600 INPUTS PROCESSED
625 INPUTS PROCESSED
650 INPUTS PROCESSED
675 INPUTS PROCESSED
700 INPUTS PROCESSED
725 INPUTS PROCESSED
750 INPUTS PROCESSED
775 INPUTS PROCESSED
794 INPUTS PROCESSED
\\\\\
THIS INPUT TAPE FINISHED. IS THERE ANOTHER TAPE? Y/N     N
PLEASE STANDBY WHILE INDEXES ARE PREPARED

YOUR INDEXING IS COMPLETED
NUMBER OF POSTING POINTS- - 63224   .
TOTAL NUMBER OF POSTINGS- - 134912  .
```

The Teletype log of all on-line operations by the LISTS staff was kept, and was available for analysis. Teletype logs from the participating libraries (as mentioned above) were also collected, and were found to be useful in analyzing the library's on-line operations.

USER INTERVIEWS

Even though there was regular interaction between the users and the LISTS staff, with many visits made to each library, at the end of the operational experiment a special team was sent around to visit each library and to conduct interviews according to a form that had been prepared for this phase of the project. (Appendix E contains a sample of this form.) The interviewers had not previously had much contact with the libraries during the course of the experiment. This was felt to be useful in gathering unbiased comments from the users.

MONITORING OF OPERATIONS BY LIBRARY STAFF MEMBERS

It had originally been planned that the library staff members in the participating libraries would keep track of their operations for subsequent evaluation in the experiment. This was rarely done because, in many situations, those using the terminal were not doing so as part of their regular job. For example, in at least one of the libraries, the primary terminal user was the head librarian. He did not normally type book orders, look up catalog records, or perform any other part of the technical processes at his library. Yet he was most interested in LISTS operations, and learned to use a terminal perhaps better than anyone else on his staff. The recording of his use of the system, however, was not particularly useful for comparison with normal, manual processes. In contrast to that situation, in other libraries so many people used the system for such short periods of time that their usage statistics were also not valuable for comparison purposes.

For some tasks, however, records were kept and were made available to the LISTS project staff for evaluation. These are discussed below, in Section V.

DATA ANALYSIS

Two procedures were used, one manual and one automated, for analyzing the data collected during the LISTS experiment. With respect to the time-sharing system, some computer-based programs were available to us to analyze the collected records. Some examples of this analysis are included in Exhibits 11 and 13. Exhibit 11 shows the elements of the file used for this analysis. Exhibit 13 gives minimums and maximums with respect to the use of two of the programs. Taking the CPU (Central Processing Unit) time, terminal time, and file size, anyone interested in computer processing of library records can calculate what it would cost to manipulate a file using the BIRCH program,

EXHIBIT 13

USE OF CDMS TO ANALYZE CPU AND TERMINAL
TIME FOR TWO LISTS PROGRAMS

```
PRINT MIN C26 , MIN C18 WHERE C22 EQ BIRCH
----

MIN CPU TIME USED IN SECONDS = 1.

MIN TERMINAL TIME IN MINUTES = 7.

----
REPEAT WHERE C22 EQ INFIG
----

MIN CPU TIME USED IN SECONDS = 0

MIN TERMINAL TIME IN MINUTES = 3.

----
PRINT MAX C26, MAX C18 WHERE C22 EQ BIRCH

----

MAX CPU TIME USED IN SECONDS = 67.

MAX TERMINAL TIME IN MINUTES = 82.

----
REPEAT WHERE C22 EQ INFIG
----

MAX CPU TIME USED IN SECONDS = 17.

MAX TERMINAL TIME IN MINUTES = 82.

----
```

depending on the cost of the machine at his computer center. Since prices vary from center to center, it is not particularly useful to know exactly what a particular job costs at a given installation, but rather what are the machine units used.

Wherever possible, operational costs were calculated in terms of equipment, programs, line charges, and supplies. The data to be analyzed did not require sophisticated mathematical manipulations. For the most part, the analysis involved using straightforward statistical operations, such as minimum, maximum, average, sums, and counts.

INFORMATION DISSEMINATION

PAPERS

Some dissemination of work on the LISTS project has already taken place. Before the execution of the SDC contract with the Council on Library Resources, a general description and overview of the system was given at the Annual Clinic on Library Applications of Data Processing held at the University of Illinois in May 1968.

In November 1969 the LISTS experiment up to that time was described at a meeting of the Public Library Executives Association of Southern California. A paper on the serials control subsystem of LISTS was submitted and accepted by the 1969 Annual Conference of the American Society for Information Science, and has been printed in the Proceedings of that conference.

At the Fall Joint Computer Conference in November ~~1970~~ 1969, in a post-session sponsored by the Special Libraries Association, a paper was presented on the use of the general-purpose time-sharing system for creating a library technical processing support system, and other aspects of LISTS.

It is recognized that these early reports on LISTS covered only the conduct of the experiment and obviously could not include any results, except those subjective results that were already apparent at the time of the various presentations.

The results of LISTS will now be made available for publication in various professional journals in the field of information science and librarianship, and as many talks, presentations, and demonstrations as possible will be scheduled in the coming months at local, regional, and national meetings of library associations or information science organizations.

IV. PROBLEMS ENCOUNTERED

We expected that an undertaking of the size and complexity of LISTS experiment would encounter many minor and some major problems. It did. The significant problems we ran into are described in this section; the experienced reader may chuckle over the problems with which he is familiar, and the inexperienced reader may note some things to beware of in working on future automation projects. We also had anticipated the need to reschedule parts of the project, and were correct in our anticipation: programs were not always operational when expected, the MARC tapes were almost one year late, terminals were not installed on time, and features of the time-sharing system were not available on schedule.

COMPUTER TIME-SHARING

Some of our earliest problems were due to software. Since we were using a time-sharing system that was itself in the process of development, our own programs operated incorrectly occasionally because of the way they interacted with the system programs. This happened again on several occasions as our on-line MARC-based file became larger and encountered hitherto unsuspected problems within the general time-sharing system. For example, one component of the time-sharing system known as SPAM, which handles input and output to terminals or other peripheral devices, had several "bugs" that we discovered in this way. The on-line MARC file was destroyed several times, necessitating expensive file regeneration from scratch.

The libraries, without exception, had difficulty in obtaining their terminals and getting them installed and operating properly. This varied from simple delays of a few weeks to the situation the University of Southern California encountered, in which an incorrect machine was installed several times before the correct one was finally installed.*

Telephone circuits continued to be a problem thoughout the course of the experiment for those libraries located at a considerable distance from SDC. Fullerton Junior College experienced many breaks in the telephone connection to the computer. This was probably caused by sharing an extension phone, and the problem could have been solved by using a private line (as they had been advised at the time of installation). Although the University of California at Riverside library withdrew from the experiment, we did try to demonstrate LISTS to them, only to discover that the telephone net in that area used

---

*This problem was finally traced to the Business Office of USC. The Library's order had been altered by some uninformed person who thought the teletype-writer was being installed for interlibrary loans.

3-wire rather than 4-wire circuits. It was necessary to boost the telephone signal with a voicepiece amplifier in order to achieve data transmission. Users occasionally got a busy signal when calling the computer's basic telephone number; in such cases it was necessary to try one or more other numbers to find an open line. Users also sometimes had difficulty in getting reconnected after an interruption caused by system or communication failure; in rare instances it was necessary to call the computer operator or LISTS monitor to solve the problem.

At the beginning of the actual operational period, the computer itself was fairly unreliable. Almost every day, the time-sharing system would be interrupted one or more times for several minutes. Since the LISTS staff was using the machine in the afternoon for further testing of file maintenance and update, and for the operations that were centralized, this proved to be rather disruptive for the whole experiment. File maintenance and update, especially for the MARC file, was a problem, since if the system failed during addition of new material to the file, the process would have to be started all over again by first deleting the existing file from the disk pack, copying from backup tapes back onto the disk, and then rerunning the file update program. During the early months of the experiment this sometimes happened several times a week.

FILE MANAGEMENT

As could be expected in a developmental program, a few errors were discovered by users. The most serious was a program limit of 10,000 for entries under any one index term; three-quarters of the way through the experiment it was painfully obvious that there were more than that number of records for monographs with a publication date of 1969. Fortunately, publication date was not a popular access point into the file. A second major problem was a program loop that occurred when two or more terms were linked in a search statement and neither could be found in the index. Neither error caused users a significant amount of trouble, and both were easy to circumvent.

Although not reported by librarians, the LISTS staff had an occasional problem with the length of listings for search terms having a multiple meaning. A truncation indicator may be used to retrieve all entries for all terms beginning with the same characters as the input search term, up to the truncation point; the program responds by listing all the corresponding terms in the index and requests the user to indicate which term he wants. Because many terms may begin with the same set of characters, a list of over 100 terms may be printed out, consuming many minutes. There is no easy way to interpret the printout process. Perhaps the librarians did not encounter this difficulty because they did not make use of the truncation capability.

The staff of the LISTS project were quite dissatisfied with the user/executive interface. The primary complaint regarded the low information content of system error messages. The other major problem encountered was inadequate user-oriented documentation for the command language. The staff wasted considerable time learning the "unwritten" laws of the system.

DATA BASE SUITABILITY

The lack of foreign-language publications in MARC severely curtailed LISTS' usefulness at USC, but the lack of retrospective coverage in MARC, combined with the serious time lag between publication and availability of cataloging information, was reported by all the participating libraries to be the principle drawback to more effective use of LISTS's capabilities.

Nor were the records on the MARC tapes processed without some difficulties. There were frequent errors in the tagging of elements, and in a number of instances the coding structure consistently did not conform to the standards set forth in the MARC handbook.

The serials file obtained from University of California at Riverside proved to be rather "dirty." In data processing jargon, "dirty" means that a file does not meet expected specifications--that it has extraneous characters or unexpected deviations from the specified format. The same type of problem presented difficulty in handling the Fullerton book catalog. This was compounded by the number of files necessary, since the converted catalog records had been input in relatively small increments.

MISCELLANEOUS PROBLEMS

LISTS users had varying degrees of difficulty in understanding some of the operations of the system. Some libraries changed personnel* during the course of the experiment, necessitating retraining that was, to some degree, unsuccessful because of lack of time. The location of the Teletypes in some of the libraries proved to be a problem because of noise or disruption of other activities, or, in some instances, because the curiosity of others in what the terminal operator was doing distracted the operator.

The off-line circulation-control system at Fullerton had been expected to be relatively trouble free. However, the IBM 357 data recorder proved to be a continuing problem. This machine broke down as frequently as once a week, and got out of alignment on an almost daily basis, causing the punches in the transaction cards to be off-centered. These cards then gave erroneous data to the circulation program, or could not be read into the computer at all. A second irritating problem was a program error that caused some overdue notices to be printed for books that had been returned to the library; librarians noted a tenfold increase in patron complaints after one mailing of the computer-produced overdue notices. (This was subsequently corrected.)

* including the head librarian in three instances

Fullerton also experienced difficulties in attempting to use a local data-processing service bureau to run the circulation-control program. The punched-card stock that the college purchased, and which was rather elaborately overprinted to their specifications, proved to be a non-standard weight, and tended to warp during the very low humidity imposed by warm local weather conditions. This warping caused the computer card reader to balk frequently. Although SDC's card reader could handle the cards without errors, the service bureau's reader was not so well adjusted and proved to be inadequate to the demands placed upon it even after the library switched to using standard card stock. In the end, the service bureau's handling of Fullerton's work was less than satisfactory, and the daily program run was done by SDC at Santa Monica.

V. RESULTS

LISTS USAGE

Each participating library had the use of the LISTS retrieval capability for searching the MARC file starting in June 1969 (or as soon as they had received their Teletype terminal). In November 1969, the final version of the book-ordering portion of the system became available, and each library was given the opportunity to experiment with it. The circulation subsystem was tested by Fullerton Junior College during the summer session and was operational through the fall semester. None of the libraries worked with the serials-control function, since this was to be left to the University of California at Riverside library, who later withdrew. A book-form catalog for Fullerton was ready for printing at the end of the experiment.

In general, the libraries used the on-line retrieval capability of LISTS only about 5% of the available time of about 480 hours. Fullerton was the heaviest user, but even there the system was operated irregularly, and in sessions lasting only an hour or so. The libraries used the book-ordering capability only once or twice. Once they had operated the system enough to gain some understanding of its features, the users tended to leave the terminals off except to try out system changes when these were announced and demonstrated. There were only one or two cases when LISTS was used to obtain information during the course of normal library operations.

There appear to be two major reasons for the low rate of usage for the on-line retrieval portion of LISTS: the limited bibliographic coverage available in MARC, and the press of normal library duties.

EVALUATION OF LISTS BY THE USERS

BACKGROUND OF THE LIBRARY PERSONNEL USING LISTS

LISTS users were about evenly divided between professional librarians and clerks or technicians. Their experience as librarians averaged between five and ten years, mostly in their present library. One librarian had only six months' experience and another had 28 years; both are currently employed in acquisitions. Half the library clerks were high-school graduates; the other half had some college or held baccalaureates. Only one of the users had any significant prior experience with computers or data processing.

Although most users must be considered highly educated and experienced, there was no indication that more than one user encountered an abnormal amount of difficulty in learning to operate the part of LISTS with which he was concerned. Both clerical and professional librarians operated the terminals

successfully and readily learned to use the on-line retrieval program in its basic form. While most users were relatively young, the few older persons involved reported no undue problems in learning to use the system, although they did express the opinion that they preferred to leave that type of work to younger librarians.

DEALING WITH THE TIME-SHARING EXECUTIVE PROGRAM

Because the command words used in dealing with the time-sharing system itself were devised by programmers for programmers, they were not especially well suited for non-technical users; however, the commands are few in number, and the main ones can be learned quickly (if not well understood). Most LISTS users found the commands of the time-sharing executive satisfactory for their purposes. Only one user expressed dissatisfaction with the command language, calling it "much too strict ... limiting." It seems fair to conclude that librarians can learn and use a short list of system commands even when such commands are not really designed for non-technical users.

No one reported having difficulty in using the Teletype terminal keyboard. Typing errors could be corrected by using the back slash character to erase the last character typed, or lines could be cancelled as a whole. This arrangement appeared to be satisfactory to the users. There were no complaints about slow computer response time, and no one indicated any special problem in pacing his operation to terminal or computer response speeds.

ON-LINE RETRIEVAL (THE BIRCH PROGRAM)

Despite the fact that the BIRCH program was infrequently used, the users claimed to be generally satisfied with the on-line retrieval program. It was apparent that they failed to understand and use the full range of BIRCH's capabilities and tended to be confused by some of its idiosyncracies. The three modes of conversation--long, short, or symbolic--were not found to be particularly useful; some participants always used the symbolic version, while others left program operation in the long version, which it uses automatically until a change is specified. Those using the long version were not annoyed by the lengthy, two-line printout each time BIRCH requested input from the user.

Most of the participants were confused by the Boolean operators AND, OR, and NOT, and failed to grasp the significance of linking search statements. No one reported making use of the NOT operator, and many stated that they often used AND in place of OR, or vice versa. Because of this failure to link search terms or search statements, users occasionally had the program limit of seven search statements active at one time. When this occurred, BIRCH would respond with a somewhat cryptic statement that left some users wondering what to do next.

Users said they found the most commonly used BIRCH commands to be generally satisfactory. However, several of the available commands were not used by anyone, and some users made do with just the PRINT command. Although some participating librarians claimed to be content with their control of BIRCH capabilities, it was obvious to the LISTS staff that the users failed to take optimum advantage of the available program commands. The prime example was that none of the librarians used the NEIGHBOR command to explore the contents of the index file for the MARC records. The NEIGHBOR command is analogous to to looking forward or backward a few entries in a card catalog, either for the purpose of browsing to see what is there, or for ensuring that you have the specific entry in which you are interested, or to verify the spelling of an author's name or the form in which the Library of Congress has used it. The failure of the librarians to use this powerful command was surprising. Their reason for not using it, according to most of them, was that they did not know about it. Since the LISTS staff did demonstrate the command in each library more than once, and the command was described and illustrated in the LISTS Users' Manual, it must be concluded that the users generally didn't recognize the value of browsing in the MARC file's index. An additional conclusion must be that the LISTS staff failed to provide sufficient training to the users, both personally and through the LISTS instructional material. This will be discussed further below.

The EXPLAIN command was reported the least satisfactory. EXPLAIN provides the capability for printing out on line the full description of a BIRCH command and examples of its use. The problem with using EXPLAIN is that the name of the command that is to be explained must be known and typed in by the user. In many instances, the participants attempted to use EXPLAIN (without using an appropriate command word) as a means of asking for help when they did not know what to do next or had received an error or warning message from BIRCH or from the time-sharing system. A secondary complaint was the wordiness of some of the explanations.

A few users were somewhat confused by the ERASE command. This command is used to erase some or all of the active search statements, and requires that the user specify which option he wants. Some users reported difficulty in specifying the option, primarily because they didn't understand how the command was supposed to work.

The PRINT command, which has a relatively large number of options, and thus has a high potential for confusing users, was generally reported to be satisfactory. The major suggestions offered by the librarians for improving it concerned adding additional options, such as a PRINT ALL for use after several search statements have been activated, or a capability for setting up a standard list of entry elements to be printed for a search series.

None of the librarians reported using the FIND, DIAGRAM, or RENAME commands. All had little importance for the simple type of retrieval required for general library work; they would be more useful in a report or journal citation retrieval environment (such as in MEDLARS and similar retrieval systems).

Users were satisfied with the capability for generating special bibliographies on- or off-line. However, in the instances where this capability was actually tried, results were unsatisfactory because the MARC file covered too short a time span to provide much useful information. Some librarians could see an advantage in being able to provide patrons or faculty members with special bibliographies on demand, but it would be highly desirable to have most or all of the catalog records for the library's holdings in the file to serve this purpose. For short bibliographies, the librarians posed no serious objection to the printout format used by BIRCH.

The great majority of participants in the LISTS experiment quickly lost any awe they might have had of the machine (computer or teletype) and readily gained a comfortable degree of facility in using the BIRCH program. However, they failed to use the on-line searching capability with imagination, seldom venturing beyond the limits of the training provided by the LISTS staff. The User's Manual was available, and other system manuals had been delivered to the librarians for whatever use they might wish to make of them. These included a text-editing program and various other components of TS/DMS. No one ventured to play with this system simply for the fun of using it* and finding out what else it could do, even though they had been repeatedly informed that nothing they could do at the terminal would harm the computer in the slightest and they were all well aware that the actual use of the computer cost them nothing except their communication costs.

Instead, many users became irritated, then frustrated, when they found that the information available in MARC was too limited and too late, and did not seem to serve any of the practical purposes they had expected it to. For example, one cataloger was excited by the system at first because it appeared to offer help for the bibliographic checking function and as an index to LC proof slips; however, since no foreign-language material was included (he knew this in advance), he soon gave up using the system. The acquisitions librarians in one of the libraries were disappointed and somewhat dismayed to discover that a number of entries in the latest National Union Catalog supplement were not in the MARC file.

---

*Some of the librarians were reluctant to use the system because the phone call to the computer cost them two to three dollars per hour, even though their library was committed to spend up to $2000 for communications during the course of the experiment. None of the libraries spent half that amount.

In sum, most of the participants felt that on-line retrieval was a good idea, but only if enough bibliographic data were in the file to permit selection and ordering of new books, preparation of cataloging materials, some retrospective bibliography, and indications of the holdings of one's own and associated libraries. These kinds of bibliographic data will, of course, be increasingly available as the MARC files accumulate. If Project RECON becomes operational, there should be a rapid buildup of retrospective data that may be of great assistance in building machine-readable catalogs for most public and school libraries. When MARC coverage is extended to foreign materials, university libraries will benefit greatly. Unfortunately, non-book materials--which are becoming increasingly important for all libraries--are not likely to be served by MARC or be under good bibliographic control in the near future.

ON-LINE ACQUISITIONS

The original plan for LISTS was to provide a capability for on-line maintenance of an in-process file for each library for books ordered through the system. Early in the operational period, the set of programs that performed this function were operational but not completely debugged; it was possible to place a day's worth of orders, but the file could not be updated reliably with new orders or arrival information. The design of the maintenance procedures for the in-process files was admittedly a clumsy one, even if the programs had worked correctly: within the constraints imposed by the funds available and the time frame provided for the LISTS experiment, it was necessary for file maintenance to rely on a set of existing programs, which were designed to perform an altogether different function. Nevertheless, the acquisitions function was completed to the degree that there was sufficient experience to evaluate from the standpoint of book ordering.

Users were generally satisfied with the short-form book ordering operation, * and found that the time required for ordering was about the same as, or slightly less than, that needed for typing out a normal book order. One acquisitions librarian was quite enthusiastic about this automated ordering capability, and would be eager to use it if the MARC data base were more complete and current.

The long form of ordering was excessively slow and awkward, but the user's main complaint focused instead upon the exasperating need for precision and attention to detail in inputting order information.

One user suggested that short-form ordering could be improved by eliminating the BIRCH step; instead, the order program should accept an LC card number or other identification for a book, then proceed to verify that a record exists in the MARC file. (The LISTS staff had considered this, but there were insufficient resources to do the required programming.) Another user suggested that orders should be captured on paper tape for transmission to

*Please refer to Appendix B for details of the book ordering procedure.

the computer at a specified time of the day. While such a procedure might speed up the initial preparation of a list of orders, the LISTS staff believes that the disadvantages of paper tape outweigh the advantages. Under a paper tape procedure, the orderer would have to wait for a day to find out what orders were not written because there was no MARC record; the danger of erroneous ordering would be increased by the small factor of unreliability in paper tape transmission and the likelihood of occasional keyboard mistakes; and the small savings in the initial costs for console and telephonic transmission would be overborne by the costs for error correction. It is obvious, nevertheless, that the book-ordering function for any automated library system will be subject to a great deal of engineering improvement as a result of operating experience, new computing machinery and methods, new terminal devices, new communications media, and new organizational procedures in the book trade.

THE BATCH-MODE CIRCULATION CONTROL SYSTEM AT FULLERTON

In spite of a number of problems with both the computer programs and the data recorder, Fullerton's head librarian has contracted for continued use of the circulation control system after the LISTS experiment ended. This is high praise for the system. Fullerton's use of the system demonstrated certain time-saving benefits. Manual compilation of circulation statistics, performed automatically by the computerized system, required 45 minutes or more each day. Circulation interfiling took 5-7 minutes each day. Computer printing of overdue notices saved about 1 minute per notice and did not require proofreading. Using the listing for books with hold requests saved about 1 minute for each request. There was a savings in time for charging and discharging books. In addition to these savings, the borrower search capability of the automated system provided a useful option not available to the library under its manual circulation system.

The major drawback experienced with the circulation control system was the unreliability of the data recording device. This machine failed more than once a week, on the average, requiring the library staff to fall back on the manual circulation control procedures for the average four-hour period required to bring the data recorder into service again.

In spite of mechanical and program problems, the LISTS circulation control program must be rated successful. The library staff, as well as students in training to be library technicians, readily learned to operate under automated procedures. Patrons were able to find out whether a book was on loan; under the manual system, the public-service staff was unable to spend the time to check through the circulation records to answer such questions. Fullerton gained better control over its collection, saved some clerical time and was able to give patrons improved circulation service.

TRAINING

Three methods had been used to train the participating librarians in operating parts of the LISTS system: instruction by the LISTS staff to individuals or small groups, in sessions of 2 to 3 hours; distribution of a User's Manual; and hands-on self-instructional practice. In addition, the BIRCH program incorporated an on-line tutorial to assist users with specific difficulties.

All the participants agreed that personal instruction from the LISTS staff, combined with hands-on practice at the terminal, was the most useful training method. This opinion is borne out by the fact that only those LISTS features covered at length in the personal instruction were actually used by the librarians. That training was sufficient to provide the librarians with a fairly good comprehension of the main purposes and design features of LISTS, and ground them in the basics of operating their part of the system. However, this relatively heavy investment in training time was not enough to bring the librarians to more than a minimum level of proficiency; it was obvious that more training would be desirable to bring the users to the point at which they could capitalize on all the system's features and operate it efficiently. In particular, a second round of instruction should be given two or three months after the initial period to serve as a review,* introduce more sophisticated concepts of operation, and provide an interchange of discussion on problems and discoveries. Our experience with LISTS emphasizes the importance that training ought to be accorded in planning, designing, and installing a library automation system. If the amount of time allotted to personnel training is sufficient, it is possible to go far in compensating for the defects and malfunctions to be expected of any new and complex system. Insufficient training results in early frustration on the part of the user, whereas adequate training and continued assistance from the system experts can obviate many user complaints.

The participants rated the LISTS User's Manual fairly low as a training medium. Their comments on the worth of the Manual were, however, partially contradicted by obvious failures to read it. If true that it was of low utility, the main reason may have been its imperfections; it was not a polished work of the manual writer's art. Changes were issued irregularly, and some parts were evidently unclear. The User's Manual for the batch circulation control system had more editorial effort expended on it and was well indexed; yet even it was not used very frequently by the staff at Fullerton. (Its main use was as a text for the library technician class given at the college!)

---

*This presupposes that the librarians will have used the system regularly and consistently during this time.

A good user's manual requires a heavy investment of skilled labor to produce and maintain. Such an expenditure is usually justified by the argument that the costs of personal instruction are even greater. The LISTS experiment suggests that that argument should be closely examined by library automation system designers; it may be cheaper and more effective, in the long run, to devote fewer resources to user manuals and spend more on personal instruction. Presumably, once the original staff in the library have been trained, they can provide on-the-job training with little effort to new staff members.

The on-line tutorial component of BIRCH was reported on by only a few users. One found it of value primarily as an index to the User's Manual--a use that had not been anticipated by the system designers, but one that seems worth exploiting in the future.

In general, the LISTS participants evaluated the training as a weakness in the system. A number of suggestions were offered for improving the user documentation, such as avoiding or clearly explaining any computer jargon, using film or slide presentations, or holding formal classes for system operators. Several participants also suggested that a set of manuals would be desirable, with each volume pointed at a specific purpose, such as an introduction to computers and library automation, an overview of LISTS, a simple operator's manual, and a detailed reference guide.

USER RATINGS OF LISTS OPERATIONS AND PRODUCTS

At the end of the experiment, users were asked to rank the benefit to their library of LISTS operations and products. On-line searching of bibliographic files and automated book ordering were ranked as most important, followed generally by circulation control and catalog cards. Of less importance were the preparation of bibliographies, a book catalog (except for Fullerton), computer-produced claim letters, and serials control. Accounting reports, a computer-based union catalog, and indexes to MARC entries were rated lowest.

These ratings probably should not be generalized, since each library tends to have its own set of priorities for automated assistance. Nevertheless, it must be noted that most of the areas ranked highest for automation are those for which time-sharing is not well suited. While on-line search and retrieval is useful for bibliographic searching, and as a selection tool, the bulk of the acquisitions function and production of technical processing materials can be done quite adequately--and more economically--by off-line (batch) programs. It would appear that an efficient and effective design for a major library system cannot be effected without careful integration of both on-line and off-line capabilities, using a wide range of computer hardware from very small to very large.

COST RESULTS*

Book Orders

To produce book orders from the MARC file required both time-sharing processing and batch processing. Using time-sharing, a user would require approximately 1.15 minutes per order at the terminal. There was also approximately 7 minutes of overhead time for the user each time he wished to use the program. (Overhead time consists of log in, log out, waiting for disk mounts, file opening and closing, etc.) We have reduced the cost figures to units of 100. One hundred book orders require 115 minutes of terminal time (plus 7 minutes overhead time) and 66.95 seconds central processing unit (CPU) time per 100 orders.

The LISTS staff had to run two time-sharing programs at SDC in order to complete the processing of information input by the user. The first required a minimum of about 10 minutes and a maximum of about 15 minutes. It is not particularly dependent on the number of orders being processed within the above limit (i.e., 100). Fifteen minutes would have easily accommodated 100 book orders. For 100 book orders, the CPU time for this program was about 4 seconds. The second time-sharing program also was not particularly dependent upon the number of orders being processed, and for 100 or fewer orders the time was about 10 minutes. About 4.2 seconds of CPU time were used. Adding these figures together, we have a total of 140 minutes of terminal time per 100 orders and 75.15 seconds CPU time per 100 orders, plus 7 minutes of overhead time for the first (user) program, for a total of 147 minutes terminal time for 100 orders. At the current flat rate of $38.00 per hour for terminal time, this would cost $93.10.

The batch processing involved processing the orders first through a program that looked up default values and formatted the records, and then through a program that would print the orders on the proper purchase order forms. Running and printing were separated simply because of the constraints of SDC's computer center, the operations of which did not lend themselves to mounting special forms during time-sharing. Most printing is done on a smaller computer as a secondary process. For the processing, then, of the orders: for 100 orders, the time is 1.5 minutes of CPU + input/output time, and for printing,

---

*Since the costs for computer time varied during the course of the experiment we have chosen to use current prices for computer time now obtaining at SDC. These prices are $38.00 per terminal hour for time-sharing and $443.00 per hour for batch processing on the large IBM 360 (360/67) computer. The cost for use of the 360/30, which is primarily used for printing, is $30.00 per hour. It is hoped that by giving terminal time, central processing unit time (CPU), and off-line printing time, the actual cost in terms of the machine resources used will be more meaningful than the particular prices charged at SDC's computer center.

using an IBM PN print train, the time is 3 minutes per 100 book orders. As a matter of interest, the cost at SDC averaged slightly over $7.00 per billable CPU minute, which in the above case would have been 2.5 billable units and would cost, therefore, approximately $17.50 per 100 orders. Printing was done at $30.00 an hour, and for 3 minutes would have been $1.50.

A brief word of explanation is in order. For the particular processing time quoted above, under the processing of book orders in a batch mode, SDC uses a charging algorithm originally developed by Douglas Aircraft Corporation.* This algorithm is used under Operating System 360 MVT (multiprogramming with a variable number of tasks). The time given is reasonably close to the time that one would use on any equivalent system, no matter what the particular charging algorithm might be. It should be pointed out that the Douglas algorithm does penalize input and output to a certain extent; however, the output is only a small part of this particular program; therefore, the penalty cannot be considered great.

The combined total for both time-sharing and batch processing of book orders would be $112.10 for every 100 book orders. For this cost one would have not only printed book orders but a machine readable file that could be developed into a complete in-process file were all of a given library's book orders to be processed through the system. To manipulate further the book orders in the machine file would have taken additional time-sharing time, and the additional cost would be similar to report generation costs of the on-line circulation control system (see Appendix A).

Catalog Cards

Catalog-card production did not make use of the on-line MARC file; rather, it went back to the original MARC tapes. The primary reason for this was that the on-line MARC file did not include all elements of the original MARC record; especially, it lacked subject headings. The second reason was the need for the expanded character set for catalog card printing. While the on-line system will accommodate upper and lower case, the terminal devices used in the participating libraries (the KSR 33 Teletype) lack lower-case letters. Therefore

---

*Billable time in minutes =

$$0.05 + \left[1.0 + .03D + (.0007 + .000001R)\ R\right] t$$

where D = number of data sets (each DD statement in the Job Control Language instructions = one data set)

and R = Region (size of core storage)

and t = (CPU time + 16.6 milliseconds) times (number of I/O operations)

plus a $\frac{1}{2}$ minute overhead charge for the complete job.

there was no value in having a lower-case capability in the on-line file. It would also have been considerably more difficult to have manipulated the expanded character set within the time-sharing system, since other applications running under time-sharing use ordinary binary-coded-decimal characters rather than the extended-binary-coded-decimal-interchange code needed for the expanded character set.

The catalog-card production program was operated strictly by the SDC staff and used magnetic tape for input and output.

Based on our experience during LISTS operations, the terminal time to process 100 MARC records and to expand them into a catalog card set was 21 minutes. The central processing unit (CPU) time per 100 MARC records was 1.44 minutes. Off-line printing of these catalog cards from the resulting tape proceeds at about 500 lines per minute using the IBM TN print train. The off-line printing, as in the other cases, cost $30.00 per hour.

The catalog cards produced (see Exhibit 3) were rather typical of library practice, in that standard indentation practices were followed. Second and third cards used a repetition of the heading on the first card, followed by the word "continued" to indicate the continuation. Subject headings were printed at the top of the card in proper indented style, as were other added entries. Since the computer printer uses 10-character-per-inch horizontal spacing and 6-line-per-inch vertical spacing, we allowed a 49-character line length, and averaged* 12 lines per card.

All added entries were picked up, including series (both author and title series), subject headings, and all partial-title entries. Statistics were not collected on the number of entries needing more than one card for each entry point. Multiple card entries seemed to appear frequently.

Based on the current SDC rate of $38.00 per hour, 100 MARC records would cost $13.30 to convert into card images and $1.80 to print off-line for a total of $15.10. This works out to a cost of $.15 per set, but of course it does not include the personnel time necessary, nor the cost of the special forms and the cutting and bursting of these forms, delivery, etc.

Batch Circulation

Some discussion has already been given of the conversion of the catalog at Fullerton Junior College to machine-readable form so that a circulation system could be created to use data collection equipment. The only costs that are available (and these are costs to Fullerton Junior College and not to the LISTS experiment) are the number of man hours of typing necessary to convert the

---

*Calculated on the basis of 1,000 MARC records.

catalog records, using a typewriter having an optically scannable typeface. The total is 1,767 hours. Optical scanning costs were approximately $500.00. Approximately 48,000 catalog records were converted.

Operating the circulation system on the SDC computer required .021429 hours per 100 transactions processed (the time is CPU + input/output time). This time is produced by the Douglas Aircraft Corporation algorithm discussed in the section on acquisitions. At the current SDC charges of $443.00 per hour for batch time, this would be $9.49 per 100 items processed or something slightly more than $.09 per transaction. Considering that daily circulation reports (listing all items charged out by call number), overdue notices (with borrower name and address printed thereon), listings of delinquent borrowers, listings of all returned books, borrower search reports, and a historical file could be produced from this system, this is not a particularly expensive process.

The batch circulation system in unique in several respects among the various subsystems of LISTS. For one, it is the only subsystem for which the project has any comparative statistics with respect to a previous manual system it supplanted. The other unique aspect is the fact that the batch circulation system actually superseded the previous manual system, whereas all other subsystems of LISTS were used solely on an experimental basis.

As discussed previously, while it is highly desirable to compare the costs of LISTS operations with those of manual processes, the project was not funded sufficiently to allow a study to be made of manual processes, and it had to be left to the staff in the participating libraries to gather whatever figures they could with respect to manual operations. Nor was there time to train the staff of the libraries to collect information in a systematic fashion. The figures for the manual circulation system were gathered, then, in an unsystematic way, and cannot be considered entirely accurate. They were not derived from time and motion studies, but from memory by the personnel involved, many of whom were part-time employees. Each staff member was asked to furnish data about some particular task he had performed as part of the manual circulation operation. The fact that the people involved were, to a large extent, part-time employees is important because it cannot be expected that students and other part-time employees will have any overview of the circulation operation and the way in which their own particular part of it fits into the whole. Since these part-time employees have many other activities, their work in the library is not necessarily of primary importance to them. Thus, the way they perform their jobs and the way they think about them cannot be expected to be similar to the way a dedicated, full-time employee would think and work.

The batch circulation program was also run on an IBM 360/50 at a service bureau in Anaheim, California, close to the Fullerton Junior College campus. Time on that machine was charged by a different method, so that despite the

fact that the equipment itself is much slower, the overall cost was about 80% of the cost on the 360/65* at SDC. This, of course, does not include costs such as courier service, handling, costs of tapes, punch cards, personnel time, etc.

Rather than give the detailed statistics concerning the manual operation, we will give some broad comparisons that might be helpful to some readers in evaluating the merits of an automated circulation system vs. a typical manual system.

The Fullerton Junior College previously used a single card circulation system filled out by the borrower. Approximately 4 hours and 15 minutes per week were necessary to record statistics. The automated system produced statistics automatically as a by-product. There is no filing necessary in the automated system.

The Fullerton staff seemed to feel that the automated system required almost as much time to handle reserve requests from students as had the previous manual system. However, under the new system, accurate records are kept of the process and requests are recorded in the file, whereas under the previous manual system the records may not have been so accurate and readily available.

Discharge procedures were slightly speeded up in the new system over the previous manual system.

All systems for circulation control have a potential for error. A fact often overlooked in manual systems is that errors are very frequent and almost impossible to discover except by accident. Only systematic searches can uncover the number of errors in a manually maintained circulation file, and almost no libraries ever do this, except perhaps when taking inventory. Even on those occasions it is rare that the number of errors found is recorded. Therefore, it is not possible to compare the benefits of accuracy in an automated system with the supplanted manual system, since there is no way to know how many errors there may have been in the previous system. (These errors take the form of misfiled records, the wrong card being pulled for returned books, call numbers copied incorrectly on charge cards, overdue notices incorrectly typed--sent either to the wrong borrower or listing the wrong book, etc.)

It is possible, of course, that there are also errors in the automated system. In general, though, these errors have to do with the human interface. For example, borrowers have been known to move book cards from one book pocket to another so that when the charged book is returned, it may not always have the

---

*The large IBM 360 at SDC runs as a 360/65 during operations under OS/MVT and as a 360/67 during operations under TS/DMS.

correct book card, and therefore the wrong book will be discharged (unless the operator of the data collection device is sharp-eyed enough to notice the discrepancies between call numbers on the book and on the book card).

There is a net saving on the processing of overdue notices, particularly in staff time. The other item that makes the automated system more efficient is the increased accuracy of the information placed on the overdue notice--that is, correct name and address, correct spelling of call numbers, authors, titles, etc.

Some things are possible in automated systems that were not practical in the manual system. For example, borrower searches, and search for "classes" of borrowers, can be made. It is now possible to do an inventory of faculty charges, and to produce lists of loans for each faculty member automatically. It is possible to do this in a manual file, but if the records are kept by call number, it means that they have to be hand sorted into faculty groups, alphabetized by individual, either photographed or typed, then sorted back into call-number sequence for refiling.

It is the opinion of the LISTS staff that the batch circulation operation is cost effective, based on the cost of $.09 per transaction. Information and experience gained during the experiment should enable the cost to be reduced somewhat below $.09 per transaction for the computer portions of the operation.

Bookform Catalog

The bookform catalog is a product that requires machine-readable catalog records from the individual library. (The costs of producing these within the LISTS experiment were borne by Fullerton Junior College and are not a part of the experiment. The Fullerton costs are discussed briefly in the section on Batch Circulation Control, since one of the products of the conversion were machine-readable book cards.) The process of producing a book catalog falls into four parts:

1. File building

   Building the initial file from converted, optically scanned characters requires one minute per 100 records of CPU + input/output time (using the Douglas Aircraft algorithm for OS/MVT); at a cost of $7.38 per minute, this is $7.38 per 100 records.*

---

*An initial record, which is the converted bibliographic data concerning one book, will be expanded, or "exploded," into several entries in the book catalog. This is analogous to a catalog-card set that develops, from the one main entry, the added cards for subject headings and other added entries. The expansion ratio in the Fullerton Junior College Library Catalog is 3.312.

2. Exploding and sorting the initial input record into the various representations--that is, author, title, subject--requires 1.05 minutes CPU plus input/output time per 100 entries, or $7.75 per 100 entries.

3. Formatting of author or subject catalog pages

   This process requires .096 minute per 100 entries of CPU plus input/output time, or $.71 per 100 entries.

4. Formatting a page of the title catalog

   This process requires .071 minute CPU plus input/output time per 100 entries, or $.52 per 100 entries.

This is a total of $16.36 per 100 entries (but note that the first item (1) above was for 100 records, from which more than 100 entries might be created).

For subsequent editions of a catalog--that is, for updating the master catalog--one has only items 2 through 4 above, for a total of $8.98 per 100 entries. Any new records would have to go through the file-building process, but the costs for that would apply only to the new records.

There are, of course, additional costs, such as personnel time, paper, special printing processes, binding, etc. All of these are extremely variable and dependent upon the time and place where they are carried out. One may ascertain what this amount of processing produces by inspecting the samples of the book catalog in Exhibits 14, 15, and 16.

On-line Bibliographies

The LISTS program allows the production of bibliographies (from the MARC file) either on-line or off-line, but both processes require use of the time-sharing system to retrieve the records; whether to print them on line or off line is a user option. Based on our considerable experience with the program used for on-line retrieval of the MARC records, approximately 2.46 seconds of CPU time are necessary for every 10 minutes terminal time. In addition, about 12 seconds CPU time and about 5 minutes of terminal time are required for overhead.

A skilled operator can do quite a number of searches in 10 minutes. However, the amount of printing on a terminal that could take place in 10 minutes is fairly small. A skilled user could very easily input 7 searches in the space of 10 minutes (once the diskpack is mounted and the program is started). Depending on what he is searching, the user might retrieve anywhere from 20 to 20,000 records in that space of time. Reasonable searches (reasonable being defined as approximately 100-200 records) could certainly be accomplished in

EXHIBIT 14

SAMPLE OF THE AUTHOR AND MAIN ENTRY CATALOG

ADAMS, Ephraim Douglass, 1865-1930
Great Brttain andthe American Civil War. BY **. New York, Russel, (1958?)
2 V. IN L. ILLUS.
E469.A25 43294

ADAMS, Geroge Burton, 1851-1925
The growth of the French na'ion. BY **. New York, Macmillan, 1896.
350 P. ILLUS.
DC39.A3 43287

ADAMS, Robert E, 1879-
The hardships of a Kansas homesteader. BY **. New york, Vantage, 1967.
F686.A3 2.00 1804

ADAMS, Nicholson Barney, 1895-
The herttage of Spain, an introduction to Ahspanish civilization. BY **. New york, Holt, (1943)
331 P. ILLUS.
DP48.A3 43302

ADAMS, Henry, 1838-1918
History of the United States during the administrations of Jefferson and Madtson. BY **. EnglewoodCliffs, N. J., Prentice, (1963)
2 V.
Contents.--1. Jefferson.
3E02.1.A253 43285

ADAMS, Henry, 1838-1918
History of the United States of America BY **. New York, Scribner, 1891-1901.
9 V.
Conte4ts.--I-II. The first administration of Thomas Jefferson. 1898.--III-IV. The second administration of Thomas Jefferson. 1898.--V-VI. The first administration of James Madison.901.--
E302.1.A234 43273

ADAMS, James Truslow, 1878-
History of the United States. BY **. New York, Scribner, 1933-
V. ILL,;S.
Contents.--I. The rise of the union.--II. A half-century of expansion.--III. Civtlstory
E178.A27 43411

ADAMS, Arthur E, ed.
Imperial R,ceful modernizatio4 or revolution? BY **. Boston, Heath, t1965)
108 P.
DK262.A53 43252

ADAMS, Samuel Hopkins, 1871-
Incredible era; the life and times of Warren Gamaliel Harding. BY **. Boston, Houghton Mifflin, 1939.
E786.A34 3.20 4298

ADAMS, Franklohin Pierce, 1881-comp.
Innocent m3 p. BY **.
PR1195.H8A3 43297

ADAMS, Leverett Allen, 1879-
An introduction to the vertebrates. BY **. New York, Wiley, 1933.
414 P. ILLUS.
QL605.A3 43413

ADAMS, Leverett Allen, 1879-
An introduction to the vertebrates. BY **. New york, J. Wiley, 1933.
QL605.A3 6508

ADAMS, Leverett Allen, 1879-
An introduction tothe vertebrates. 2d ed. BY **. New york, J. Wiley, 1938.
QL605.A3 1938 6509

ADAMS, Henry, 1838-1918
John Randolph. BY **. New York, Houghton, (1908?)
E176.A5 VOL.16 43274

ADAMS, George Plimpton, 1882-
SEE Knowledge and society.
Knowledge and society.
CB19.K55 (c1938

ADAMS, Thomas Francis, 1927-
Law enforcement. BY **. Englewood Cliffs, N.J., (1968)
HV7921.A35 6.95 16777

ADAMS, Henry, 1838-1918
The life of Albert Gallatin. BY **. New York, Peter Smith, 1943.
697 P. ILLUS.
E302.6.G16A3 43275

ADAMS, Georgia (Nein) Sachs, 1913-
Measurement a4dvaluation in education, psychology, and guidance. BY **. New York, Holt, (1964)
654 P. ILLUS
LB3051.A53 43289

ADAMS, Henry, 1838-1918
Mont-Saint-Michel and Chartres. BY **. New York, Houghton, 1936.
397 P. ILLUS.
DC20.A2 43276

ADAMS, Ansel Easton, 1902- illus.
SEE Brower, David Ross, 1912- , ed.
Notman apart; lines from Robinson Jeffers.
F869.B63B7 OVER (1965) 159 P. ILLU

ADAMS, Randolph Greenfield, 1892-1951
Political ishdeas of the American Revolution; Britan4ic-American contributions to the problem of imperial organization, 1765 to 1775. 3d ed. BY **. New york, Barnes & Noble, (1958)
216. ILLUS.
E210 A22 1958 43304

ADAMS, Charles J., ed.
A reader's guide to the great religions. BY **. New York, Free Pr., 1965.
Z7833.Z35 REF 6.95 8371

ADAMS, Mildred, 1894-
The right to be people BY **. Philadeled states
JK1896.A3 43299

ADAMS, Arthur E, ed.
The Russian Revol,tion·andbolshevik victory, why and how? BY **. Boston, Heath, (1960)
108 P.
DK265.17.A65 43253

ADAMS, Ben, 1913 or 14-1961
San Francisco,. BY **. Boston, Houghton, 1962
534 P.
F869.S3A291963 43254

ADAMS, George Burton, 1851-1925, ed.
Select documentsp. BY ** AND Great Britain. Laws, statutes, etc..
JN111.A4 43288

SAMPLE OF THE SUBJECT CATALOG

ACCULTURATION
Child, Irvin Long.::Italian or American? The second generation in conflict.
F104.N6C5 1943.

Doob, Leonard Willtam, 1909-::Becoming more civiltzed.-3New Haven, Yale Univ. Press, 1960.
GN320.D65

Foster, George McClelland, 1913-:: Culture and conquest: America's Spanish heritage.
F1408.3.F6 1960.

Handlin, Oscar, 1915- ed.::Children of the uprooted.
E184.A1H22 (1966)

Hunter, Guy::South-East Asia--race, culture and nation.
DS509.5.H84 1966.

Interuniversity Summer Research Seminar, University of New Mexico, , 1956. ::Perspectives in American Indian culture change.
E77.I55 1956 (1961)

Malinowski, Bronislaw, 1884-1942::The dynamics of culture change.
GN645.M3 1945.

Mead, Margaret, 1901-::New lives for old; cultural transformation--Manus, 1928-1953.
GN671.N5M44 1956.

Morner, Magnus::Race mixture in the history of Latin America.
F1419.A1M6 (1967)

Reichel-dolmatoff, Gerardo::The people of Aritama; the cultural personality of a Colombian mestizo village.
HN303.R4 (c1961

Ways of mankind.
GN400.W3 TAPE GN n.d. 28S. 12 IN.

World Federation for Mental Health:: Cultural patterns and technical change; a manual.
GN320.W74 (1953)

ACETYLENE
Goodrich (B. F.) company, Akron, O.:: Bricks withoutstraw; the story of synthetic rubber.
TS1925.G6 (1944)

ACHIEVEMENT MOTIVATION
Heckhausen, Heinz::The anatomy of achievement motivation.
BF683.H4 1967.

McClelland, David Clarence::Motivating economic achievement.
HD82.M29 (1969)

ACIDS
VanderWerf, Calvin Anthony, 1917-:: Acids, bases, and the chemistry of the covalent bond.
QD477.V3 (1961)

ACNE
Bobroff, Arthur::Acne, and related disorders of complex0 p. illus.
RL131.B6

ACOUSTICS
SEE
ARCHITECTURAL ACOUSTICS
HEARING
MUSIC--ACOUSTICS AND PHYSICS
SOUND 4055

ACQUISITIONS (LIBRARIES)
Carter, Mary (Duncan), 1896-::Building library collections. 2d ed.
Z689.C29 1964 1964. 287 P.

Goldhor, Herbert, 1917- ed.::Selection and acquisition procedures in medium-sized and large libraries.
Z689.G6 Champa

Tauber, Maurice Falcolm, 1908-::Classification systems.
Z696.T2 1961.

Wulfekoetter, Gertrude, 1895-::Acquisition work: processes involved in building library collections.
Z689.W85 1961.

ACROBATS AND ACROBATISM
SEE ALSO
GYMNASTICS 4056

Horne, Virginia Lee::Stunts and tumbling for girls.
GV545.H6 (1943)

La Porte, William Ralph, 1889-::The tumbler's manual.
GV551.L3 1938.

Ryser, Otto E::A teacher's manual for tumbling and apparatus stunts. 3d ed.
GV545.R9 1951 c1951)

Ryser, Otto E::A teacher's manual for tumbling and apparatus stunts. 3d ed.
GV545.R9 1951 c1951)

V-Five Association of America::Gymnastics and tumbling. 3d. ed.
GV461.V2 1959 (1959)

ACRONYMS
Gale Research Company::Acronyms dictionary.
PE1693.G3 REFER (1960)

Goldstein, Milton
PE1693.G6 REFER Dictio

Pugh, Eric::A dictionary of acronyms & abbreviations.
T8.P8 REF (1968)

ACTING
SEE ALSO
AMATEUR THEATRICALS 4057

Blunt, Jerry::The composite art of acting.
PN2061.B47 (c1966 450 P. ILLU

Crocker, Charlotte::Taking the stage; self development through dramatic art.
PN2061.C87 (c1939

Duerr, Edwin::Radio and television acting; criticism, theory, and practice.
PN4193.R3D8 (1950)

Eustis, Morton::Players at work; acting according to the actors.
PN2065.E8 (c1937

Franklin, Miriam A.::Rehearsal. 4th ed.
PN2061.F7 1963 1963,

EXHIBIT 16

SAMPLE OF THE TITLE CATALOG

IDOLS BEHIND ALTARS.
  Brenner, Anita
N6550.B7 1929. 359 P. ILLU

THE IDYLLS OF THEOCRITUS.
  Theocritus
PA4443.E5H3 1924 (1924)

IES LIGHTING HANDBOOK. 4TH ED.
  Illuminating Engineering Society.
TK4161.I45 1966 1966.

IF I HAVE FOUR APPLES.
  Lawrence, Josephine
Fiction 1935.

IF IT DIE, AN AUTOBIOGRAPHY.
  Gide, Andre Paul Guillaume, 1869-
PQ2613.I2Z52 (c1935

IF MEN PLAYED CARDS AS WOMEN DO, A COMEDY IN ONE ACT.
  Kaufman, George Simon, 1889-1961
PS3521.A727I4 c1926.

IF MEN WANT PEACE.
  If men want peace. BY Universities committee on post-war international problems. Washington branch. AND Harrison, Joseph Barlow, ed.
JC362.I35 1946.

IF MORNING EVER COMES.
  Tyler, Anne
Fiction 1964.

IF; OR, HISTORY REWRITTEN.
  Squire, john collings, sir, 1884-1958, ed.
D210.S7 (1964,

IF THIS BE TREASON;A PLAY IN THREE ACTS AND SEVEN SCENES.
  Holmes, John Haynes, 1879-
PS3515.O44I7I4 1935.

IF WINTER COMES.
  Hutchinson, Arthur Stuart Menteth, 1880-
Fiction

IF WOMEN MUST WORK.
  Brophy, Loire, Mrs.
HD6053.B745 1936 153 P.

IF YOU CAN COUNT TO FOUR. (5TH ED., REV.)
  Breckenridge Jones, James, 1917-
BF637.S8E7 1966 (1966) 246 P.

IF YOU DON'T MIND MY SAYINGSO...
  Krutch, Joseph Wood, 1893-
AC8.K84 1964.

IF YOU MARRY OUTSIDE YOUR FAITH; COUNSEL ON MIXED MARRIAGES. REV. 2D ED.
  Pike, James Albert, Bp., 1913-
HQ1031.P5 1962 (1962)

IF YOU REALLY WANT TO KNOW: A CATCHER CASEBOOK.
  Marsden, malcolm m, comp.
PS3537.A426C34 (1963)

IFE IN THE HISTORY OF WEST AFRICAN SCULPTURE.
  Willett, Frank
NB1097.W4W5 (1967)

THE IF'S AND OUGHT'S OF ETHICS.
  De Boer, Cecil
BJ1011.D4 1936.

IG FARBEN.
  Sasuly, Richard
HD9654.9.I5S3 1947.

IG FARBEN.
  Sasuly, Richard
HD9654.9.I5S3 1947.

IGNEOUS ROCKS AND THE DEPTHS OF THE EARTH-
  Daly, Reginald Aldworth, 1871-
QE461.D212 1933.

THE IGNORANT ARMIES.
  Halliday, Ernest Milton, 1913-
DK265.42.U5H3 (1960)

IGOR STRAWINSKY.
  Armitage, Merle, 1893-ed.
ML410.S932A7 1936 158 P. ILLU

IGY: YEAR OF DISCOVERY; THE STORY OF THE INTERNATIONAL GEOPHYSICAL YEAR.
  Chapman, Sydney, 1888-
QC801.3.C45 (1959) 111 P. ILLU

III TRAGEDIES: BLOOD WEDDING, YERMA, BERNARDA ALBA.
  Garcia lorca, federico, 1899-1936
PQ6613.A763A27 (1947)

IKEBANA: THE ART OF JAPANESE FLOWER ARRANGEMENT.
  Davidson, Georgie
SB450.D3 1967.

IL DUCE; THE LIFE OF BENITO MUSSOLINI.
  Hibbert, Christopher, 1924-
DG575.M8H47 (1962)

IL PENTAMERONE; OR, THE TALE OF TALES.
  Basile, Giovanni Battista, 1575 (ca., -1632 AND Burton, Sir Richard Francis, 1821-1890, tr.
PQ4607.B5Z34 (c1927 455 P.

IL TROVATORE. HIGHLIGHTS.
  Verdi, Giuseppe, 1813-1901
M1508.T7V4 TAPE LM 2617 2S. 12 IN.

IL TROVATORE. HIGHLIGHTS.
  Verdi, Giuseppe, 1813-1901
M1508.T7V4 TAPE LM 2617 2S. 12 IN.

IL TROVATORE. SELECTIONS.::Il trovatore. Highlights.
  Verdi, Giuseppe, 1813-1901
M1508.T7V4 TAPE LM 2617 2S. 12 IN.

IL TROVATORE. SELECTIONS.::Il trovatore. Highlights.
  Verdi, Giuseppe, 1813-1901
M1508.T7V4 TAPE LM 2617 2S. 12 IN.

THE ILIAD.
  Homerus
PA4025.A35 TAPE TC 1193 2S. 12 IN.

THE ILIAD.
  Homerus. AND Butler, Samuel, 1835-1902, tr.
PA4025.A2B85 196 (1964)

ILIAD::The complete works of Homer.
  Homerus AND Lang, Andrew, 1844-1912, tr. AND Leaf,Walter, 1852-1927, tr. AND Myers, Ernest, 1844-1921, tr.
PA4025.A1L3 1935 (1935)

the 10 minutes, provided that the user had some idea of what he was searching for. If a user had retrieved 200 items and had asked for an off-line listing of the retrieved records, it would have taken him approximately two minutes additional terminal time to transfer the retrieved items to a magnetic tape (from the disk file) for subsequent off-line printing. The actual number of lines printed off line would depend upon the elements requested for printing by the user,* but the printing process would proceed at 1100 lines per minute. The actual printing time would be very short. Off-line printing costs $30.00 per hour (to the nearest minute thereof). We might say, then, that to retrieve and print off line 200 MARC records would require 17 terminal minutes; at the cost of $38.00 per terminal hour, this would be $10.77 (including printing).

---

*See LISTS User's Manual in Appendix B for details of the BIRCH program.

VI. CONCLUSIONS

VALUE OF THE EXPERIMENT

LISTS was an experimental test of the usefulness to libraries of a time-shared general-purpose data-management system, using the centrally prepared MARC II bibliographic records distributed by the Library of Congress. As an experiment, we believe the project has been highly successful in testing the concepts set forth in Section I. In answer to the question, "Is automation clearly an improvement over manual processing?", our answer must be somewhat equivocal. In some cases the answer is "yes," and in others "no."

The "yeses" include book ordering, bibliographic search in the MARC records, circulation control, and the bookform catalog. In the book-ordering process, we believe that the potential value of the MARC file will be shown when more bibliographic data are available and when the speed of cataloging at the Library of Congress increases. When that occurs, automated book ordering will be more clearly preferable to manual processes. It is not clear that direct input on a terminal is preferable to batch input if the Library of Congress card order number is known; if that number is not known, then on-line search by other access points is clearly useful, and is not now available (except for author search) in manual operations with traditional bibliographic tools.

There are undoubted advantages in having direct access to bibliographic data in the on-line MARC file but it is probable that most librarians do not know how to exploit this capability in terms of their own technical processes, or in terms of its potential value to library patrons. This statement is not meant to cast aspersions on librarians; it is rather to suggest that considerable training will be needed before they will be able to exploit this entirely new and quite complex capability. Automated circulation control is clearly preferable to manual methods (except in a very small library, one with less than 10,000 circulation per year). The LISTS experiment is, however, not the first to point that out.

Bookform catalogs have many uses, and most of their advantages over a card catalog are inherent in the book form. They cannot, however, be produced by other than automated means for any extended period of time.

In all the experimental service areas except that of the serials control subsystem, the information gathered during the experiment is a sufficient basis on which to decide whether automation offers particular advantages over manual processing.

The second question asked in Section I was "Is the cost of automation tolerable...?" Again, the answers are not simple. Some of the LISTS processes seem to be cost-effective, but others are clearly not. The experiment indicates that it may be possible to improve some of the subsystems to the point

at which they would be cost-effective. Time-sharing does offer advantages over batch processing for direct service to library patrons, but is still too expensive.

The experiment also showed the critical nature of the speed with which any central agency, such as the Library of Congress, disseminates the information it produces. One of the biggest disappointments in the project was that catalog records seemed to take so long to appear in the MARC file after books they describe had been published. This drastically limits the usefulness of the MARC tapes for many libraries, although it is obvious that the existing pattern, in some libraries, of ordering books prior to publication (only to hold the books on the shelves in the processing department for weeks or months pending the receipt of catalog copy from the Library of Congress) is not particularly beneficial to the library patron, either. A library could just as easily wait to order a book until the catalog record appeared in the MARC file; cataloging could then be accomplished immediately. The net waiting time would probably be about the same as in present practice.

Perhaps the most important result of the experiment for the participants was the understanding they gained of what automation is likely to offer for library applications, together with an appreciation of how much time and effort must go into making a major computer-based system truly useful. Although few users felt that they could not understand and adjust adequately to the skills required for operating a LISTS-type system, there was a concern among many participants that it was necessary to gain some familiarity with the jargon used by automation specialists. While communication between LISTS experimenters and participating librarians was enhanced by the LISTS staff's familiarity with both the library and computer science fields, there were many times when terms from both fields had to be carefully defined and explained to avoid misunderstandings.

As one participant pointed out, librarians do not need to know much about computer science; however, it is important that librarians have some familiarity with its concepts and terminology if they are to make effective and imaginative use of library automation. Computer and systems experts can provide the tools for library automation, but librarians themselves are responsible for deciding how those tools can best be used.

The LISTS participants gained a healthy appreciation of the complexities and uncertainties of working with a large-scale computer-based system. They learned that creating such a system can be a long and sometimes frustrating process--one that frequently takes far longer than originally scheduled. They also learned that computer-based systems, at this stage in the art of automation, are always improvable. All of us have come to realize how large the investment in automation is likely to be before many libraries reap substantial benefits. Thanks to their LISTS experience, many of the librarians cooperating in this experiment will inspect with a knowledgeable skepticism the estimated costs of any proposals to automate their libraries.

VALUE AND POTENTIAL OF THE LISTS CONCEPT

One of the critical needs in librarianship at the present time seems to be the education and training of librarians for future library operations that will most likely involve automation. The tools and techniques developed as part of the LISTS project are available and can be used for training purposes. These general-purpose programs can be adapted readily to different files, and can be used to model a system that can be produced in a more efficient form by special programming. Meanwhile, LISTS, as developed, can be used for testing ideas and training personnel for future automation.

It is clear that TS/DMS is not satisfactory for a production operation. Library files are very large, and TS/DMS cannot now support large files for more than one user, although, theoretically, the system can support many more disk drives than it now has. It appears that a dedicated machine is preferable for bibliographic operations. This conclusion has been reached by other investigators (e.g., the BALLOTS project at Stanford University).

If automated library systems are to pay off in lower costs and improved information services, there must be an adequate bibliographic data base. Many systems, generally at the State library level, are planning to provide much of the needed bibliographic data on their own by converting existing catalogs and capturing data at the point of initial processing. It would seem to make better economic sense to provide sufficient funds for the Library of Congress to (1) provide an adequate level of machine-readable cataloging service nationwide, and (2) cooperate with the national libraries of other countries to provide a common standard for international bibliography. It seems unlikely that many libraries should subscribe directly to the MARC II service in its present form. Why, then, should not regional centers be developed to maintain MARC records in on-line files to be accessed by libraries within the region? Clearly, a national center is not now feasible because of the communications cost; regional centers might well be, especially if they used smaller computers dedicated solely to library processing. Technology itself is changing rapidly. Within the lifetime of the LISTS project, disk storage devices have more than doubled in capacity, with a proportionally smaller increase in cost. There is no indication that future advances will be any less dramatic. With only minor modifications, LISTS could readily handle a regional bibliographic union file of MARC and local records.

If enough institutions were to convert their catalog records, it would soon be possible for most other libraries to convert their own bibliographic records simply by identifying their own holdings in listings of the already converted records. The file structure and retrieval programs developed for LISTS appear to be quite capable of handling files of up to 2 million catalog records. The vast majority of American libraries have holdings much smaller than that. All that is really necessary to proceed is the initial conversion effort.

Not only would a large, cooperatively developed file of bibliographic data help individual libraries with their own conversion, but it could be used as a regional union catalog. Many State libraries are known to be considering the possibilities of developing automated statewide systems. LISTS is a tool that is available now for furthering such projects. With on-line access of this sort, it would be necessary to print catalogs only at infrequent intervals. Such a union catalog would make vastly greater resources available to the patrons of any one library than have been available in the past, and without the need to wait to learn whether or not desired materials are available at a distant library, since the file could be queried, on low-cost terminals, instantaneously.

We believe that the LISTS experiment has given at least some indication that it is more efficient, in the long run, for libraries to join together to develop a system that all can use on a common computer than to continue to develop a separate system at every library for use on its own equipment.

APPENDIX A

THE ON-LINE CIRCULATION SIMULATION

I. INTRODUCTION

It had originally been planned that the SDC Technical Library would play a significant role in the LISTS experiment, by utilizing whatever elements of the total system seemed reasonably useful in their operations. Because of certain unique methods of operation, it was decided (after the experiment had gotten underway) that the on-line circulation system was the only component that had immediate value to the SDC Technical Library. The serials control subsystem would also have been valuable, but to use that subsystem would have required that the file of serials records be keyboarded for input into the on-line system. The Library lacked sufficient manpower to undertake a keyboarding task of this nature. The same disadvantage might be expected with respect to the on-line circulation system; however, in this case it was our original intention that the on-line system would simply be started on some particular day with that day's input, and that while there would be a workload it would be spread over each day and not be unmanageable with the existing staff at the SDC Library.

Unfortunately, an SDC corporate reorganization changed the Library's operation in such a way that the circulation operation was discontinued. The Library was, therefore, unable to participate.

Since most of the planning had already been done, it seemed profitable to attempt to simulate the circulation system, using the LISTS staff and secretaries in the Library and Documentation Systems Department. For input, we took a dump of the last overdue file from the old SDC Library circulation system. That system was a punched-card operation in which a single Hollerith card was used for each item in the file. Exhibit 1 shows a sample record from that file.

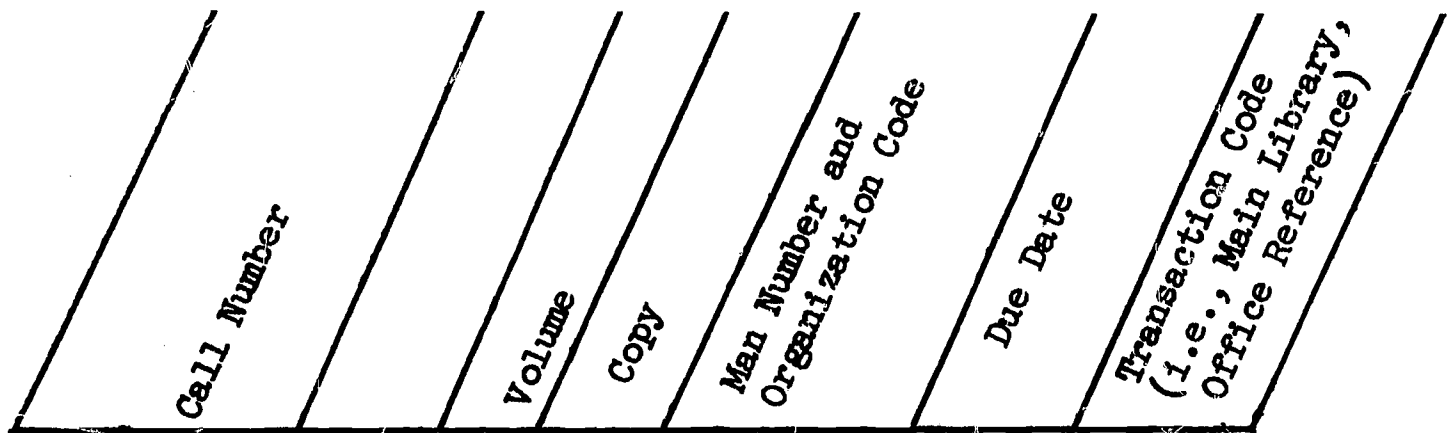
This limited amount of information was not, of course, all that we had planned to put into the new system; it suffices for a simulated system, since additional elements would work in the same way that the existing elements work in terms of retrieval, or in their use in reports, etc.

II. THE SIMULATION

The simulation was not a mathematical simulation, but simply a group of people pretending that they were library staff and inputting records into the computer system as the library staff would have done. That the person who keyboarded the information into the time-sharing system was not a library clerk but a secretary is of concern only in terms of the input rate she was

EXHIBIT 1

A LOAN RECORD FROM THE FORMER CHARGING SYSTEM
OF THE SDC TECHNICAL LIBRARY



able to achieve, not being intimately familiar with the data elements used in the circulation system or having any previous experience with time-sharing or terminal operations. The secretary's task was to read from a printout made from the previous circulation system, translate a string of numbers and letters into the circulation system file structure, and code each element according to the instructions contained in the on-line circulation system users' manual. This manual is included in this report in Appendix D.

After an initial training period of two hours, the secretary input data every afternoon for a period of about two weeks, using the time-sharing system during a two-hour period late in the day. The total number of records input into the system was 180 (see Exhibit 2 for details of the file).

The programs used for creating the file were standard programs available within TS/DMS. Initially, the GENERATE program was used, but it proved unreliable, since any number of errors on the part of the keyboard operator would wipe out all input for that day. Instead of GENERATE, we decided to use a program called EDIT, a general text-editing program available as a utility within the time-sharing system. EDIT allows correction of errors, moving of a line from one position to another, copying lines from one location to another, making corrections of the basis of character strings found within a line, and so on. EDIT also has the virtue of preserving the input file in case the GENERATE program encountered too many errors to complete the processing.

In a real-life operation, the file would have been updated daily; in our simulation, an accumulated file was added every three or four days, and a new CDMS file created. The file structure itself was created by a program called DEFINE, which allows for the file structure to be established and changed as necessary. (See Exhibit 3)

For operation of the system after the data had been input, two programs were used: one was the QUERY program that is the standard retrieval program for CDMS. With this program questions such as, "What books are on loan to borrower number XXXX?" can be answered instantly on the terminal. (See Exhibit 4) Also, this program would have been used to count the number of loans made each day with a question phrased, "PRINT COUNT CALL NUMBER WHERE DUE DATE EQUAL XX AND LOCATION EQUAL XX." The librarian would select the due date that had been input that day for all new loans (or any other date he wished to check for the total number borrowed) and use that date as the date in the QUERY statement shown above along with the proper location, Branch or Main. The question could be asked a second time for the main library, or it could be asked for both libraries without specifying the location, simply by asking for a count of the loans for a particular due date. (See Exhibit 5)

Certain other types of statistical information useful to the librarian were not available by using the QUERY program and would be produced by a report-generation program called COMPOSE. A number of reports were created, named,

EXHIBIT 2

SIZE OF THE FILE OF LOAN RECORDS
FOR THE SDC LIBRARY SIMULATION

NUMBER OF ELEMENTS	13 ELEMENTS.
NUMBER OF VALUES	1437 VALUES.
NUMBER OF VALUES THAT FAIL	903.
SIZE OF DATA BASE	73440 BYTES.
SIZE OF VALUES	5532 BYTES.
AVERAGE BYTES PER VALUE	51 BYTES.
SIZE OF ALL SPARES	16186 BYTES.
SIZE OF UNUSEABLE BYTES	27696 BYTES.
SIZE OF CONCORDANCE TABLES	24 PERCENT.
SIZE OF DATA TABLES	18 PERCENT.
SIZE OF UNIQUE NAME VALUES	15 PERCENT.
PERCENT OF SPARES	22 PERCENT.
DATA EXPANSION RATIO	13.28

EXHIBIT 3

THE FILE DESCRIPTION OF THE ON-LINE CIRCULATION FILE

```
DESCRIBE
----
     C1   CALL NUMBER (NAME)
     C3   AUTHOR (NAME)
     C4   TITLE (NAME)
     C5   TOTAL TIMES BORROWED (NUMBER)
     C2   COPY ID (REPEATING GROUP)
    C21       COPY NUMBER (NUM IN 2) FORMAT IS 00
    C22       BORROWER ID (NUM IN 2)
    C23       ORG. CODE (NUM IN 2) FORMAT IS 000000
    C24       DUE DATE (DATE IN 2)
    C25       STATUS (NAME IN 2) VALUES ARE C,D,H,R,T
    C26       LOCATION (NAME IN 2) VALUES ARE B,M,O
    C27       TIMES RENEWED (NUM IN 2) FORMAT IS 000
    C28       BORROWER NAME (NAME IN 2)
    C29       BORROWER ROOM (NUM IN 2)
----
```

EXHIBIT 5

COMMAND TO PRODUCE COUNT OF ITEMS CHARGED
FROM MAIN LIBRARY ON JAN. 9, 1970

```
  PRINT COUNT ENTRIES WHERE C26 EQ M AND C24 EQ 01/30/70 AND *
*C25 EQ C
----

COUNT ENTRIES = 13

----
```

EXHIBIT 4

COMMAND TO PRODUCE LISTING OF CALL NUMBERS OF ITEMS
CHARGED TO EMPLOYEE #4048

```
NEXT
PRINT C1 ,\WHERE C22 EQ 4048
----

         C1

T1755 A43

Q180 A1G92

HD2721 B63

HG8053 A511

HD205 C62

T1755 D28
```

and stored so that they could be called within the COMPOSE program to produce the desired reports. Exhibits 6 through 18 show several such COMPOSE reports. Once the report structure is composed, the process of getting a specific output is very easy; for example, an overdue program was created and stored under the name CIRCRPT1 that would produce first notices for overdues whenever the librarian chose to run it. When the COMPOSE program asks for the report name, the user types in CIRCRPT1 and identifies the disk pack on which the program is stored. When COMPOSE has found this report program, it responds by saying NEXT. At this point, the user can either make changes to the report program, ask for a review, or a proof of what the report format will look like, or he can give the command RUN, whereupon the program proceeds to produce the report.

Three overdue-notice programs were created--for the first notice, the second, and the third. Each has a slightly different output, but they all do basically the same thing: produce a report, organized by borrower ID number (which is an employee number at SDC), print the call number and (had there been any) the author and title, the due date, and a message asking the borrower either to return or renew the books. The language of the second and third notices differs slightly from that of the first.

There is another circulation report that is also run, for third notice recipients, that produces a listing for the librarian arranged by employee number, giving the call number and the number of items overdue to each borrower. Had we had a "live" system instead of a simulated system, we would have added borrower telephone extension numbers and room numbers to the ID number as part of the file. In that case, these additional elements could have been printed on the report for the librarian so that he would not even have had to look up phone numbers in order to call the borrower and ask for the return of the books. This report program also produces a sum total of the items overdue.

Other report programs were created; one, for example, could produce a listing showing the computer file number of each loan record along with the call number. This could be useful for updating a record when the loan status is changed from a temporary to a permanent loan, called "office reference" at SDC and coded as an "O". A sample is shown of such a listing in Exhibit 12.

Other reports of interest to the librarian might have included one showing the number of items borrowed by each employee, arranged in employee-number order, and the number of items borrowed by each department within the company. These statistics would tell the library manager who his most active customers are, both by individual and by organization. Corporate management could use this type of information to determine optimum library location. The company has a number of buildings, which are somewhat scattered. Naturally, there is always a problem of space, and the location of a library might better be

EXHIBIT 6

REPORT SPECIFICATION AND PROOF OF PROGRAM TO
PRODUCE FIRST OVERDUE NOTICE

```
REVIEW
      DATA BASE NAME IS SDC LIBRARY CIRCULATION FILE
      QUALIFY DUE DATE  WHERE DUE DATE LS 'K CUTOFFDATE
      SORT BY BORROWER ID
T1    T. LIBRARY OVERDUE NOTICE
      FEED 2 AFTER
H2    H. <THE MATERIALS LISTED BELOW ARE OVERDUE.  PLEASE RETURN THEM>
      FEED 1 AFTER
H3    H. <AT ONCE, OR, IF YOU WISH TO RENEW THEM, CALL EXT. 7232>
      FEED 2 AFTER
H4    H. BORROWER, ORGANIZATION CODE, DUE DATE
      SPACE TO 20, 56 BEFORE F2, F3
      FEED 1 AFTER
C1    C. BORROWER ID, ORG. CODE
      SPACE TO 24 BEFORE F2
      FEED 2 AFTER
C3    C. 'L <CALL #>, 'L COPY
      SPACE TO 18 BEFORE F2
C2    C. CALL NUMBER, COPY NUMBER, AUTHOR, TITLE, DUE DATE
      MASK C2F3 = XXXXXXXXXX
      MASK C2F4 = XXXXXXXXXXXXXXX
R1    R. BORROWER ID = 'L <NUMBER ITEMS OVERDUE>, COUNT CALL NUMBER
      MASK R1F2 = 000
      FEED PAGE AFTER
      PUT TOP T1
NEXT
PROOF

      ....5....0....5....0....5....0....5....0....5....0....5....0....5

T1                          LIBRARY OVERDUE NOTICE

H2    THE MATERIALS LISTED BELOW ARE OVERDUE.  PLEASE RETURN THEM
H3    AT ONCE, OR, IF YOU WISH TO RENEW THEM, CALL EXT. 7232

H4    BORROWER            ORGANIZATION CODE                    DUE DATE


C1    XXXXX                     000000
C3    CALL #              COPY
C2    XXXXXXXXXXXXXX     00     XXXXXXXXXX     XXXXXXXXXXXXXXX     99/99/99

R1    NUMBER ITEMS OVERDUE    000

      ....5....0....5....0....5....0....5....0....5....0....5....0....5
```

EXHIBIT 7

OVERDUE NOTICES (FIRST) PRODUCED BY REPORT SPECIFICATION ABOVE IN EXHIBIT 6

LIBRARY OVERDUE NOTICE

THE MATERIALS LISTED BELOW ARE OVERDUE. PLEASE RETURN THEM AT ONCE, OR, IF YOU WISH TO RENEW THEM, CALL EXT. 7232

BORROWER	ORGANIZATION CODE	DUE DATE
9070	6221	

CALL #	COPY	
HF5548 B72 196	1	01/20/69

NUMBER ITEMS OVERDUE 1

LIBRARY OVERDUE NOTICE

THE MATERIALS LISTED BELOW ARE OVERDUE. PLEASE RETURN THEM AT ONCE, OR, IF YOU WISH TO RENEW THEM, CALL EXT. 7232

BORROWER	ORGANIZATION CODE	DUE DATE
13138	6321	

CALL #	COPY	
JA74 E13	2	01/07/69

NUMBER ITEMS OVERDUE 1

EXHIBIT 8

REPORT DESCRIPTION AND PROOF FOR SECOND OVERDUE NOTICE

```
REVIEW
      DATA BASE NAME IS SDC LIBRARY CIRCULATION FILE
      QUALIFY DUE DATE  WHERE DUE DATE LS 'K CUTOFFDATE
      SORT BY BORROWER ID
T1    T. <LIBRARY OVERDUE NOTICE*******SECOND NOTICE*******>
      FEED 1 AFTER
T2    T. THIS IS THE SECOND OVERDUE NOTICE
      FEED 2 AFTER
H2    H. <THE MATERIALS LISTED BELOW ARE OVERDUE.  PLEASE RETURN THEM>
      FEED 1 AFTER
H3    H. <AT ONCE, OR, IF YOU WISH TO RENEW THEM, CALL EXT. 7232>
      FEED 2 AFTER
H4    H. BORROWER, ORGANIZATION CODE, DUE DATE
      SPACE TO 20, 56 BEFORE F2, F3
      FEED 1 AFTER
C1    C. BORROWER ID, ORG. CODE
      SPACE TO 24 BEFORE F2
      FEED 2 AFTER
C3    C. 'L <CALL #>, 'L COPY
      SPACE TO 18 BEFORE F2
C2    C. CALL NUMBER, COPY NUMBER, AUTHOR, TITLE, DUE DATE
      MASK C2F3 = XXXXXXXXXX
      MASK C2F4 = XXXXXXXXXXXXXXX
R1    R. BORROWER ID = 'L <NUMBER ITEMS OVERDUE>, COUNT CALL NUMBER
      MASK R1F2 = 000
      FEED PAGE AFTER
      PUT TOP T1, T2
NEXT


PROOF

      ....5....0....5....0....5....0....5....0....5....0....5....0....5

T1          LIBRARY OVERDUE NOTICE*******SECOND NOTICE*******
T2                  THIS IS THE SECOND OVERDUE NOTICE

H2    THE MATERIALS LISTED BELOW ARE OVERDUE.  PLEASE RETURN THEM
H3    AT ONCE, OR, IF YOU WISH TO RENEW THEM, CALL EXT. 7232

H4    BORROWER            ORGANIZATION CODE                    DUE DATE

C1    XXXXX                     000000
C3    CALL #            COPY
C2    XXXXXXXXXXXXXX     00     XXXXXXXXXX     XXXXXXXXXXXXXXX     99/99/99

R1    NUMBER ITEMS OVERDUE     000

      ....5....0....5....0....5....0....5....0....5....0....5....0....5
```

EXHIBIT 9

SECOND (OVERDUE) NOTICES PRODUCED BY
REPORT DESCRIPTION ABOVE IN EXHIBIT 8

LIBRARY OVERDUE NOTICE*******SECOND NOTICE*******
THIS IS THE SECOND OVERDUE NOTICE

THE MATERIALS LISTED BELOW ARE OVERDUE. PLEASE RETURN THEM AT ONCE, OR, IF YOU WISH TO RENEW THEM, CALL EXT. 7232

BORROWER	ORGANIZATION CODE	DUE DATE
9070	6221	

CALL #	COPY	
HF5548 B72 196	1	01/20/69

NUMBER ITEMS OVERDUE 1

LIBRARY OVERDUE NOTICE*******SECOND NOTICE*******
THIS IS THE SECOND OVERDUE NOTICE

THE MATERIALS LISTED BELOW ARE OVERDUE. PLEASE RETURN THEM AT ONCE, OR, IF YOU WISH TO RENEW THEM, CALL EXT. 7232

BORROWER	ORGANIZATION CODE	DUE DATE
13138	6321	

CALL #	COPY	
JA74 E13	2	01/07/69

NUMBER ITEMS OVERDUE 1

EXHIBIT 10

REPORT DESCRIPTION AND PROOF FOR THIRD OVERDUE NOTICE

```
REVIEW
      DATA BASE NAME IS SDC LIBRARY CIRCULATION FILE
      QUALIFY DUE DATE  WHERE DUE DATE LS 'K CUTOFFDATE
      SORT BY BORROWER ID
T1    T. <***LIBRARY OVERDUE NOTICE*******THIRD NOTICE*******>
T2    T. <THIS IS THE THIRD NOTICE!  A COPY OF THIS NOTICE IS BEING
      SENT>
      FEED 1 AFTER
T3    T. <TO YOUR DEPARTMENT HEAD.>
      SPACE TO 2 BEFORE F1
      FEED 2 AFTER
H2    H. <THE MATERIALS LISTED BELOW ARE OVERDUE.  PLEASE RETURN THEM>
      FEED 1 AFTER
H3    H. <AT ONCE, OR, IF YOU WISH TO RENEW THEM, CALL EXT. 7232>
      FEED 2 AFTER
H4    H. BORROWER, ORGANIZATION CODE, DUE DATE
      SPACE TO 20, 56 BEFORE F2, F3
      FEED 1 AFTER
C1    C. BORROWER ID, ORG. CODE
      SPACE TO 24 BEFORE F2
      FEED 2 AFTER
C3    C. 'L <CALL #>, 'L COPY
      SPACE TO 18 BEFORE F2
C2    C. CALL NUMBER, COPY NUMBER, AUTHOR, TITLE, DUE DATE
      MASK C2F3 = XXXXXXXXXX
      MASK C2F4 = XXXXXXXXXXXXXXX
R1    R. BORROWER ID = 'L <NUMBER ITEMS OVERDUE>, COUNT CALL NUMBER
      MASK R1F2 = 000
      FEED PAGE AFTER
      PUT TOP T1, T2, T3
NEXT
PROOF

      ....5....0....5....0....5....0....5....0....5....0....5....0....5
```

```
T1          ***LIBRARY OVERDUE NOTICE*******THIRD NOTICE*******
T2     THIS IS THE THIRD NOTICE!  A COPY OF THIS NOTICE IS BEING SENT
T3     TO YOUR DEPARTMENT HEAD.

H2    THE MATERIALS LISTED BELOW ARE OVERDUE.  PLEASE RETURN THEM
H3    AT ONCE, OR, IF YOU WISH TO RENEW THEM, CALL EXT. 7232

H4    BORROWER            ORGANIZATION CODE                    DUE DATE


C1    XXXXX                     000000
C3    CALL #             COPY
C2    XXXXXXXXXXXXXX      00     XXXXXXXXXX     XXXXXXXXXXXXXXX     99/99/99

R1    NUMBER ITEMS OVERDUE     000

      ....5....0....5....0....5....0....5....0....5....0....5....0....5
```

EXHIBIT 11

OVERDUE (THIRD) NOTICE PRODUCED BY
REPORT DESCRIPTION ABOVE IN EXHIBIT 10

LIBRARY OVERDUE NOTICE****THIRD NOTICE*******
THIS IS THE THIRD NOTICE! A COPY OF THIS NOTICE IS BEING SENT TO YOUR DEPARTMENT HEAD.

THE MATERIALS LISTED BELOW ARE OVERDUE. PLEASE RETURN THEM AT ONCE, OR, IF YOU WISH TO RENEW THEM, CALL EXT. 7232

BORROWER	ORGANIZATION CODE	DUE DATE
9070	6221	

CALL #	COPY	
HF5548 B72 196	1	01/20/69

NUMBER ITEMS OVERDUE 1

LIBRARY OVERDUE NOTICE****THIRD NOTICE*******
THIS IS THE THIRD NOTICE! A COPY OF THIS NOTICE IS BEING SENT TO YOUR DEPARTMENT HEAD.

THE MATERIALS LISTED BELOW ARE OVERDUE. PLEASE RETURN THEM AT ONCE, OR, IF YOU WISH TO RENEW THEM, CALL EXT. 7232

BORROWER	ORGANIZATION CODE	DUE DATE
13138	6321	

CALL #	COPY	
JA74 E13	2	01/07/69

NUMBER ITEMS OVERDUE 1

EXHIBIT 12

SAMPLE LISTING OF FILE INDEX NUMBER
(COMPUTER ASSIGNED) WITH ASSOCIATED CALL NUMBER

I = 14 C1 = QA765 B26

I = 16 C1 = QA76 B27

I = 18 C1 = QA76 B31

I = 20 C1 = QA765 B32

I = 22 C1 = QA765 B42

I = 24 C1 = QA76 B64

I = 26 C1 = QA76 A83 1967

I = 28 C1 = HC106 B688

I = 30 C1 = HC110 A4W86

I = 32 C1 = HM131 B43

I = 34 C1 = T1755 A43

I = 36 C1 = Q180 A1H12

I = 38 C1 = Q180 A1S67 1962

I = 40 C1 = Q180 A1G92

I = 42 C1 = Q180 A1G92

EXHIBIT 13

REPORT SPECIFICATION AND PROOF FOR LISTING OF BORROWERS RECEIVING THEIR THIRD OVERDUE NOTICE

```
REVIEW
      DATA BASE NAME IS SDC LIBRARY CIRCULATION FILE
      QUALIFY DUE DATE  WHERE DUE DATE LS 'K CUTOFFDATE
      SORT BY ORG. CODE, BORROWER ID
T1    T. <OVERDUE NOTICE REPORT: BORROWERS RECEIVING THIRD NOTICES>
H1    H. ORG. CODE, BORROWER ID, CALL* NUMBER, COPY* NUMBER, DUE DATE
C1    C. ORG. CODE, BORROWER ID, CALL NUMBER, COPY NUMBER, DUE DATE
R1    R. BORROWER ID = 'L <NUMBER OVERDUES>, COUNT CALL NUMBER
      MASK R1F2 = 000
R2    R. OVERALL =  'L <TOTAL NUMBER ITEMS OVERDUE>, COUNT CALL NUMBER
      MASK R2F2 = 0000
      LINK H1, C1
NEXT
PROOF

      ....5....0....5....0....5....0....5....0....5....0....5....0....

T1       OVERDUE NOTICE REPORT: BORROWERS RECEIVING THIRD NOTICES

H1    ORG. CODE     BORROWER ID     CALL                COPY      DUE DATE
                                    NUMBER              NUMBER

C1       000000     XXXXXXXXXXX     XXXXXXXXXXXXXXX        00     99/99/99

R1    NUMBER OVERDUES     000
R2    TOTAL NUMBER ITEMS OVERDUE     0000

      ....5....0....5....0....5....0....5....0....5....0....5....0....
```

EXHIBIT 14

REPORT SHOWING PERTINENT INFORMATION REGARDING EMPLOYEES AND LOANS FOR WHICH THEY ARE RECEIVING THEIR THIRD OVERDUE NOTICE.
THIS REPORT IS PRODUCED BY THE SPECIFICATIONS GIVEN ABOVE IN EXHIBIT 13.

OVERDUE NOTICE REPORT: BORROWERS RECEIVING THIRD NOTICES

ORG. CODE	BORROWER ID	CALL NUMBER	COPY NUMBER	DUE DATE
3571	10644	HD1694 A5H88	3	08/07/69
3571	10644	HD6331 B34	2	05/12/69
NUMBER OVERDUES	2			
3581	475	TK3226 F6	7	09/08/69
NUMBER OVERDUES	1			
4416	13976	UA23 H53	12	08/28/69
NUMBER OVERDUES	1			
6221	9070	HF5548 B72 196	1	01/20/69
NUMBER OVERDUES	1			
6321	2791	QA76 A83 1967	2	07/21/69
NUMBER OVERDUES	1			
6321	13138	JA74 E13	2	01/07/69
NUMBER OVERDUES	1			
8332	13809	QH3085 B6 V1	2	06/11/69
8332	13809	HM73 D4 1962	2	06/04/69
8332	13809	PK7 F69	1	06/30/69
8332	13809	PK7 F69	1	06/30/69
NUMBER OVERDUES	4			
TOTAL NUMBER ITEMS OVERDUE	11			

EXHIBIT 15

REPORT SPECIFICATION AND PROOF TO PRODUCE LISTING OF LOANS
BY EMPLOYEES FOR PERIOD DEC. 5, 1969 TO JAN. 11, 1970

```
NEXT
REVIEW
      DATA BASE NAME IS SDC LIBRARY CIRCULATION FILE
      QUALIFY BORROWER ID  WHERE C24 GR 12/26/69 AND C24 LS 02/01/70
      SORT BY BORROWER ID
      DERIVE  DUMMY =   COUNT C1  WHERE C22 EXISTS
T1    T. <TOTAL LOANS BY BORROWER>
H1    H. BORROWER ID, <TOTAL ITEMS BORROWED>
R1    R. BORROWER ID = BORROWER ID, DUMMY
      MASK R1F2 = 000
      SPACE TO 1 BEFORE F2
R2    R. OVERALL =  'L <TOTAL LOANS>, DUMMY
      MASK R2F2 = 00000
NEXT
PROOF

      ....5....0....5....0....5....0....5

T1          TOTAL LOANS BY BORROWER

H1    BORROWER ID     TOTAL ITEMS BORROWED

R1    99999                 000
R2    TOTAL LOANS         00000

      ....5....0....5....0....5....0....5
```

EXHIBIT 16

REPORT PRODUCED BY SPECIFICATION IN EXHIBIT 15

TOTAL LOANS BY BORROWER

BORROWER ID	TOTAL ITEMS BORROWED
316	1
639	6
820	2
1359	1
1553	1
1587	1
01992	1
2056	2
02056	1
2102	1
2212	1
02385	2
2499	1
03287	1
3465	1
03518	1
3555	5
3624	2
4234	3
04234	1
4581	1
4628	1
8230	1
08270	1
8487	1
8571	1
8738	1
8928	1
9642	1
09785	2
10076	1
10371	2
10815	1
10855	1
10959	2
11164	2
11917	3
11990	1
13053	1
13059	1
13160	2
13359	1
13419	1
13568	1
13598	2
13815	1
13869	1
14010	1
14099	1
14152	1
14262	1
14263	11
14272	1
14358	2
TOTAL LOANS	88

EXHIBIT 17

PROOF AND REVIEW STATEMENT OF REPORT TO PRODUCE
LISTING OF LOANS BY DEPARTMENT (ORGANIZATION CODE)
FOR THE PERIOD DEC. 5, 1969 TO JAN. 11, 1970

```
PROOF

     ....5....0....5....0....5....0...

T1   TOTAL LOANS BY ORGANIZATION CODE

H1   ORG. CODE     TOTAL ITEMS BORROWED

R1   000000              0000
R2   TOTAL LOANS      00000

     ....5....0....5....0....5....0...

NEXT
REVIEW
     DATA BASE NAME IS SDC LIBRARY CIRCULATION FILE
     QUALIFY ORG. CODE  WHERE C24 GR 12/26/69 AND C24 LS 02/01/70
     SORT BY ORG. CODE
     DERIVE  DUMMY =    COUNT C1  WHERE C23 EXISTS
T1   T. <TOTAL LOANS BY ORGANIZATION CODE>
H1   H. ORG. CODE, <TOTAL ITEMS BORROWED>
R1   R. ORG. CODE = ORG. CODE, DUMMY
     MASK R1F2 = 0000
     SPACE TO 18 BEFORE F2
R2   R. OVERALL =   'L <TOTAL LOANS>, DUMMY
     MASK R2F2 = 00000
```

EXHIBIT 18

REPORT PRODUCED BY REPORT DESCRIPTION OF EXHIBIT 17, SHOWING TOTAL LOANS FOR EACH DEPARTMENT OF THE COMPANY DURING THE PERIOD 12/05/69 TO 01/11/70 (I.E., THREE WEEKS PRIOR TO DATES USED IN REPORT DESCRIPTION)

TOTAL LOANS BY ORGANIZATION CODE

ORG. CODE	TOTAL ITEMS BORROWED
1521	1
1739	1
2511	1
3834	1
4221	1
4333	1
4414	3
4421	11
4422	2
4424	5
4612	1
4616	1
4621	8
4772	5
4774	1
4782	2
4811	1
4841	1
4881	1
5345	1
6111	1
6112	2
6114	2
6221	7
6321	4
6751	1
7410	3
8040	1
8101	1
8322	5
8323	4
8342	1
8351	1
8431	1
8432	3
8461	1
9833	1
TOTAL LOANS	88

close to its heaviest users than simply where space is most readily available. At one time this was a concern of management, but in order to gather such data it was necessary to make a survey of all employees. Had our LISTS on-line circulation system been available, it would have been a matter of only minutes to have produced a listing showing the number of loans made to an individual or department.

Reports could have been generated, using some of the statistical operators such as MAXIMUM, which could have identified the borrower having the most items on loan. Such a report would have been very simple, requiring the addition of about one line of instruction in the report from Exhibit 15. Similarly, the program that produced the count of loans for a particular period of time could have been used to produce a count of the number of loans made over any period simply by changing the date used in the QUALIFY statement (see Exhibit 17).

There is no question that the CDMS set of programs could have managed the library circulation program very well, indeed, in terms of its flexibility, the products that it can produce, and the control it would have allowed the librarian to maintain. What is not clear is whether or not the program would have been economical, in the long run, to use.

Costs

The program SIZES, which calculates the size of any particular file in CDMS, was used to generate the listing shown in Exhibit 2. The total amount of storage taken for the 180 records was 73440 bytes.

It appears that a reasonably good typist, after an initial learning period, familiar with the file structure and the data itself, could have input records for new items at about one record per minute when the total information called for in the file structure was to be entered. One element of information we were not able to incorporate into the simulation was renewal transactions that the library normally encountered. These would have required far less input time, since, for renewals, only two elements need to be entered: (1) the new due date, and (2) a one added to the loan counter, showing how many times the item has been renewed.

Using the figures available to us for the last five months of operation of the old SDC Library circulation system (July through November 1969) the average number of loans per month for both of the libraries (Main and Branch) was 404, and the renewals averaged 195 per month. Using an average of 20 working days per month, the daily number of original loans would have been 20.2. The average daily renewals would have been 9.75. If we use the input rate of one record per minute for new loans, and a very conservative estimate of three records per minute for renewals, plus 10 minutes of overhead time

to log into the system and to get the programs running, we have (in round figures) 34 minutes at the terminal per day for input. The precise costs for this use of the computer would depend upon the current rate structure of the time-sharing system. At one time during the course of the project, the rates were based on a rather complex set of costs that included the terminal connect time at $9.00 per hour, the central processing unit (CPU) time at $12.00 per minute, $9.00 per hour for access to the first logical cylinder of a disk pack, and $.0432 per hour for each logical cylinder beyond the first. There is no way that we can calculate what a real circulation file would have cost, since we have no idea what records would have gone into the file. The actual space taken by the file would depend upon variables such as the extent of duplication of names in the indexes set up by CDMS. If a borrower has 20 books out under his name, or man number, the man number need be repeated in the CDMS indexes only once; however, 20 borrowers each having one book out would require 20 entries in the index table for their man numbers. That is, without knowing exactly the make-up of the file, it is not possible to calculate the space necessary, and there is no way to predict this from the sample we had. However, it is known that with this particular set of programs (i.e., CDMS) and with a disk pack half filled (which would have been, perhaps, the maximum the SDC library file would reach) the time-sharing costs might have been as much as $80.00 or $90.00 per hour. However, beginning on April 1, 1970, a new price structure went into effect that would have allowed a user to use a system on an hourly basis at $38.00 per hour without regard to the central processing unit time or file size or terminal connect time. On that basis the half hour input per day would have cost $19.00. Even at that rate the costs seem rather high to carry for a long period of time. It would appear, therefore, that on-line input is <u>not</u> cost effective unless the charges are less than $10.00 per hour.

It is known in the Los Angeles area (and presumably elsewhere) that there is some time-sharing service available that offers on-line text editing at about $5.00 per hour, provided the file remains very small. It is conceivable that such services might be used for input and the data transferred then into the other computer system to do the generation and updating of files, etc.

Another alternative to reduce input costs could be to keyboard the material on an off-line device such as a typewriter with an optically scanable type face, or the old reliable keypunch. It is estimated that in this way input costs could be kept quite low, since prestoring of punched cards on the SDC time-sharing system can be accomplished at a charge of $30.00 an hour. Since the card reader will handle approximately 1,000 cards a minute, it is easy to see that less than a minute is necessary to handle the day's input. (The cost to the nearest minute, exclusive of the manpower to operate the keypunch, would only be .50¢ for the typical daily load at SDC.) Someone would still have to run subsequent programs to add these prestored cards into the circulation file, but this probably could be accomplished in about 5 minutes of

terminal time, on the average. Thus, it is conceivable that, even at a charge of $38.00 per hour for time-sharing, the overall input costs could be held under $4.00 per day.

QUERY and COMPOSE would produce the various on-line reports. QUERY is used to obtain daily statistics of new loans or renewals, or to obtain a listing of materials on loan to any given borrower. Such queries could be accomplished in less than a minute, once the program is in operation and the disk pack is mounted. Switching from one program to another takes approximately 20 seconds in the system; had the daily circulation statistics been obtained immediately after finishing daily update, there would have been no additional overhead time to account for, other than the 20 seconds to switch to QUERY from GENERATE. It seems fair to cost this at slightly under two minutes--about $1.00. Since the same process is used, or could be used, to secure statistics for longer periods of time, there is no additional cost to prepare weekly, monthly, semiannual, etc., statistics over and above what is necessary for the daily operation. That is, to prepare a monthly report of the total number of loans would take no more time than to get the daily statistics. It would cost about $1.00 additional per month to obtain the monthly statistics.

COMPOSE, which produces other types of statistical reports, takes somewhat longer and depends upon the total load of the system to a much greater extent than does QUERY. This is due primarily to the fact that the COMPOSE operation can involve such processes as sorting on more than one field to prepare the desired report, and sorting is accorded a lower priority in the time-sharing system. This means that the terminal must be connected for a longer time and the process will, therefore, cost proportionally more.

The precise figures for the experimental file are as follows: Based on a file size of 180 records, and average time for the various reports and notices as derived during the simulation, we can calculate that every 10 items on an overdue report will take about 3 minutes. At a cost of .64¢ per minute this would equal $1.92. This seems rather expensive for the management reports such as those in Exhibits 14, 16, and 18. But for overdue notices this is not quite as expensive, particularly if the off-line mode of printing were used rather than printing the overdue notices on the terminal. This was not tested during the simulation, but using data derived from other sources we can calculate that the process would be roughly 10 times faster. It would then cost about .20¢ per overdue notice, a price that might be tolerable.

The major advantage of CDMS appears to be the speed and ease with which a system can be established and tested, rather than its overall cost benefits.

LISTS SYSTEM
USER'S MANUAL

Table of Contents

Appendices

Chapter 1 - Introduction

A. General Information

This manual contains instructions for using the LISTS (Library Information System Time-Sharing) System by means of a remote keyboard terminal connected to the time-shared IBM 360/67 computer at System Development Corporation in Santa Monica, California. Each chapter of this manual contains instructions for a particular function that is a part of the overall LISTS System. Chapters for functions not yet implemented will be added from time to time as the functions become ready for operation, and existing chapters may be modified to reflect changes in system operation.

Appendix 1 contains two lists: one showing the legal console commands for the time-sharing system and the other showing specific instructions for your library. This instruction set includes legal commands to the time-sharing executive program (TSS) of the computer, as well as the various commands appropriate to the LISTS programs. We suggest that this instruction set be placed near your terminal, so that library personnel may have it available for reference while operating the terminal.

Every person using the LISTS system should be thoroughly familiar with this manual before beginning operation. Especially review the chapters that discuss particular processes or functions that you intend to use before you use them for the first time. From time to time, special instructions may be necessary because of changes in the operation of the executive programs or for other reasons; these will be issued in the form of bulletins as rapidly as possible after they become effective. These bulletins should be kept with the summary instruction sheets, and persons using the system should always check to see whether or not a new bulletin has been issued that might affect the operation they intend to perform.

B. General Procedures

The procedures for operating LISTS and the time-sharing system (TSS) are termed interactive. That is, you retrieve information from LISTS much as you do by statement and response in a dialogue with another person. LISTS, of course, is not a person, but a computer program system. Each message you type on your terminal elicits a response from LISTS or the TSS, depending on which you address. In addition to the substantive information the system may request, in certain operations you will receive other responses such as simple acknowledgment of a command, a query for more information, an instruction to carry out your next message, or even an unsolicited message from a computer operator in Santa Monica or the SDC LISTS monitor. You may also receive error messages if you fail to phrase messages in terms that LISTS can understand. The TSS will always tell you when you have used illegal terms. Usually, it will indicate the nature of the illegality. If you are in doubt, your best recourse is to call the SDC LISTS monitor for help.

C. Message Rules

You can talk to TSS or to LISTS by using the proper addressing symbols, the correct command, and the required terminal action. If you are addressing TSS, you must precede every line with a virgule, or slash (/); no special symbol is used to address LISTS. You must end every message (or command) by pressing the carriage return key ((CR)). If a line of command exceeds one line, then you must type in a special symbol (*) at the end of the line, before you use the carriage return key. This will indicate to the LISTS program that the line has not ended.

D. How to Dial the System

1. Pick up the hand set on the data phone.
2. When you hear the dial tone, dial this number: 451-8731* (in Los Angeles)
3. Wait for a "tone" from the computer; when you hear the "tone" press ORIG button.
4. You are now ready to communicate with TSS.

E. TSS Commands and Their Use

Two TSS commands, LOGIN, and the name of the LISTS program you want to use, are required in order to prepare LISTS for operation.

1. LOGIN

TSS requires each user to identify himself before it will allow LISTS to be loaded and operated. To identify yourself, type /LOGIN XXX CR. (See

*If your terminal is not a teletype you cannot use this number.

Condensed Instruction Sheet for your log number.) The system responds to the LOGIN command with a message like this: LOGGED IN 10:22 5/12/69.

2. LOAD and GO

When TSS asknowledges your identification, type XXX.* TSS responds with this message: PGM STARTED. It is possible that the program may not be immediately available, in which case you may receive the following message: PROGRAM NOT CATALOGED. TSS is telling you that the program must be loaded from tape. If this occurs, call the SDC LISTS monitor.

After TSS responds with the message PGM. STARTED, you will receive instructions as to how to continue. See the appropriate chapters for the various functions you wish to perform. Sample printouts of actual operations are included in Appendix 2, to which you may wish to refer. (These are sample printouts only, you should not attempt to use the codes and responses indicated in the samples, since they may not necessarily be appropriate for your library.)

Appendix 1 contains additional TSS commands that may be of use to you.

F. How to Redial When Disconnected

If you are accidentally disconnected from the computer during LISTS operation, you can recover your lost channel and resume operation where you were in LISTS prior to disconnect. The list of telephone numbers below shows the channel with which each number is associated. The first response from the computer system immediately after you LOGIN indicates the channel to which you have connected. Lock at the beginning of your input record to see what channel you were using. Dial the telephone number associated with that channel for reconnection. As soon as you have made connection, if you do not receive a message from the time-sharing system indicating the channel to which you are connected, press the carriage return key on your terminal. You should receive a message indicating the channel to which you are connected or you may receive a message to RELOG or REDIAL. To relog, type in /RELOG xxxx xxx with your log number. If the channel number does not match the previous channel number that you were using, immediately redial the proper number. If you are not successful in reestablishing connection with the proper channel, call your SDC LISTS monitor.

---

* The X's indicate the program name. See Condensed Instruction Sheet for available programs.

Telephone and Channels for Teletype Model 33 or 35

Channel 070	#213-828-9711
071	213-828-9712
072	213-828-9713
073	213-828-9714
074	213-828-9715
075	213-828-9716
076	213-828-9717
077	213-828-9718
078	213-828-9719
079	213-828-9710
07A	213-828-9721
07B	213-828-9722
07C	213-828-9723

G. How to Terminate LISTS Operation

When you wish to end program operation, type /QUIT XXX.* You should receive the message PGM. QUIT from TSS. At this point, if you are not going to use your terminal any longer, you should log out by typing /LOGOUT. The system will respond with a message LOGGED OUT and indicate the time and date.

H. Cancellation of Errors

A single character can be cancelled on a teletype terminal by using the back slash character (\) (uppercase L). A reasonable number of characters can be cancelled by this method; however, be very careful that you count the number of characters cancelled. You may cancel an entire line by pressing the "BREAK" key.

I. Problems

At any time in the sequence, you may encounter certain problems. If, in logging in, the computer responds NO LOGIN, PARAMETER ERROR, try to log in again. If the error continues, check with the Monitor. If, in loading a program, you get no response for several minutes or receive certain types of error messages, disconnect, recall the computer, and relog. If the problem persists, call the monitor at SDC. If you do not receive a line feed after a carriage return, the system is probably down. See the list of error messages in Appendix 1.

---

* The X's indicate the program name. See Condensed Instruction Sheet for available programs.

Chapter 2 - Retrieval

A. Introduction

This chapter describes the use of the program BIRCH (derived from Bibliographic search) that allows a user to search Library of Congress MARC data. In the future, additional files of bibliographic data will be available (see Note 1 at the end of this chapter) for searching. Each file will have a unique name. (BIRCH will operate in an identical manner on any of the files that will be added to the system at a later date. The number of elements and their names may change for these different files, but the method of operation will be the same.) The MARC file is called BIBLIO. In the following pages you will find a sample operating session with BIRCH and BIBLIO. At the end of the chapter there is the record of an actual session operating BIRCH. With these guides you should be able to use this part of LISTS.

B. BIRCH Details

The BIRCH program has three modes of operation: long, short, and symbolic. Each message of the program has three forms matching the above three modes. As you use the program, you will want to shorten the computer response as you gain familiarity with the system.

You may change from one mode to another at any time the system is waiting for a response from you. Such occasions are announced by the printing of the word USER followed by a colon. To change forms, you type the word "LONG" or "SHORT" or "SYMBOLIC" (enclosed in quotes) and hit the Carriage Return key. Inputting this command will change the form of the last message you received. If you wish to change the form of all messages, then use the word ALL after the three versions commands-- LONG, SHORT or SYMBOLIC as follows: "LONG ALL", "SHORT ALL", or "SYMBOLIC ALL". At that point, the computer will restate its last response in the new form you have requested, long, short, or symbolic, as the case may be.

BIRCH offers a number of methods of retrieval. These are indicated in Figure 1 which shows the name and number of each element in the BIBLIO file (page B-7). The first retrieval element is the LC card number. To use it as a retrieval element, it is input with the hyphen that one normally sees on a printed Library of Congress catalog card.

The LC class number is a retrieval element. In some instances, it may include as many as eight characters. There is also a method to retrieve on a portion of the class number. This is discussed below in the section dealing with the "truncation" character. There is also a "neighbor" command which allows you to scan the index to see what particular class numbers actually exist. This technique is discussed in the section on Neighbor, below.

ELEMENT NUMBER	ELEMENT NAME	LENGTH	RETRIEVAL ELEMENT	QUALIFYING ELEMENT
01	LC CARD NUMBER	12	X	X
02	MARC DATE	6		X
03	LC CLASS NO.	8	X	X
04	DEWEY CLASS NO.	3	X	X
05	LC PRICE	20		
06	EDITION	20		
07	CONFERENCE INDICATOR	1	X	X
08	AUTHOR-CONFERENCE	64	X	X
09	SHORT TITLE	63		
10	TITLE	189		
11	SERIES NOTE	126		
12	PLACE OF PUBLICATION	20		
13	PUBLICATION DATE	4	X	X
14	LEVEL	1	X	X
15	PUBLISHER	63		
16	FORM OF CONTENTS	4		

ONLY DESIGNATED RETRIEVAL ELEMENTS MAY BE USED FOR RETRIEVAL.
ONLY DESIGNATED QUALIFYING ELEMENTS MAY BE USED FOR QUALIFYING
PLACE AN * BEFORE A MUST RETRIEVAL TERM.

QUALIFYING MAY BE DONE ON ELEMENT 02 BY INDICATING A RANGE
IN WEEKS. FOR EXAMPLE, 00-05 OR 04-04.

END LIST OF INPUT VALUES FOR RETRIEVAL OR QUALIFYING WITH **

ENTER A RETRIEVAL ELEMENT NUMBER, AN ELEMENT NAME, OR "MIXED".

Figure 1. The MARC Database

The Dewey decimal class is a retrieval element. In this program, the decimal class is defined as the digits before the decimal point, or the interger portion of the Dewey number. A sequence of three digits input will retrieve any item in the file that carries those numbers in its Dewey class number.

Certain indicators have been added to the MARC record by the Library of Congress. For example, there is a conference indicator, a fiction tag, a juvenile indicator and certain tags for form of contents. Festschrifts are also indicated in these codes. Please see Appendix 3 for a complete description of the various codes used. To retrieve on these codes, all that is necessary is to type the particular code in which you are interested. If it is ambiguous, you will receive a message from the program indicating the ambiguity and asking you to specify which meaning you intend.

An author, either personal, corporate or conference, is a retrieval element. In the original MARC record these three types of authors are indicated separately; we have combined them into one element. You may input up to 36 characters of an author's name. Characters beyond that point are not looked at by the retrieval program, but the actual printed record is 64 characters. We believe that this is long enough for most purposes.

Publication date is also a retrieval element. There are, however, two forms of date. One is the element Publication Date (PD). In this field, dates are carried as four digits plus a possible code. This is explained in detail in Appendix 3. The four digits are the typical form of the year, such as 1969. However, it is not possible to retrieve on 1969 in this field, because the number of items posted to the date 1969 exceeds the capacity of the program to handle them (i.e., in excess of 10,000). Thus, a second element is provided called year of publication (YR).* This element contains six digits. The first four are the "LC Date" as described in the MARC handbook. It indicates the date the material was prepared at the Library of Congress and gives some indication of the currency of the cataloging record. However, you may not retrieve on that date. You can, however, use the final two digits of a year to retrieve in conjunction with some other element. For example, suppose that you wish to retrieve works published in the period 1964 through 1968. You could use the element publication date to retrieve that information by inputting the statement 1964 or 1965 or 1966 or 1967 or 1968. You could also accomplish the same purpose using the year element (YR) in conjunction with retrieval on some other element. For example, you might phrase a question such as: U.S. and 64 + 68. This would indicate to the program that you wished all publications carrying a corporate authorship of United States (i.e., U.S.) and published during the period of 1964 through 1968. 1969 by itself, as you may recall, cannot be retrieved directly, since there are too many items posted to the 1969 file. However, you could retrieve U.S. publications published in 1969 by the following strategy: type in "U.S. AND 69+69". This would select all items having a corporate author of United States and published in 1969.

NOT IMPLEMENTED

*Not implemented in this version.

To recapitulate, the BIRCH program allows you to retrieve the following elements in the MARC file: Card Number, Library of Congress Class, Decimal Class, Author, Publication Date, and codes for Conference, Fiction, Juvenile. Elements such as price, edition statement, publisher, place of publication, series, and title cannot be used as retrieval elements, but can be printed. In printing these elements, you can specify which ones you wish printed or, if you choose, you may exclude specific elements from printing. A simple command PRINT will give you a printout in compact form with all of the elements in the record printed one after another. If you should wish to specify certain elements, all that is necessary is to include the two letter abbreviations of the various elements after the word PRINT. For example, suppose that you wish to print on-line only author and short title. The command would be "PRINT AU,ST". This would result in the printing of the LC card number, the author and short title, as well as the computer assigned number. The LC card number and the computer assigned number are always printed, whether or not requested. If, however, you wish to print only the LC card number and the computer assigned number, then you must use the abbreviation for card number CN, and nothing else following the print command. To exclude elements, use the word EXCLUDE following the command PRINT, and then list the elements by their two letter abbreviations which you wish to exclude from the printout.

Commands

There are eleven commands in the BIRCH program. Some of these commands you will rarely use, others, you will use frequently. It is important not to confuse search statements with commands. Search statements are those terms which the computer will use to search the MARC file to find catalog records which contain elements matching the terms of your search statement. For example, you might wish to search for a particular LC card number. The card number itself then would be a search statement, that is, the computer would look in the MARC file to see if such a card number were there. An author name would constitute a search statement. Search statements need only be input when the computer asks for a search statement. Whatever sequence of characters you input will be searched for in the indexes to the MARC file. A command, however, gives an instruction to the program to do something other than search for particular words, numbers, etc. Commands are always enclosed in quotes. For example, after you have retrieved something, you may well want to look at the record that has been retrieved. This is accomplished by a "PRINT" command.

You may wish to start a new search series after you have reached the maximum, which is 7. This can be accomplished in several ways. For example, there is a command, ERASE, which will wipe out the last search statement you input. If you follow the word ERASE with a number from one to six, it will erase back to the number you have input. For example, if you should put in the command ERASE 2, then all search statements following search statement 2 will be erased and you will begin at 2. If you simply wish to restart at search

statement 1, you may use the word ABORT as a command. This will always start the entire search sequence over again at 1. You can also use the command RESTART at any point. RESTART takes you right back to the beginning of the program with the opening salutation. It also changes back any new command names which you may have introduced. Command words may be changed at the user's convenience by using the RENAME command. For example, suppose that you do not like the word ABORT, but would instead prefer to use some other term. Suppose that you would prefer to use the term "wipe" instead of abort. You could use the command RENAME to accomplish the change as follows: "RENAME ABORT TO WIPE".

The command DIAGRAM will give you a recapitulation of all search statements which you have used in an existing sequence. The word diagram without any number will give you a recapitulation of every search statement in an existing series back to number one. The use of a number following the word DIAGRAM will give you the recapitulation of only that search statement.

The command FIND is used when you want to put in a search statement at some point in the program other than a normal search statement. For example, suppose the program has asked you how many more items from some search statement you wish to have printed on-line. Instead of answering this, you may wish to indicate that you would like a new search statement at that precise moment. This can be done with the command FIND followed by whatever term you wish to have searched.

The Print Command

The print command is the most complex command in the BIRCH program. Therefore, we have devoted a special paragraph to discuss this command thoroughly. The basic command word, of course, is PRINT. As mentioned above, elements of the record can be either included or excluded in the printing of a record from the file. If the word EXCLUDE is not present then any elements specified are assumed to mean INCLUDE. In addition to these two forms of modification, it is also possible to give a command to print from any search statement. For example, suppose that you input three distinct retrieval statements, each one of them under a search statement number. Suppose after you have been informed the results of the third search statement by the program, that you decide you wish to print an entry from the first search statement. This is easily accomplished, simply by following the elements of the print command with the phrase SSN X, where X is a digit indicating the search statement number desired. If you should wish to print an entry from the search statement number one, the command form would then be "PRINT SSN 1".

Unless otherwise specified, a simple print command will print the number of records retrieved up to five from any retrieval statement. For example, if there are three records in the file which meet your search statement, and you give the command word PRINT, all three of these records will be printed on your terminal. Suppose, however, you wish to print only one, this is easily done by following the command word PRINT by the number you wish to have printed. This number can be any number from 1 up to the number of records meeting the criteria of your search statement; however, the program will always stop after having printed five on the terminal and ask if you wish to continue. This avoids a "runaway" terminal which could occur if, by some mistake you might indicate you wish several hundred printed on the terminal. While it is possible to stop printing at your terminal by use of the "break" key, this is not recommended since the precise mechanism for extricating oneself from such a predicament is somewhat complex and not readily managed by the inexperienced. Off-line printing, of course, is much faster than printing on a terminal; therefore, in the off-line printing mode, any number you request will be copied onto a magnetic tape for later printing by the computer operator. To achieve off-line printing, the user simply adds the word OFF-LINE following the PRINT command. For example, suppose that you have input a search statement requesting, let us say, all the fiction titles from the latest MARC tape. To make the illustration complete, let us assume that the latest MARC tape is number 30. The precise statement to retrieve and then print all of the juvenile works contained on this latest MARC tape and published in 1969 would be as follows: J (LV) AND 30 AND 69+69 (CR) The program responds with the number retrieved, then the user types in "PRINT OFF-LINE" (CR). The program will ask you for your name and address and you may wish to indicate a title for this printout. This can be accomplished by completing the title on the first line of the ORGANIZATION NAME asked for by the program.

The print command also has one final feature which is important to mention here. Frequently, the number of items which you find in the file matching your search statement may be rather large. You may wish to look at one or more of these records before asking for an off-line printout. The normal manner in which the program selects the items for printing, unless controlled by an additional command word, is in order of the most recent first, and then successively less recent items. This is entirely analogous to filing in an accession file of the most recent items first. Looking at the items on your terminal in this manner, would show you the most recent items entered into the file first. Suppose, however, that you wish to print some of the earlier records retrieved. This can be accomplished by use of the word SKIP included in a print command: e.g., "PRINT 3 AU,ST SKIP 20". This particular command would result in the printing of the author and short title of three records beginning at the 21st record from the end. Should you have retrieved 500 records and wish to look at the very first record of those retrieved, then your command would be "PRINT 1 SKIP 499".

The NEIGHBOR Command

BIRCH2 has a command called NEIGHBOR which allows you to look at the values stored in the index. This can be useful, especially in searching authors' names or subject headings (when they are added to the file). For example, you may know an author's last name and perhaps his first initial, but not his full first name; or perhaps you know only the last name of the author. By using the NEIGHBOR command, you can indicate to the system that you would like to look at the index in the area designated by the term which you wish to search. For example, suppose that you are interested in an author named Abodaher. You could use the NEIGHBOR command to find out what Mr. Abodaher's first name might be by the following: "NEIGHBOR ABODAHER." This command will display five terms surrounding the name Abodaher in the alphabetical index. The actual printout of this search is shown below. By including a number from 1 to 10 after the term to be searched, you can control the number of items displayed from the index. This is also demonstrated in the printout below.

```
SS 1 /C?

USER: "NEIGHBOR ABODAHER"

PROG:
        TERM                                    ELEMENT TYPE
POSTINGS

ABLEMAN, PAUL.                                  AUTHOR-CONFERENCE       1
ABLIN, FRED,                                    AUTHOR-CONFERENCE       1
ABODAHER, DAVID J.                              AUTHOR-CONFERENCE       1
ABORTION LAW REFORM ASSOCIATION.                AUTHOR-CONFERENCE       1
ABRAHAM BAR  HIYYA,                             AUTHOR-CONFERENCE       1
ABODAHER  ITSELF DOES NOT EXIST IN THE INDEX.

SS 1 /C?
```

```
"NEW\IGHBOR ABODAHER 8"

PROG:
        TERM                                        ELEMENT TYPE
POSTINGS

ABISCH, ROZ.                                    AUTHOR-CONFERENCE          1
ABLEMAN, PAUL.                                  AUTHOR-CONFERENCE          1
ABLIN, FRED,                                AUTHOR-CONFERENCE          1
ABODAHER, DAVID J.                      AUTHOR-CONFERENCE          1
ABORTION LAW REFORM ASSOCIATION.    AUTHOR-CONFERENCE          1
ABRAHAM BAR  HIYYA,                             AUTHOR-CONFERENCE          1
ABRAHAM, GERALD ERNEST HEAL,                    AUTHOR-CONFERENCE          1
ABRAHAM, ROY CLIVE.                             AUTHOR-CONFERENCE          1
ABODAHER  ITSELF DOES NOT EXIST IN THE INDEX.

SS 1 /C?
```

The EXPLAIN Command

The EXPLAIN command is perhaps the most important one of all in that through its use you can learn all the details about any other command names. "EXPLAIN COMMANDS" will give you a list of all the commands in BIRCH2 (see below). You can then ask for an explanation of each of the command words which will be printed out on your terminal.

The complete explanation of BIRCH2 is included in Appendix 4.

```
USER:"EXPLAIN COMMANDS"

PROG:
BIRCH2 HAS ELEVEN COMMANDS.  THEIR NAMES ARE:

    FIND, NEIGHBOR, DIAGRAM, ERASE, ABORT, RESTART, PRINT, RENAME,
    VERSION, EXPLAIN, AND STOP.
```

Additional Features

There are several additional features that, while not commands, are important in the operation of BIRCH2. There are two special symbols that are reserved in their usage, unless you choose to rename them. These are the pound sign (#) (sometimes known as a hashmark) and the dollar sign. The # is used to indicate a truncated term. You might, for example, input the words INTERNATIONAL CONFERENCE# and receive a listing of all the international conferences that are in the MARC file. This is also useful in inputting an author's name, for you have no way of knowing beforehand whether the Library of Congress has included the author's dates or not in a particular catalog entry. Without the # you must input a term precisely or you will not retrieve an author who might well be in the index. Use of the truncation symbol, however, will allow you to put in the author's last name and perhaps only his first initial and retrieve any item matching the string of characters you have input preceding the pound sign. An example of this is included below.

```
SS 1 /C?

USER:O'NEILL#

PROG:MM (O'NEILL#)

1 O'NEILL, EUGENE GLADSTONE,        AUTHOR-CONFERENCE
2 O'NEILL, JOHN JOSEPH,                 AUTHOR-CONFERENCE
3 O'NEILL, JOHN PATTON,                 AUTHOR-CONFERENCE
4 O'NEILL, JUDITH,                   AUTHOR-CONFERENCE
5 O'NEILL, ROBERT JOHN.                 AUTHOR-CONFERENCE
6 O'NEILL, WILLIAM L.                   AUTHOR-CONFERENCE
7 O'NEILL, WILLIAM L.,                  AUTHOR-CONFERENCE

SPFY

USER:1 "PRINT ST"

PROG:PSTG (1)



CN- 69-16466          NUMBER OF HITS- 01
CAN-2944
ST- SELECTED WORKS.

SS 2 /C?

USER:
```

The dollar sign ($) is used within the program to cancel a complete line of input. This is in addition to the other methods discussed on Page 6 in Chapter 1 of this Manual. Please note, however, that use of the dollar sign to delete a line is applicable only in the BIRCH program and not in the INFIG program.

If you desire to have off-line printout, you must use the command "PRINT OFF-LINE." (This may be the entire file if you so desire.) You will receive the following message if you indicate you wish off-line printout: "XDLO TO BE MOUNTED ON XX." This is a message to the computer operator to mount a tape for printout. XX in the message indicates the number of the tape drive to be used. You will be asked to input your address. After the output has been copied and you have terminated the operation, you should immediately type in the following message exactly as given: /DIAL 41 PLEASE PRINT XDLO ON XX (CR). The XX is the number of the tape drive that was indicated in the message above: "XDLO TO BE MOUNTED ON XX." Simply repeat that number in your message that you have directed to the computer operator by the command /DIAL 41. It is necessary to ask the operator to perform this task since the XDLO tape indicated above is, in a sense, like a scratch pad and is used temporarily for one user and then, later, perhaps for another. If you do not ask the operator to perform the printing job for you the data you have collected off-line on the tape will be lost. Expect to receive some question from the computer operator. The question will begin: FROM 41, and will probably read something like "who is this?" or some such interrogation. In your response, use the name of your institution.

For example: /DIAL 41 Beverly Hills Public Library
San Marino Public Library
Pierce College
Fullerton Junior College
UC Riverside
USC

If the Operator (on channel 41) asks for a Work Order and Org. Code, use the following:

/DIAL 41 J9000, 6321

Remember, any time you wish to send a message to the computer operator, you must put in the command /DIAL 41. If you do not do this, your message will not go to the operator, but will, instead, be lost within the computer. Any message you get from another terminal, such as the operator's terminal, will always begin with the words "FROM XX" where the XX will be the number of the terminal used by the party sending the message to you. Sometimes, the LISTS monitor may send a message to your terminal.

Chapter 3 - Book Order Procedure

A. Introduction

In this chapter, the procedure for creating book orders and the establishment of an in-process file is discussed and illustrated.

B. Book Order Creation

There are two methods of creating book orders. One assumes that the necessary bibliographic data is available in a LISTS file, e.g., the BIBLIO file containing records from MARC tapes. The other method is needed for the creation of book orders for which the bibliographic information is not available in a LISTS file. We will first discuss the situation in which bibliographic information does exist in a LISTS file.

1. General Considerations

Figure 1 shows the elements in the data base for the in-process file (INPROS). The procedure to operate this program is similar to that required to operate the BIRCH program. Figure 1 must be used to guide the user in inputting data since the INFIG program does not include a display of the elements as does the BIRCH program. The number sequence produced by the computer will follow through the elements as listed in Figure 1. Some elements are always skipped because the data will be input at some other time or in another program,* but it is up to the user to decide which elements are not necessary for his purposes and which elements are common to a series of orders. By proper skipping of elements or setting of standard values, the user can create orders at a high rate of speed.

Each library has a typical pattern of operation that will vary somewhat from any other library. To accommodate these differences the number of elements used in the INFIG file is larger than any one library would need. Therefore, you must identify those elements you will not need, so that the INFIG program will not expect input for those elements. For example, you may not wish to use a non-standard vendor; therefore, you would want to skip Element 19 (refer to Figure 1). Or, if your library uses Dewey, you would want to skip Element 3.

---

* Elements 2, 7, 17, and 29 are automatically assigned values and skipped by the program.

THE ELEMENTS OF THE INPROS FILE

Element Number	Retrieval	Qualifying	Description
1	Yes	Yes	LC CARD NUMBER
*2	No	Yes	ORDER DATE
3	Yes	Yes	LC CLASS NO.
4	Yes	Yes	DEWEY CLASS NO.
5	No	No	LC PRICE
6	No	No	EDITION
*7	Yes	Yes	LIBRARY
8	Yes	Yes	AUTHOR-CONFERENCE
9	No	No	SHORT TITLE
10	No	No	TITLE STATEMENT
11	No	No	SERIES NOTE
12	No	No	PLACE OF PUBLICATION
13	Yes	Yes	PUBLICATION DATE
14	Yes	Yes	LEVEL
15	No	No	PUBLISHER
16	Yes	Yes	BOOK ORDER NO.
*17	Yes	Yes	ORDER STATUS
18	Yes	Yes	VENDOR, STANDARD
19	No	No	VENDOR, UNUSUAL
20	Yes	Yes	REQUESTOR
21	Yes	Yes	FUND
22	No	No	NUMBER OF COPIES
23	No	No	VOLUMES
24	No	No	BINDING
25	Yes	Yes	ARRIVAL DATE
26	No	No	NET PRICE
27	No	No	CALL NUMBER
28	Yes	Yes	PURCHASE ORDER NO.
29	Yes	Yes	ORDER CODE
30	No	No	VOUCHER NO.
31	No	No	ITEMS RECEIVED

Figure 1. The Elements of the INPROS File

Some data about orders remains constant for a number of orders, e.g., the vendor, or the fund to be charged. You may set elements to standard values as necessary. You must first identify the element number to be set, and then the value.

After indicating the date of the orders, element numbers to be skipped and/or set, and the values to be set, the program will ask whether you wish the Long or Short mode of operation. The following sections discuss each form in turn.

2. Short-Form Ordering

If bibliographic information is already in a file such as BIBLIO, then a response of "S" to the question: "L/S?" will put the user into the short-form mode which picks up bibliographic information from the BIBLIO file. To capture the bibliographic information, it is necessary to input the CAN (A Computer Assigned Number obtained from the BIBLIO file using the BIRCH program). The program next will ask whether you wish to use default values. These are the values you have just set. You should use the default values, unless a particular title to be ordered requires different values.

If you have no data to input for an element, just hit the space bar and carriage return on your terminal.

After each title has been ordered, you are asked whether you have additional input or are through. You may also delete the entry just entered if you discover a serious error.

3. Long-Form Ordering

The difference between the two versions of order creation have been termed "long" and "short," corresponding to the amount of information that must be input. The "long" form is that used to create a book order when there is no bibliographic information in the file. In the sample operation in Appendix 3, the user will note the initiating protocol and eventually the question ... L/S? ... asked by the program. Responding with an L places the program into the mode to create orders when bibliographic information is not in the file. The only difference in this mode of book-order creation is the need to input bibliographic data. Otherwise, the process is identical to short-form ordering. You may, of course, skip bibliographic elements which you do not need. Also, if you have no data for an element, you may just hit the space bar and then the carriage return on your terminal.

C. Ordering Processing

After you have completed an operation using the INFIG program and have created an in-process file, the LISTS staff at SDC will run several programs that, ultimately, result in a printed book order for each item you have entered into the file. Also, an in-process file is created for each user.

Elements of the In-Process File

"INPROS"

Element Number	Retrieval	Qualifying	Description
0	No	No	CAN: A computer-assigned accession number for the entry. Format: 24 bit binary integer
1	Yes	Yes	LC CARD NUMBER: the number assigned by LC for the catalog card for a bibliographic entity. Format: 12 character string (alphameric)

Element Number	Retrieval	Qualifying	Description
2	No	Yes	ORDER DATE: the date order information was output to the book order program, BORIS. Format: 6 character string (numeric) Coding: characters 1-4 = Julian date, characters 5-6 = two digit number indicating the week of the order, starting with 00 for the first week of LISTS operation.
3	Yes	Yes	LC CLASS NO.: the LC classification for the entry. Format: 8 character string (alphameric)
4	Yes	Yes	DEWEY CLASS NO.: the integer portion of the Dewey decimal classification number for the entry. Format: 3 character string (numeric)
5	No	No	LC PRICE: price for the entry as given in the MARC record, not necessarily in American dollars. Format: 20 character string (alphameric) In some cases, there will be no price for an entry stored in the bibliographic file since the Library of Congress was unable to supply the data. In that case, price must be input by the user even though the rest of the bibliographic data is taken from the MARC file. If the user does not specify the price, then this will be a blank element at output time. Format is 20 characters alphameric.
6	No	No	EDITION: edition description is given in the MARC record. Format: 20 character string (alphameric)
7	Yes	Yes	LIBRARY: an indicator for the library placing the order, for billing purposes, and an indicator for the address to which the order should be shipped. Format: 2 character string (alphameric) Coding: the first character contains an alphameric symbol for the library placing the order; the second character, if not blank, contains an alphameric symbol for the library to which the order is to be shipped. The symbols and their

Element Number	Retrieval	Qualifying	Description
			meaning are: H = Beverly Hills Public Library F = Fullerton Junior College P = Pierce College M = San Marino Public Library S = SDC Technical Library R = UC, Riverside U = USC
8	Yes	Yes	AUTHOR-CONFERENCE: index to 1, 2, or 3 substrings for the entry's author(s), including corporate and conference authored publications, as given in the MARC record. Format: two 16 bit binary integers Coding: first integer contains the word at which the first 64 character substring begins; the second integer, a multiple of 64, contains the total of the substring lengths. If there is no author, both integers are set to $16,448_{10}$ (blanks).
9	No	No	SHORT TITLE: index to the substring for the short title for the entry, as given in the MARC record. Format: two 16 bit binary integers Coding: same as for Element 8, except there is only one substring, of up to 63 characters.
10	No	No	TITLE STATEMENT: index to the substring for the entry's complete title statement, as given in the MARC record. Format: two 16 bit binary integers Coding: same as for Element 8, except there is only one substring, of up to 189 characters.
11	No	No	SERIES NOTE: index to the substring for the entry's series note, as given in the MARC record. Format: two 16 bit binary integers Coding: same as for Element 8, except there is only one substring, of up to 126 characters.
12	No	No	PLACE OF PUBLICATION: place or places where the entry was published, as given in the MARC record. Format: 20 character string (alphabetic)

Element Number	Retrieval	Qualifying	Description
13	Yes	Yes	PUBLICATION DATE: year of publication, copyright, or reprint for the entry. Format: three strings of up to 4 characters each (alphameric) Coding: If there is a single publication date, it will appear in subelement 13_1. If the publication date is not known, subelement 13_1 will contain "N.D.". If there are multiple publication dates, they will appear in subelements 13_1 and 13_2; byte 0 of subelement 13_3 will contain a symbol indicative of the dates' meaning, as follows: 13_1 = date of publication 13_2 = date of copyright 13_3 = C 13_1 = date of reprint 13_2 = date of original publication or blank 13_3 = R 13_1 = starting date of publication 13_2 = terminal date of publication, or 9999 if the terminal date is not known 13_3 = M 13_1 = earliest date of publication, when the date is questionable; 0's are used for the missing digits if the year or decade are not determined 13_2 = latest date of publication, if any; 9's are used for the missing digits if the year or decade are not determined, except that the date must be not later than that of the current year. 13_3 = Q

Element Number	Retrieval	Qualifying	Description
14	Yes	Yes	LEVEL: indicator to show intelectual level of the entry. Format: 1 character (alphameric) Coding: J for juvenile, F for fiction, blank otherwise.
15	No	No	PUBLISHER: index to the substring for the entry's publisher, as given in the MARC record. Format: two 16 bit binary integers Coding: same as for Element 8, except there is only one substring, of up to 63 characters.
16	Yes	Yes	BOOK ORDER NO.: order number supplied by the library placing the book order. Format: 8 character string (alphameric)
17	Yes	Yes	ORDER STATUS: indicator for the order's current status. Format: 1 character (alphabetic) Coding: T = order to be placed, just entered by library; C = order completed, all items received; P = order partially completed, some items or copies yet to be received; X = order cancelled; O = on order, nothing received yet.
18	Yes	Yes	VENDOR, STANDARD: code for the vendor with whom the book order is placed. Format: 6 character string (alphameric) Coding: vendor codes or acronyms of 1 to 6 symbols. If the library does not specify a vendor, LISTS chooses the default vendor for that library. Refer to N-(L)-24176/310 for the list of standard and default vendor codes.
19	No	No	VENDOR, UNUSUAL: index to a string for the name and address of a vendor specified by the ordering library if the vendor is not a standard one carried by LISTS. Format: two 16 bit binary integers Coding: same as for Element 8 of BIBLIO, except there is only one substring of up to 100 characters.

Element Number	Retrieval	Qualifying	Description
20	Yes	Yes	REQUESTOR: name or other information specified by the library to identify the person requesting the order or the source used for selection. Format: 20 character string (alphameric)
21	Yes	Yes	FUND: index to a string for the name of the fund specified by the library against which the order is placed. Format: two 16 bit binary integers Coding: same as Element 8 of BIBLIO, except there is only one substring, of up to 40 characters.
22	No	No	NUMBER OF COPIES: number of copies ordered. Format: 3 character string (numeric) Coding: if the library does not specify a value for this element, it remains blank and LISTS will assume a default of 1.
23	No	No	VOLUMES: list of volumes (of a set, if the whole set is not wanted) for the order. Format: 12 character string (alphameric) Coding: volume number, comma, optional space, etc.; a hyphen may be used to indicate a range of volumes. For accounting purposes, each volume will be considered as 1 copy, so that total items on an order will be the product of the number of copies ordered and the number of volumes specified. This element may be blank.
24	No	No	BINDING: indicator for the type of binding desired. Format: 1 character (alphameric). Coding: L = library binding; P = paperbound; M = microform; X = Xerox; R = regular (to override a default L code for juvenile books); otherwise blank. The book order function may substitute a default for this element, if it is blank, when making up the book order, but the value in INPROS will remain blank.
25	Yes	Yes	ARRIVAL DATE: up to three dates, input by the library and converted to LISTS coding, stating when the ordered items were received, or when the order was cancelled. Format: three strings of 4 characters each, one date per string (alphameric).

Element Number	Retrieval	Qualifying	Description
			Coding: character 1-4 = Julian date; characters 5-6 = two digit number indicating the week of the arrival, starting with 00 for the first week of LISTS operation.
26	No	No	NET PRICE: total money paid to vendor to date on this order. Format: 8 character string (numeric). Coding: American dollars and cents, with trailing blanks.
27	No	No	CALL NUMBER: call number specified by library (assuming it is ordering an additional copy of a book already in the collection). Format: 20 character string (alphameric). Coding: Only the book number portion of the call number appears in this element, beginning in character 9. The class number portion for the call number is obtained from LC CLAS NO. or DEWEY CLASS NO.
28	Yes	Yes	PURCHASE ORDER NO.: Not used at present. If a library uses a purchase order number, it will appear in BOOK ORDER NO.
29	Yes	Yes	ORDER CODE: a code assigned by LISTS to provide a unique identification for each book order, and which may be used by the library to refer to the order. Format: 4 character string (alphabetic). Coding: formed by starting with BBBB for each library and varying the least significant letter (s) by 1 for each new order, i.e., BBBC, BBBD, BBBF, etc. The vowels and Y are not used, to avoid "nasty" words.
30	No	No	VOUCHER NO.: voucher number and date of payment for the order, as input by the library. Format: 20 character string (alphameric).
31	No	No	ITEMS RECEIVED: description of receipts for partial or complete shipments, as input by the library. Format: three strings of 6 characters each (alphameric).

Appendix 1

Condensed Instruction Sheet

-- Your LOGIN code is:

-- The SDC Monitor may be reached at: 393-9411, Ext. 6933 or 6938

-- Current LISTS programs available: 1) BIRCH
2) INFIG

-- Your ID key is:

-- Protocol for Logging-in:

/LOGIN your LOG NUMBER

-- Protocol for BIRCH: /BIRCH V40506

Don't forget carriage return at end of each line!

SAMPLE PROTOCOL

Dial this number: 828-9711
When you hear the tone from the computer, press ORIG button
Hit "return" key (the computer will respond)
Type: /LOGIN (CR)
(the computer will respond)
Type: /BIRCH (or other appropriate program name) V40506

INFIG Protocol

INFIG V40506

The program will give a starting message and identify itself. It will ask if you have an existing daily file. The first time you use the program your answer will be No, but, thereafter, unless you are instructed otherwise, your normal answer will be Yes. It will then ask for your I.D. Use the following as appropriate:

Beverly Hills - BEVH

Fullerton - FULJ

Pierce - PIER

USC - USOC

The program will print a series of messages if the files are new. Otherwise, there will be a wait of about 1 to 2 minutes, after which you will be asked for the order date. This date must be 6 digits, no more, no less. Use a Julian date. Note that only the last two digits can be used for retrieval, therefore, a Julian date form is preferred, leaving the last two digits to indicate the week of the order (beginning 01 for the first week).

Next, you will be asked if you wish to skip elements. Answer as appropriate. If you answer Yes, you will be asked for the element numbers to be skipped. Input these one at a time, waiting after each one for the program to print an asterisk. When you have input the last element number you wish to skip, then hit the space bar and the carriage return.

Next, you will be asked if you wish to set any values for specific elements. If you wish to, answer yes. The program will ask for the element number, and then the value to be set. When you have finished setting all the values you wish, input a space and then carriage return.

At this point you are ready to begin inputting the orders. Refer to the examples in Appendix 3 for each form of ordering (i.e., Long or Short). Remember, if you have no data to input at any point, input a space, and carriage return.

When you are finished, wait until the program announces "Job Complete" before undertaking further operation of your terminal or logging out.

Legal Console Commands

```
/LOAD      p:a          [21                                          ]
/LOADD     p:a          [Vtnnnn  [S1]   [PRI]   [DH]                 ]   [AT xxxxx]
/APPEND    p:a  p:b     [        [S2]   [PUB]   [DM]                 ]
/OVERLAY   p:a  p:b     [                       [DL]                 ]
/p:a

/RESTORE   p:a    [Vtnnnn  [S1]   [PRI]   [DH] ]   [AT xxxxx]
/RESTORED  p:a    [        [S2]   [PUB]   [DM] ]
                                          [DL]

/SAVE  p:a  { /  }  [N]  [Vtnnnn  [S1]  [PRI]  [DH] ]   [AT xxxxx]
            {p:b }  [E]  [        [S2]  [PUB]  [DM] ]
                                               [DL]

/CYLS  Vtnnnn

/DELETE  f:n  [F {1} ]  [Vtnnnn]  [PRI]
                 {2}              [PUB]
                 {3}
                 {4}

/DIAL  terminal:number  message
/DIALON
/DIALOFF
/DRIVES
/DRUMS
/GO  program:name
/FORGET

/LISTF  {ALL     }   Vtnnnn
        {PUBLIC  }
        {user:id }

/LOGIN  user:id  account:number  [pass:word]
/LOGOUT
/QUIT  [program:name]
/STATUS  [program:name]
/STOP  [program:name]
/TIME
/USERS
/CLEAR
/EDIT
/NEDIT
/RELOG  user:id  account:number
```

Appendix 2

Use of BIRCH

```
/BIRCH V40506
*** PGM.STARTED

BIRCH2
THE BIRCH PROGRAM (BIBLIOGRAPHIC SEARCH) IS NOW OPERATING.  IF YOU
WOULD LIKE INFORMATION ABOUT THIS VERSION OF BIRCH TO BE PRINTED OUT
HERE, JUST STRIKE THE SPACE BAR ON THE KEYBOARD ONE OR MORE TIMES,
AND THEN PRESS THE CARRIAGE RETURN KEY.  OTHERWISE, YOU MAY NOW ENTER
SEARCH STATEMENT NO. 1, OR ANY COMMAND.



USER: 71-561\24 OR 70-79\\91351 OR 69-17057 OR 69-16356 OR

PROG:
CNT 1 ---CONTINUE SS 1 ---CONTINUE WITH THE NEXT
ENTRY IN  SEARCH STATEMENT 1.

USER: 72-3823 OR 70-4963 OR 68-67272 OR 69-14263 OR 72-3823 "PRINT"

PROG: PSTG---NUMBER POSTINGS---THE NUMBER OF UNIT RECORDS
MATCHING THIS SEARCH STATEMENT IS (8)



CN- 71-5624          NUMBER OF HITS- 01
CAN-20310
MD- 280919
LC- DA30
DC- 942
PR- *BLANK*
ED- *BLANK*
CO- *BLANK*
AU- BRODRICK, GEORGE CHARLES,                    1831-1903.
AU- FOTHERINGHAM, JOHN KNIGHT,                   1874-1936.
ST- THE HISTORY OF ENGLAND, FROM ADDINGTON'S ADMINISTRATION TO THE
T- THE HISTORY OF ENGLAND, FROM ADDINGTON'S ADMINISTRATION TO THE
   CLOSE OF WILLIAM IV'S REIGN, 1801-1837,  BY GEORGE C. BRODRICK.
   COMPLETED AND REV. BY J. K. FOTHERINGHAM. LONDON, NEW YORK, LO
SE- THE POLITICAL HISTORY OF ENGLAND, V. 11
PL- NEW YORK,
PD- 1969 1906 R
LV- *BLANK*
PB- AMS PRESS,  1969.
FO- B
```

CN- 70-91351 NUMBER OF HITS- 01
CAN-20165
MD- 280919
LC- PR140
DC- 820
PR- *BLANK*
ED- *BLANK*
CO- *BLANK*
AU- VINES, SHERARD, 1890-
ST- THE COURSE OF ENGLISH CLASSICİSM FROM THE TUDOR TO THE VICTORIA
T- THE COURSE OF ENGLISH CLASSICISM FROM THE TUDOR TO THE VICTORIAN AGE.
SE- *BLANK*
PL- NEW YORK,
PD- 1969 1930 R
LV- *BLANK*
PB- PHAETON PRESS, 1968 I.E. 1969
FO- B

CN- 69-17057 NUMBER OF HITS- 01
CAN-20052
MD- 280919
LC- PS3515E3
DC- 813
PR- 4.95
ED- *BLANK*
CO- *BLANK*
AU- HEMINGWAY, ERNEST, 1899-1961.
ST- THE FIFTH COLUMN,
TI- THE FIFTH COLUMN, AND FOUR STORIES OF THE SPANISH CIVIL WAR.
SE- *BLANK*
PL- NEW YORK,
PD- 1969
LV- *BLANK*
PB- SCRIBNER 1969
FO- *BLANK*

CN- 69-16356 NUMBER OF HITS- 01
CAN-20043
MD- 280919
LC- GR825
DC- 398
PR- 7.95
ED- *BLANK*
CO- *BLANK*
AU- HAMEL, FRANK.
ST- HUMAN ANIMALS;
TI- HUMAN ANIMALS; WEREWOLVES OTHER TRANSFORMATIONS.
SE- *BLANK*
PL- NEW HYDE PARK, N.Y.,
PD- 1969 1915 R
LV- *BLANK*
PB- UNIVERSITY BOOKS 1969
FO- B

CN- 69-14263 NUMBER OF HITS- 01
CAN-20006
MD- 280919
LC- PS3515E3
DC- 813
PR- *BLANK*
ED- *BLANK*
CO- *BLANK*
AU- HOWELL, JOHN M., COMP.
AU- HEMINGWAY, ERNEST, 1899-1961. THE SHORT HAPP
A- HEMINGWAY, ERNEST, 1899-1961. THE SNOWS OF K

ST- HEMINGWAY'S AFRICAN STORIES;
TI- HEMINGWAY'S AFRICAN STORIES; THE STORIES, THEIR SOURCES, THEIR CRITICS COMPILED BY JOHN M. HOWELL.
SE- SCRIBNER RESEARCH ANTHOLOGIES
PL- NEW YORK,
PD- 1969
LV- *BLANK*
PB- SCRIBNER 1969
FO- B

HOW MANY MORE ON-LINE?

USER: NONE

```
PROG:
SS 2 /C?---SEARCH STATEMENT 2 OR COMMAND?--
ENTER SEARCH STATEMENT NUMBER 2 OR ANY COMMAND.

USER: "PRT 8 CN"

PROG:


CN- 71-5624          NUMBER OF HITS- 01
CAN-20310



CN- 70-91351         NUMBER OF HITS- 01
CAN-20165


CN- 69-17057         NUMBER OF HITS- 01
CAN-20052


CN- 69-1 356         NUMBER OF HITS- 01
CAN-20043


CN- 69-14263         NUMBER OF HITS- 01
CAN-20006

CONTINUE PRINTING? (YES/NO)

USER: YES
```

PROG:

CN- 68-67272 NUMBER OF HITS- 01
CAN-19956

CN- 70-4963 NUMBER OF HITS- 01
CAN-13912

CN- 72-3823 NUMBER OF HITS- 02
CAN-6543

SS 2 /C?---SEARCH STATEMENT 2 OR COMMAND?--
ENTER SEARCH STATEMENT NUMBER 2 OR ANY COMMAND.

USER: 029

PROG: PSTG---NUMBER POSTINGS---THE NUMBER OF UNIT RECORDS MATCHING THIS SEARCH STATEMENT IS (18)

SS 3 /C?---SEARCH STATEMENT 3 OR COMMAND?--
ENTER SEARCH STATEMENT NUMBER 3 OR ANY COMMAND.

USER: "SHORT ALL"

PROG:
SS 3 /C?---SEARCH STATEMENT 3 OR COMMAND?

USER: ABODAHER# OR BRODRICK, GEO# OR FREUND, WILL# "PRT ST"

PROG: PSTG---NUMBER POSTINGS (4)

CN- 71-5624 NUMBER OF HITS- 01
CAN-20310
ST- THE HISTORY OF ENGLAND, FROM ADDINGTON'S ADMINISTRATION TO THE

CN- 68-67272 NUMBER OF HITS- 01
CAN-19956
ST- FOOD SERVICE EQUIPMENT: ESTIMATED NUMBER OF UNITS BY KIND OF BU

```
CN- 69-13841          NUMBER OF HITS- 01
CAN-13759
ST- THE HISTORY OF ENGLAND FROM ADDINGTON'S ADMINISTRATION TO THE C


CN- 69-13048          NUMBER OF HITS- 01
CAN-9377
ST- FREEDOM FIGHTER: CASIMIR PULASKI,

SS 4 /C?---SEARCH STATEMENT 4 OR COMMAND?

USER: /STOP BIRCH
PGM.STOPPED
```

DESCRIPTION OF BIBLIO ELEMENTS

If an element is a retrieval element, it is indexed to allow retrieval of the entry based on a given value for that element. Similarly, if an element is a qualifying element, it is indexed to allow inclusion or exclusion from retrieval based on a given value for the element.

Element Number	Retrieval	Qualifying	Description
-	No	No	CAN: a computer-assigned accession number for the entry. Format: 24 bit binary integer
-	No	No	DATED: a number indicating the week of the MARC tape from which the entry came. Format: 8 bit binary integer Coding: conversion of characters 5-6 of Element 2.
1	Yes	Yes	LC CARD NUMBER: the number assigned by LC for the catalog card for a bibliographic entity. Format: 12 character string (alphameric)
2	No	Yes	MARC DATE: the date on which the MARC tape was received. Format: 6 character string (numeric) Coding: Characters 1-4 = Julian date; characters 5-6 = two digit number indicating the week of receipt, starting with 00 for the first MARC tape, 01 for the second tape, etc.
3	Yes	Yes	LC CLASS NO.: the LC classification for the entry. Format: 8 character string (alphameric)
4	Yes	Yes	DEWEY CLASS NO.: the integer portion of the Dewey decimal classification number for the entry. Format: 3 character string (numeric)
5	No	No	LC PRICE: price for the entry as given in the MARC record, not necessarily in American dollars. Format: 20 character string (alphameric)
6	No	No	EDITION: edition description as given in the MARC record. Format: 20 character string (alphameric)

Element Number	Retrieval	Qualifying	Description
7	Yes	Yes	CONFERENCE INDICATOR: indicates whether or not the entry was authored by a conference, symposium, etc. Format: 1 character (alphameric) Coding: blank if not authored by a conference; C if conference authored
8	Yes	Yes	AUTHOR-CONFERENCE: index to 1, 2, or 3 substrings for the entry's author(s), including corporate and conference authored publications, as given in the MARC record. Format: two 16 bit binary integers Coding: first integer contains the word at which the first 64 character substring begins; the second integer, a multiple of 64, contains the total of the substring lengths. If there is no author, both integers are set to $16,448_{10}$ (blanks).
9	No	No	SHORT TITLE: index to the substring for the short title for the entry, as given in the MARC record. Format: two 16 bit binary integers Coding: same as for Element 8, except there is only one substring, of up to 63 characters.
10	No	No	TITLE: index to the substring for the entry's complete title, as given in the MARC record. Format: two 16 bit binary integers Coding: same as for Element 8, except there is only one substring, of up to 189 characters.
11	No	No	SERIES NOTE: index to the substring for the entry's series note, as given in the MARC record. Format: two 16 bit binary integers Coding: same as for Element 8, except there is only one substring, of up to 120 characters.
12	No	No	PLACE OF PUBLICATION: place or places where the entry was published, as given in the MARC record. Format: 20 character string (alphabetic)
13	Yes	Yes	PUBLICATION DATE: year(s) of publication for the entry. Format: three strings of 4 characters each, one year per string (alphameric)

Element Number	Retrieval	Qualifying	Description
			Coding: If there is a single publication date, it will appear in subelement 13_1. If the publication date is not known, subelement 13_1 will contain "N.D.". If there are multiple publication dates, they will appear in subelements 13_1 and 13_2; byte 0 of subelement 13_3 will contain a symbol indicative of the dates' meaning, as follows: 13_1 = date of publication 13_2 = date of copyright 13_3 = C 13_1 = date of reprint 13_2 = date of original publication 13_3 = R 13_1 = starting date of publication 13_2 = terminal date of publication, or 9999 if the terminal date is not known 13_3 = M 13_1 = earliest date of publication, when the date is questionable; 0's are used for the missing digits if the year or decade are not determined 13_2 = latest date of publication, if any; 9's are used for the missing digits if the year or decade are not determined, except that the date must be not later than that of the current year. 13_3 = Q
14	Yes	Yes	LEVEL: indicator to show intelectual level of the entry. Format: 1 character (alphameric) Coding: J for juvenile, F for fiction, blank otherwise.

Element Number	Retrieval	Qualifying	Description
15	No	No	PUBLISHER: index to the substring for the entry's publisher, as given in the MARC record. Format: two 16 bit binary integers Coding: same as for Element 8, except there is only one substring, of up to 63 characters.
16	No	No	FORM OF CONTENTS: indicator for the entry's type, such as a dictionary, encyclopedia, etc., as given in the MARC record. Format: 1 character (alphameric) Coding: as described in The MARC II Format.

FORM OF CONTENT CODES

Certain types of material used frequently for reference purposes will be specified by a one-character alphabetic code. Since some works embody several types of materials, as many as four may be specified. This information will be taken from the title statement, the subject headings, or the notes. In the unlikely eventuality that more than four types are present, the sequence of the following list will determine which four are to be recorded.

b = Bibliographies
c = Catalogs
i = Indexes
a = Abstracts
d = Dictionaries
e = Encyclopedias
r = Directories
y = Yearbooks
s = Statistics
h = Handbooks
p = Programmed Textbooks
b = blank (not a specified form of content)

Appendix 3

Use of INFIG

Beginning on page 3-3 of this Appendix, you will find a sample log from an INFIG session. In the following discussion, the line numbers refer to that sample.

After starting the program (at line 1), the program returns several messages identifying itself, and asking the question on line 4, whether or not there is an existing daily file to update. Normally, the answer will be no. The program then asks for an identification. This is a four-letter code which is assigned for each library. Obtain your code from the LISTS monitor at SDC.

From line 8 through 12, the system returns messages indicating that some files are being established on a disc pack to hold the in-process orders. The system then asks for an order date. This is the Julian form of date that we use routinely in the LISTS System. The last two digits are an indication of a cycle period such as a week or some sort of ordering cycle. In the illustration on line 13, there is a mistake that was not corrected, showing the order date is 1969, day 349, week 02. This is an incorrect statement for the present program, but will be correct for the new version of the order program due to be implemented in the near future. The current correct form of input is to have the first three digits indicate the Julian day, and the fourth digit the year.

At line 14 the program asks whether any elements are to be skipped. In this illustration, the answer is "yes," followed by the element numbers. You must wait for the computer to print an asterisk before you input the number and strike the carriage return.

At line 26, the program asks whether you wish to set the standard values for any of the elements. In this illustration, the answer is "no." (On the following pages you will find some illustrations where the answer is "yes," and you will see that the program will ask for the element number and then the value in each case.)

When you are through with a list of either elements to be skipped or of the values to be set, your final input is simply a space and a carriage return which tells the program that you are finished. The next question then, as on line 28, is whether you wish to use long-or short-form. In this illustration the short-form is used for the first item, CAN number is input, and the default option chosen (that is to use the standard values, if any, that have been set either as long-term standards or as daily standards). The answer "yes" in this instance would have captured whatever default values had been stored. Since the answer was "no," the system asks for a book order number, which in this case (on line 33) is A1. The system then asks whether there is more input (line 34), we answer "yes" (on line 35) as we start the cycle over again.

The attempt to use the term ABORT (on line 40) is unsuccessful, since INFIG does not recognize this term. The only point where a record may be terminated or discarded is at the question (as on line 34) "more input." Selection of the letter D as your response would erase the previous item that had been input, that is, delete the data concerned with one record. For example, at line 35, had the response been D it would have deleted the bibliographic and order information of the book represented by the computer assigned number 1728.

Beginning on line 42, we see the input of a long-form record. On line 63, we see the beginning of a sequence of information where the default option was not used. This allows additional information to be input concerning the order.

Please refer to your list of elements to identify the particular items that were input in the above illustration. The illustration following the first shows a similar operation where values were set and default options used or not, as the case may be, showing the effect of skipping and setting different elements on the interaction between user and program.

```
/INFIG V40506
*** PGM.STARTED 10000
LISTS BOOK ORDER FILE GENERATION PROGRAM IS OPERATING.
 HAVE AN EXISTING DAILY FILE TO UPDATE? Y/N
N
YOUR ID PLEASE
PIER
FILE JZPIERXX, 2314 VOL. 0506 OPENED ON D5
FILE JZPIER00, 2314 VOL. 0506 OPENED ON D5
FILE JZPIER13, 2314 VOL. 0506 OPENED ON D5
FILE JZPIER14, 2314 VOL. 0506 OPENED ON D5
FILE JZPIER19, 2314 VOL. 0506 OPENED ON D5
ORDER DATE? 934902
SKIP ANY ELEMENTS? Y/N
Y
ELEM. NOS?
*4
*14
*25
26
WAIT
**26
*27
*28
*
SET VALUES? Y/N
N
L/S?
S
CAN?     1728
DEFAULT? Y/N
Y
16. A1
MORE INPUT? Y/N/D
Y
L/S?
L
DEFAULT? Y/N
*ABORT*
DEFAULT? Y/N
Y
01.
03.
05. 4.98
06.
```

```
08. SAPPO, ELMINGTON E.
*
09. THIS IS THE LONG FORM
10.
11.
12. SANTA MONICA
13. 1969
*
15. ERPSO PUBLICATIONS
16. A1
MORE INPUT? Y/N/D
Y
L/S?
S
CAN?     3427
DEFAULT? Y/N
N
16. A2
18. NJPREN
19.
20. PEARSON
21. SPECIAL
22.
23.
24.
30.
31.
MORE INPUT? Y/N/D
N
        3 INPUTS PROCESSED
JOB COMPLETE

PGM.STOPPED
```

```
/INFIG V40506
PGM.STARTED 10000
LISTS BOOK ORDER FILE GENERATION PROGRAM IS OPERATING.
 HAVE AN EXISTING DAILY FILE TO UPDATE? Y/N
N
YOUR ID PLEASE
PIER
FILE JZPIERXX, 2314 VOL. 0506 OPENED ON D7
FILE JZPIER00, 2314 VOL. 0506 OPENED ON D7
FILE JZPIER13, 2314 VOL. 0506 OPENED ON D7
FILE JZPIER14, 2314 VOL. 0506 OPENED ON D7
FILE JZPIER19, 2314 VOL. 0506 OPENED ON D7
ORDER DATE? 318901
SKIP ANY ELEMENTS? Y/N
Y
ELEM. NOS?
*4
*7
*10
*14
*25
*26
*19
*23
*24
*27
*28
*30
*31
*
SET VALUES? Y/N
Y
ELEM. NO.?
16
VALUE?
33456
ELEM. NO.?
18
VALUE?
NJPREN
ELEM. NO.?
21
VALUE?
BOOK
ELEM. NO.?
L/S?
S
CAN?    20310
```

```
DEFAULT? Y/N
Y
MORE INPUT? Y/N/D
Y
L/S?
S
CAN?     20165
DEFAULT? Y/N
Y
MORE INPUT? Y/N/D
Y
L/S?
S
CAN?     20052
DEFAULT? Y/N
Y
MORE INPUT? Y/N/D
Y
L/S?
L
DEFAULT? Y/N
Y
01. WELTSCHMERZ, @ ←The "Break" key was used here to cancel this line.

67-166666
03.
05. $6.75
06.
08. WELTSCHMERZ, THADDEUS
*
09. HOW TO LIVE WITH TIME-SHARING
11.
12. MENA, ARK.
13. 1966
* 1968
* R
15. JONES PUB. CO
MORE INPUT? Y/N/D
Y
L/S?
S
CAN?     20006
DEFAULT? Y/N
N
20. BLACK
22. 4
MORE INPUT? Y/N/D
N
        5 INPUTS PROCESSED
JOB COMPLETE

PGM.STOPPED
```

Appendix 4

BIRCH2
THE BIRCH PROGRAM (BIBLIOGRAPHIC SEARCH) IS NOW OPERATING. IF YOU WOULD LIKE INFORMATION ABOUT THIS VERSION OF BIRCH TO BE PRINTED OUT HERE, JUST STRIKE THE SPACE BAR ON THE KEYBOARD ONE OR MORE TIMES, AND THEN PRESS THE CARRIAGE RETURN KEY. OTHERWISE, YOU MAY NOW ENTER SEARCH STATEMENT NO. 1, OR ANY COMMAND.

TYPING ERRORS:
WHILE TYPING A MESSAGE YOU MAY MAKE A TYPOGRAPHICAL ERROR. YOU CAN CORRECT SUCH AN ERROR IN TWO WAYS. ONE IS BY DELETING THE ERRING CHARACTER (OR SPACE) AND ALL CHARACTERS AND SPACES THAT YOU HAVE TYPED AFTER IT IN THAT LINE, AND THEN RETYPING. THE OTHER WAY IS BY DELETING THE ENTIRE LINE IN WHICH THE ERROR OCCURS, AND THEN RETYPING.

IF YOU WISH TO DELETE ONLY A FEW CHARACTERS, YOU MAY DO SO BY USING THE BACK SLASH CHARACTER " \ " WHICH IS THE UPPERCASE L KEY. YOU MUST BE CAREFUL, HOWEVER, IN USING THIS PARTICULAR METHOD OF ERROR CORRECTION, BECAUSE THE TELETYPE DOES NOT BACKSPACE. THEREFORE, YOU MUST COUNT CAREFULLY THE NUMBER OF CHARACTERS YOU WISH TO DELETE AND STRIKE ONLY THAT MANY BACKSLASHES. IF YOU HAVE MANY CHARACTERS TO CORRECT, IT IS SAFER TO USE THE FOLLOWING PROCEDURE. THE OTHER METHOD OF TYPOGRAPHICAL ERROR CORRECTION IS TO DELETE THE ENTIRE LINE AND RETYPE IT. YOU DO THIS BY THE FOLLOWING STEPS:

1. STRIKE THE DOLLAR-SIGN KEY (UPPER-CASE 8).
2. STRIKE THE CARRIAGE RETURN KEY.
3. WAIT FOR THE WORD "USER:" TO APPEAR AT THE BEGINNING OF THE NEXT LINE.
4. RETYPE THE ENTIRE LINE.

DO YOU WISH A FULL EXPLANATION OF HOW TO USE THE PROGRAM, WHICH TAKES MANY MINUTES TO PRINT, A REFRESHER CONSISTING OF A LIST OF THE COMMANDS AND THE PROGRAM'S PROCEDURAL MESSAGES, WHICH TAKES 4 MINUTES, OR NONE? TYPE THE WORD "FULL," "REFRESH," OR "NONE," AND STRIKE THE CARRIAGE RETURN KEY.

FULL EXPLANATION
SINCE THE FULL EXPLANATION OF BIRCH 2 TAKES CONSIDERABLE TIME TO PRINT, YOU SHOULD SAVE IT FOR FUTURE REFERENCE. BIRCH MAY UNDERGO EVOLUTIONARY CHANGES, THEREFORE, WHEN YOU OPERATE THIS PROGRAM, PLEASE MAKE SURE THAT THE VERSION OF THE EXPLANATION YOU REFER TO CORRESPONDS TO THE VERSION OF THE PROGRAM OPERATING AT THAT TIME.

THE BIRCH PROGRAM IS A USER-COMMUNICATIONS PACKAGE WHICH PROVIDES THE USER WITH HIGHLY FLEXIBLE TUTORIAL ACCESS TO THE MARC BIBLIOGRAPHIC FILES OF LISTS. LISTS IS A BIBLIOGRAPHIC INFORMATION STORAGE, RETRIEVAL, AND PROCESSING SYSTEM THAT STORES BIBLIOGRAPHIC RECORDS OF DOCUMENTS, CONDUCTS USER-DIRECTED SEARCHES OF ITS FILES OF DOCUMENT RECORDS, PRINTS OUT INFORMATION FROM RETRIEVED RECORDS AT THE USER'S COMMAND, AND TRIGGERS OTHER ON- OR OFF-LINE PROCESSES.

UNIT RECORD:
THE STORED RECORD FOR EACH ENTRY (CALLED A UNIT RECORD) CONTAINS 16 CATEGORIES OF INFORMATION ABOUT THAT DOCUMENT. SEE THE LIST ON PAGES B-56 AND B-57

OPERATING OVERVIEW:
THE PROGRAM IS SIMPLE TO OPERATE, ONCE YOU LEARN WHAT THE VARIOUS COMMANDS AND MESSAGES MEAN. BY ISSUING CERTAIN COMMANDS TO THE PROGRAM YOU CAUSE IT TO SEARCH THE FILES FOR UNIT RECORDS THAT MATCH THE SPECIFICATIONS YOU HAVE INCLUDED IN YOUR SEARCH STATEMENT. THESE SPECIFICATIONS ARE COMPOSED OF TERMS REFERRING TO THE CATEGORIES OF INFORMATION FROM THE UNIT RECORD. WHEN THE PROGRAM FINDS AND RETRIEVES UNIT RECORDS THAT MATCH THE SPECIFICATIONS, YOU CAN ISSUE OTHER COMMANDS THAT CAUSE THE PROGRAM TO PRINT OUT INFORMATION IN SELECTED CATEGORIES OF THE RETRIEVED UNIT RECORDS. THE CATEGORIES PRINTED OUT NEED NOT BE THE SAME ONES ON WHICH THE FILES WERE SEARCHED TO FIND THE UNIT RECORDS, ALTHOUGH THEY CAN BE. FOR EXAMPLE, YOU MIGHT SPECIFY A SEARCH USING A LIBRARY OF CONGRESS CLASSIFICATION NUMBER AND THEN HAVE ONLY THE AUTHORS' NAMES PRINTED OUT FROM THE RETRIEVED UNIT RECORDS, OR VICE-VERSA. THUS, A VERY WIDE RANGE OF COMBINATIONS OF SEARCH PATTERNS AND PRINTOUT OPTIONS ARE AVAILABLE TO YOU.

PACING YOURSELF:
YOU MUST AT TIMES WAIT FOR THE COMPUTER TO DIGEST WHAT YOU HAVE ALREADY INPUT. THEREFORE, YOU MUST PACE YOURSELF. WHEN THE PROGRAM CAN ACCEPT INPUT (YOUR COMMANDS OR ANSWERS) IT PRINTS THE WORD "USER:" AT THE BEGINNING OF A NEW LINE AND WAITS. WHEN THE PROGRAM CANNOT ACCEPT INPUT (BECAUSE AT THE TIME IT IS PROCESSING A TASK FOR YOU) IT PRINTS THE WORD "PROG:" (SHORT FOR PROGRAM) AT THE BEGINNING OF A NEW LINE. THIS IS THE SIGNAL FOR YOU TO WAIT FOR THE PROGRAM TO FINISH ITS PROCESSING, PRINT OUT ITS RESULTS, AND THEN PRINT THE WORD "USER:" AT THE BEGINNING OF A NEW LINE. YOU MAY THEN MAKE YOUR NEXT INPUT.

TWO INPUT FORMS:
YOU CAN INPUT MESSAGES TO THE PROGRAM IN EITHER OF TWO WAYS, DEPENDING ON THE SITUATION; YOU CAN ANSWER THE PROGRAM, OR YOU CAN ISSUE A COMMAND.

ANSWERING THE PROGRAM:
WHEN THE PROGRAM SUGGESTS OR REQUIRES A CERTAIN RESPONSE FROM YOU, YOUR RESPONSE IS CALLED AN ANSWER. TO INPUT AN ANSWER YOU:

1. TYPE THE ANSWER.
2. CHECK VISUALLY FOR TYPING ACCURACY, AND INSERT THE ANSWER BY STRIKING THE CARRIAGE RETURN KEY.

ISSUING A COMMAND:
A COMMAND CAN BE ISSUED AT ANY TIME THAT THE PROGRAM WILL ACCEPT INPUTS, I.E., WHENEVER THE PROGRAM HAS PRINTED "USER:" AT THE BEGINNING OF A NEW LINE. TO ISSUE A COMMAND YOU:

1. TYPE A DOUBLE QUOTE (") MARK.
2. TYPE THE COMMAND MESSAGE.
3. TYPE ANOTHER DOUBLE QUOTE MARK.
4. CHECK FOR TYPING ACCURACY, AND INSERT THE COMMAND BY STRIKING CARRIAGE RETURN KEY. (NOTE THAT COMMANDS MUST BE SURROUNDED BY DOUBLE QUOTE MARKS, BUT ANSWERS MUST NOT BE.)

BELOW ARE DEFINITIONS AND DESCRIPTIONS OF SOME TERMS AND PHRASES USED TO DESCRIBE THE BIRCH2 PROGRAM:

SEARCH STATEMENT:
A SEARCH STATEMENT IS A SET OF SPECIFICATIONS YOU INPUT TO THE PROGRAM, THAT DESCRIBE THE KIND OF UNIT RECORDS FOR WHICH YOU WISH THE PROGRAM TO CONDUCT A SEARCH. A SEARCH STATEMENT IS COMPOSED OF A SINGLE SEARCH ENTRY, (DEFINED BELOW) OR A STRING OF SEARCH ENTRIES LINKED TOGETHER BY BOOLEAN OPERATORS (DEFINED BELOW).
YOU INPUT A SEARCH STATEMENT AS AN ANSWER TO A READINESS CUE PRINTED OUT TO YOU BY THE PROGRAM, E.G.,

PROG: SS(NUMERAL)/C?--SEARCH STATEMENT (NUMERAL) OR COMMAND?--
ENTER SEARCH STATEMENT NUMBER (NUMERAL) OR ANY COMMAND?

NOTICE THAT THE READINESS CUE IS REPEATED THREE TIMES; FIRST IN A SYMBOLIC VERSION, THEN A SHORT VERSION, AND FINALLY A LONG VERSION. AS SOON AS YOU LEARN THE MEANING OF THE SHORT VERSION YOU CAN SUPPRESS THE LONG VERSION, AND WHEN THE SYMBOLIC VERSION IS COMPLETELY FAMILIAR YOU CAN INSTRUCT THE COMPUTER TO PROVIDE YOU WITH THE SYMBOLIC VERSION ONLY. (MORE ABOUT THIS WILL BE EXPLAINED LATER.)

AFTER PRINTING THE READINESS CUE, THE PROGRAM PRINTS THE WORD "USER:" AT THE BEGINNING OF A NEW LINE, AND AWAITS YOUR ANSWER. YOU MAY THEN ENTER A SEARCH STATEMENT INTO THE PROGRAM BY:

1. TYPING A SEARCH ENTRY OR STRING OF ENTRIES LINKED BY BOOLEAN OPERATORS,
2. STRIKING THE CARRIAGE RETURN KEY TO INSERT THE SEARCH STATEMENT INTO THE PROGRAM.

BOOLEAN OPERATORS:
THE TWO BOOLEAN OPERATORS ARE THE LINKING WORDS, "AND" AND "OR". THEY ARE TYPED WITHOUT QUOTATION MARKS.

THE MEANING OF (LOGICAL) "AND":
WHEN TWO OR MORE SEARCH ENTRIES ARE LINKED BY THE WORD AND, THE PROGRAM RETRIEVES ONLY THOSE UNIT RECORDS THAT CONTAIN (I.E., MATCH WITH) ALL THE LINKED ENTRIES.

THE MEANING OF (LOGICAL) "OR":
WHEN TWO OR MORE SEARCH ENTRIES ARE LINKED BY THE WORD "OR," THE PROGRAM RETRIEVES ALL UNIT RECORDS THAT CONTAIN (MATCH WITH) ANY ONE OR MORE OF THE LINKED ENTRIES.

LINKING:
MULTIPLE ENTRIES IN A SEARCH STATEMENT MUST BE LINKED TOGETHER BY EITHER THE WORD "AND" OR THE WORD "OR" INSERTED BETWEEN THEM. THE SEARCH STATEMENT MUST FORM AN UNBROKEN CHAIN, WITH LINKING WORDS APPEARING ON BOTH SIDES OF EACH ENTRY, EXCEPT BEFORE THE FIRST ENTRY AND AFTER THE LAST. AN EXAMPLE OF LINKING IN A SEARCH STATEMENT WOULD BE:

SMITH AND 1969 OR QC544

FOR THIS EXAMPLE, THE PROGRAM WILL RETRIEVE ALL UNIT RECORDS THAT HAVE BEEN INDEXED UNDER (BEEN POSTED TO) BOTH THE TERMS SMITH AND 1969, AND WOULD ALSO RECORD THOSE HAVING THE TERM QC544 AS A POSTING.

SEARCH ENTRIES:
A SEARCH ENTRY CAN CONSIST OF A SINGLE LIBRARY OF CONGRESS CLASS CARD NUMBER, A PUBLICATION DATE, THE LETTER C INDICATING A CONFERENCE PUBLICATION, THE LETTER R INDICATING A REPRINT, THE LETTER J INDICATING JUVENILE LITERATURE, THE LETTER F INDICATING FICTION, OR A SEARCH STATEMENT NUMBER (SSN).

SEARCH STATEMENT NUMBER (SSN)
THE PROGRAM AUTOMATICALLY NUMBERS EACH SEARCH STATEMENT YOU INSERT BY ASSIGNING YOUR FIRST STATEMENT THE NUMERAL 1, YOUR SECOND THE NUMERAL 2, AND SO ON. UP TO SEVEN SEARCH STATEMENTS CAN BE HELD IN ONE SEARCH SERIES.

SEARCH SERIES
A SEARCH SERIES CONSISTS OF A NUMBER OF SEARCH STATEMENTS HELD SIMULTANEOUSLY IN THE PROGRAM'S ACTIVE SEARCH REGISTER. IN THIS WAY THE PROGRAM CAN INTERPRET RELATIONSHIPS BETWEEN THE VARIOUS SEARCH STATEMENTS IN ONE SERIES.

THE MAXIMUM NUMBER OF SEARCH STATEMENTS THAT THE PROGRAM'S SEARCH REGISTER WILL HOLD AT ONE TIME IS SEVEN. THEREFORE, WHEN YOU HAVE INSERTED YOUR SEVENTH SEARCH STATEMENT IN A SERIES, THE PROGRAM WILL INFORM YOU THAT IT CANNOT ACCEPT ANOTHER STATEMENT UNTIL YOU HAVE PROVIDED ROOM IN THE REGISTER (WHICH YOU CAN DO IN A NUMBER OF WAYS TO BE DESCRIBED LATER).

WITHIN A SEARCH SERIES, EARLIER SEARCH STATEMENTS CAN BE INCORPORATED AS SINGLE SEARCH ENTRIES IN LATER ONES. SSN 1 CAN, FOR EXAMPLE, BE INCORPORATED AS A SEARCH ENTRY IN ANY OR ALL OF THE LATER SSNS 2,3,4, 5,6, AND 7. ALSO, SSN 2 CAN BE INCORPORATED IN SSNS 3,4,5,6, AND 7, (BUT IT OBVIOUSLY CANNOT BE INCLUDED IN SSN 1, SINCE THAT STATEMENT WAS COMPLETED BEFORE SSN 2.) THIS INCORPORATION FEATURE OF THE PROGRAM ALLOWS FOR VERY COMPLEX AND LOGICALLY POWERFUL NESTED SEARCHES TO BE CONDUCTED FROM THE LATER STATEMENTS IN A SERIES.

NEGATION
TO NEGATE A SEARCH ENTRY IN A SEARCH STATEMENT, YOU PRECEDE THE ENTRY WITH THE WORD "NOT". THE PROGRAM, IN SEARCHING, WILL THEN REJECT ANY UNIT RECORD ON
UNIT RECORD CONTAINING (POSTED TO) THE NEGATED ENTRY. AN EXAMPLE OF NEGATION IN A SEARCH STATEMENT IS:

QC544 AND NOT 1969

THE SEARCH (FOR THIS EXAMPLE) WOULD SELECT UNIT RECORDS POSTED TO THE LC CLASS QC544 AND NOT PUBLISHED IN 1969.

QUALIFICATION
TO QUALIFY A SEARCH ENTRY, YOU FOLLOW THAT ENTRY WITH THE CATEGORY NAME OR TWO-LETTER DESIGNATOR, IN PARENTHESES, OF THE UNIT RECORD CATEGORY TO BE SEARCHED FOR THAT ENTRY.
QUALIFICATION OF AN ENTRY MIGHT APPEAR IN EITHER OF TWO WAYS, SHOWN IN THE FOLLOWING EXAMPLES:

QC544 OR BRAIN (AUTHOR)
QC544 OR BRAIN (AU)

THE PROGRAM WOULD, FOR BOTH EXAMPLES, SELECT UNIT RECORDS CONTAINING THE CLASS QC544 AND WOULD ALSO SELECT THOSE WHOSE AUTHOR'S LAST NAME WAS BRAIN.

LONG SEARCH STATEMENTS

A LONG SEARCH STATEMENT MAY REQUIRE MORE THAN ONE LINE ON THE TELETYPEWRITER RECORD. RECALL, HOWEVER, THAT TO INSERT A MESSAGE YOU STRIKE THE CARRIAGE RETURN KEY. THEREFORE IF YOU STRIKE THE CARRIAGE RETURN KEY AFTER THE LAST ENTRY IN THE FIRST LINE OF YOUR LONG SEARCH STATEMENT, THE PROGRAM INSERTS THE LINE AS A FINISHED SEARCH STATEMENT. TO CIRCUMVENT THIS DIFFICULTY, THE PROGRAM HAS BEEN DESIGNED SO THAT IF YOU STRIKE THE CARRIAGE RETURN KEY FOLLOWING EITHER OF THE LINKING WORDS "AND" OR "OR", THE PROGRAM INTERPRETS THE SEARCH STATEMENT AS CONTINUING UNBROKEN INTO THE NEXT LINE OF THE RECORD. IN THAT CASE, IT PRINTS THE WORD USER: AT THE BEGINNING OF THE NEXT LINE, AND WAITS FOR YOU TO CONTINUE FORMULATING YOUR LONG SEARCH STATEMENT.

OVERFLOW

AN OVERFLOW CONDITION CAN SOMETIMES OCCUR IN THE PROGRAM WHEN IT TRIES TO RESPOND TO A SEARCH STATEMENT. OVERFLOW CAN COME FROM TWO CAUSES:

1. THERE MIGHT BE TOO MANY SEARCH ENTRIES IN THE STATEMENT (THE ABSOLUTE UPPER LIMIT IS 99 ENTRIES, AND THE EFFECTIVE LIMIT IS USUALLY SOMEWHAT LESS DEPENDING ON THE KINDS OF LINKS SPECIFIED BETWEEN ENTRIES.)
2. THERE MIGHT BE TOO MANY UNIT RECORDS IN THE FILE THAT ARE POSTED TO (I.E., CONTAIN) THE COMBINATION OF ENTRIES YOU SPECIFIED IN YOUR SEARCH STATEMENT).

WHEN OVERFLOW OCCURS, THE PROGRAM WILL NOTIFY YOU AND INDICATE THE CAUSE. (THE FORM OF THE COMPUTER'S OVERFLOW MESSAGE WILL BE DESCRIBED LATER.)

FORMULATION OF COMMANDS

BIRCH2 HAS ELEVEN COMMANDS AVAILABLE. THE COMMAND NAMES ARE: FIND, NEIGHBOR, DIAGRAM, ERASE, ABORT, PRINT, EXPLAIN, RENAME, VERSION, RESTART, AND STOP. IN THE FOLLOWING EXPLANATIONS OF COMMANDS, ANGLE BRACKETS WILL BE PLACED AROUND WORDS OR SYMBOLS THAT YOU MUST TYPE EXACTLY AS SHOWN. THE ANGLE BRACKETS ARE NOT TO BE TYPED. THE DESCRIPTIONS OF OPTIONAL WORDS AND SYMBOLS TO BE CHOSEN BY YOU ARE ENCLOSED IN DOUBLE COLONS. THE DOUBLE COLONS ARE NOT TO BE TYPED.

FORM OF COMMAND MESSAGES

ALL COMMAND MESSAGES HAVE THE SAME GENERAL FORM:

```
<"><COMMAND NAME>< >::MESSAGE CONTENTS::<"><CRK>
 1          2     3              4            5   6
```

IN THIS EXAMPLE, THE ELEMENTS OF EACH STEP IN COMPOSING THE COMMAND MESSAGE HAVE BEEN NUMBERED. STEPS 2 AND 4 WILL VARY, DEPENDING ON WHICH COMMAND YOU ARE USING. THE REMAINDER OF THE STEPS ARE THE SAME FOR ALL COMMANDS. THE STEPS ARE:

1. TYPE A DOUBLE QUOTE MARK; <">.
2. WITHOUT TYPING ANY SPACES, TYPE THE NAME OF THE COMMAND YOU WISH TO USE, OR ONE OF THE ACCEPTABLE VARIANTS OF THE COMMAND NAME; <COMMAND NAME OR VARIANT>.
3. IF THERE IS A VARIABLE MESSAGE, PRESS SPACE BAR ONCE.
4. TYPE THE VARIABLE MESSAGE CONTENTS;::MESSAGE CONTENTS::. THESE MAY BE OPTIONAL, AND WILL BE EXPLAINED BELOW FOR EACH COMMAND. (SOME COMMANDS NEED NO OPTIONAL MESSAGE CONTENTS, WHILE OTHERS REQUIRE EXTENSIVE VARIABLE MESSAGE CONTENTS.)
5. TYPE ANOTHER DOUBLE QUOTE MARK;<">. (CHECK FOR ACCURACY)
6. STRIKE CARRIAGE RETURN KEY TO ENTER MESSAGE; <CRK>.

IN THE EXPLANATIONS OF COMMANDS TO FOLLOW, FOUR POINTS WILL BE COVERED FOR EACH:

1. THE PURPOSE AND OPERATION OF THE COMMAND WILL BE DESCRIBED.
2. THE ACCEPTABLE VARIANTS OF THE COMMAND NAME WILL BE LISTED.
3. AN EXAMPLE COMMAND MESSAGE WILL BE PRESENTED.
4. THE VARIABLE MESSAGE CONTENT STEP OR STEPS (IF SUCH THERE ARE) WILL BE DESCRIBED FOR THAT COMMAND.

THE FIND COMMAND

THE FIND COMMAND IS USED WHEN YOU WISH TO ENTER A SEARCH STATEMENT INTO THE PROGRAM WITHOUT HAVING RECEIVED A READINESS CUE SIGNAL FROM THE PROGRAM. (FOR EXAMPLE, THE PROGRAM MIGHT HAVE JUST ASKED YOU FOR ADDITIONAL PRINTING INSTRUCTIONS ON A SEARCH YOU HAD PREVIOUSLY FORMULATED, BUT YOU MIGHT DECIDE TO ABANDON THAT PRINTOUT, IGNORE THE PROGRAM'S REQUEST, AND START ENTERING A NEW SEARCH STATEMENT. YOU WOULD USE THE FIND COMMAND IN THIS CASE.) WHEN A FIND COMMAND IS ENTERED, THE PROGRAM PROCESSES THE CONTAINED SEARCH STATEMENT IN THE USUAL MANNER.

AN EXAMPLE COMMAND MESSAGE IS:

<"><FIND>< >::SEARCH STATEMENT::<"><CRK>

THE VARIABLE COMMAND MESSAGE CONTENTS ARE:

THE SEARCH STATEMENT. (SEARCH STATEMENTS WERE DESCRIBED EARLIER IN THIS EXPLANATION.)

THE NEIGHBOR COMMAND

A NEIGHBOR COMMAND IS USED WHEN YOU HAVE A TERM IN MIND, AND WISH TO VIEW OTHER TERMS THAT ARE ALPHABETICALLY ADJACENT (NEIGHBOR) TO THAT TERM. SUCH INFORMATION MIGHT HELP YOU EXPAND OR REFINE YOUR SEARCH STATEMENT WITH VARIANTS OF THE TERM YOU HAVE IN MIND, OR MAY PROVIDE YOU OTHER RELATED TERMS. WHEN A NEIGHBOR COMMAND IS ISSUED, THE PROGRAM PRINTS OUT A LIST OF UP TO TEN ALPHABETICALLY ORDERED CONSECUTIVE TERMS, WITH THE SPECIFIED TERM APPEARING AS THE MIDDLE TERM OF THE LIST. IF THE TERM YOU SPECIFIED DOES NOT APPEAR IN THE ALPHABETIZED INDEX (FOR EXAMPLE, OF AUTHOR'S LAST NAMES), THE PROGRAM SELECTS THE TERM IN THE LIST THAT IS ALPHABETICALLY MOST SIMILAR TO THE SPECIFIED ONE, AND PRINTS OUT THE LIST OF TERMS NEIGHBORING IT.

THE ACCEPTABLE VARIANTS OF THE COMMAND NAME ARE:
<NEIGHBOR> OR <NBR>

AN EXAMPLE COMMAND MESSAGE IS:
<"><NBR>< >::JONES (AU) 7::<"><CRK>

THE VARIABLE COMMAND MESSAGE CONTENTS ARE:

1. THE SPECIFIED TERM (IN THIS EXAMPLE, JONES).
2. THE TWO-LETTER DESIGNATOR IN PARENTHESES OF THE UNIT RECORD CATEGORY FROM WHICH THE ALPHABETIZED LIST IS TO BE TAKEN. IN THIS EXAMPLE, THE LIST IS TO BE TAKEN FROM THE LAST NAME (AU). THIS STEP IS OPTIONAL. IF IT IS INCLUDED, ONLY TERMS FROM THE SPECIFIED CATEGORY OF THE UNIT RECORD ARE PRINTED. IF IT IS OMITTED, ALL ALPHABETICALLY NEIGHBORING TERMS ARE PRINTED, WITH THEIR UNIT RECORD CATEGORIES INDICATED ON THE PRINTOUT.
3. A NUMERAL SPECIFYING THE NUMBER OF TERMS YOU WANT PRINTED, UP TO TEN TERMS. THIS STEP IS OPTIONAL, AND IF IT IS OMITTED FIVE TERMS WILL BE PRINTED.

THE DIAGRAM COMMAND

YOU CAN ISSUE A DIAGRAM COMMAND WHEN YOU WISH TO VIEW THE LOGICAL STRUCTURE OF A SEARCH STATEMENT THAT YOU HAVE PREVIOUSLY ENTERED IN A CURRENT SEARCH SERIES. A DIAGRAM COMMAND MESSAGE MAKES THE PROGRAM PRINT A LOGICAL DESCRIPTION OF THE SEARCH STATEMENT YOU SPECIFY (BY NUMBER) IN YOUR COMMAND MESSAGE. IF NO NUMBER IS SPECIFIED, A DESCRIPTION IS PRINTED OF THE MOST RECENT SEARCH STATEMENT YOU ENTERED.

THE ACCEPTANLE VARIANTS OF THE COMMAND NAME ARE:
<DIAGRAM> OR <DIAG>

AN EXAMPLE COMMAND MESSAGE IS:
<"><DIAG>< >::7::<"><CRK>

THE VARIABLE COMMAND MESSAGE CONTENTS ARE:
THE SEARCH STATEMENT NUMBER (SSN) OF THE SEARCH STATEMENT YOU WISH TO SEE DIAGRAMMED.

THE ERASE COMMAND

YOU CAN USE AN ERASE COMMAND WHEN YOU WISH TO CANCEL LATER MEMBERS OF A SERIES OF SEARCH STATEMENTS YOU HAVE ENTERED DURING ONE RETRIEVAL SERIES (A SINGLE RETRIEVAL SERIES CAN HAVE UP TO SEVEN SEARCH STATEMENTS). THE ERASE COMMAND CAUSES THE PROGRAM TO DELETE THE SEARCH STATEMENT SPECIFIED IN THE COMMAND MESSAGE, AND ALL SUBSEQUENT SEARCH STATEMENTS. THE PROGRAM THEN RETURNS TO THE STATE IT WAS IN BEFORE THE SPECIFIED SEARCH STATEMENT WAS FORMULATED. IF THE SSN IS OMITTED FROM THE COMMAND MESSAGE, THE PROGRAM ERASES ONLY THE LATEST SEARCH STATEMENT ENTERED.

THE ACCEPTABLE VARIANTS OF THE COMMAND NAME ARE:
<ERASE> OR <ERS>

AN EXAMPLE COMMAND MESSAGE IS:
<"><ERS>< >::5::<"><CRK>

THE VARIABLE COMMAND MESSAGE CONTENTS ARE:
A NUMERAL REPRESENTING THE SSN OF THE SEARCH STATEMENT YOU WISH ERASED (ALONG WITH ALL SUBSEQUENT ONES.)

THE ABORT COMMAND
THE COMMAND CAN BE ISSUED WHEN YOU WISH TO CANCEL ALL MEMBERS OF A SERIES OF SEARCH STATEMENTS YOU HAVE FORMULATED DURING ONE RETRIEVAL SERIES. THE ABORT COMMAND CAUSES THE PROGRAM TO DELETE ALL SS'S FORMULATED IN THAT SERIES, AND TO RETURN TO THE STATE IT WAS IN BEFORE FORMULATION OF SS NUMBER 1.

THE ACCEPTABLE VARIANTS OF THE COMMAND NAME ARE:
<ABORT>

AN EXAMPLE COMMAND MESSAGE IS:
<"><ABORT><"><CRK>

THE VARIABLE COMMAND MESSAGE CONTENTS ARE:
NONE

THE RESTART COMMAND
YOU CAN USE A RESTART COMMAND WHENEVER YOU WISH, FOR WHATEVER REASONS, TO CAUSE THE PROGRAM TO ERASE ALL STORED RECORDS OF INTERACTIONS BETWEEN YOURSELF AND THE PROGRAM, AND TO RETURN TO GREETING THE USER AND ASKING IF HE WISHES AN EXPLANATION. THE RESTART COMMAND IS USEFUL FOR INSTALLING A DIFFERENT USER WITHOUT INTERRUPTING THE PROGRAM. IT ALSO ALLOWS A NEOPHYTE USER TO START OVER AFRESH IF HE BECOMES TOO CONFUSED.

WHEN THE RESTART COMMAND MESSAGE HAS BEEN INSERTED, THE PROGRAM RESPONDS BY REMINDING YOU THAT THE RESTART ACTION WOULD ERASE ALL COMPUTER STORED RECORDS OF YOUR JUST-ACCOMPLISHED INTERACTIONS WITH THE PROGRAM. IT THEN REQUIRES YOU TO REAFFIRM YOUR RESTART COMMAND BY TYPING THE LETTER Y AND STRIKING THE CARRIAGE RETURN KEY. IF YOU THEN TAKE THIS ACTION, THE PROGRAM ACCOMPLISHES A RESTART.

THE ACCEPTABLE VARIANTS OF THE COMMAND NAME ARE:
<RESTART> OR <RST>

AN EXAMPLE COMMAND MESSAGE IS:
<"><RST><"><CRK>

THE VARIABLE COMMAND MESSAGE CONTENTS ARE:
NONE

THE PRINT COMMAND
THE PRINT COMMAND IS USED WHEN YOU WANT TO EXAMINE ONE OR MORE OF THE CATEGORIES OF DESCRIPTIVE INFORMATION IN THE UNIT RECORDS THAT THE PROGRAM HAS RETRIEVED BY USING YOUR SEARCH STATEMENT. THE PRINT COMMAND MESSAGE IS EXTREMELY FLEXIBLE, AND ALLOWS YOU CHOICES AMONG A LARGE NUMBER OF OPTIONS.

THE ACCEPTABLE VARIANTS OF THE COMMAND NAME ARE:
<PRINT> OR <PRT>

AN EXAMPLE COMMAND MESSAGE IS:
<"><PRT>< > ::10,SSN 4,HITS, COMPACT, ON-LINE, INCLUDE, T9::<"><CRK>

THE VARIABLE COMMAND MESSAGE CONTENTS ARE:
THE OPTIONAL PRINT COMMAND SPECIFICATIONS DESCRIBED BELOW:

THERE ARE SIX PRINT COMMAND OPTIONS, ANY OF WHICH MAY BE OMITTED OR ENTERED IN ANY ORDER, WITH EACH SEPARATED BY A COMMA.

NUMBER OF ITEMS (SPECIFIED BY <NUMERALS>)	YOU WILL OFTEN WISH TO EXAMINE INFORMATION FROM ONLY SOME OF THE UNIT RECORDS IN THE SET RETRIEVED BY YOUR SEARCH STATEMENT. YOU CAN SPECIFY HOW MANY ARE TO BE SELECTED.
<SSN> (FOLLOWED BY <NUMERAL>)	THE NUMERAL IS THE NUMBER OF THE SEARCH STATEMENT FOR WHICH YOU WISH TO HAVE UNIT RECORDS RETRIEVED AND INFORMATION PRINTED.
<HITS> OR <MARC>	HITS MEANS INFORMATION FROM UNIT RECORDS IS PRINTED IN THE ORDER OF NUMBER OF HITS ON THE UNIT RECORD, THAT IS, THE NUMBER OF CORRESPONDENCES BETWEEN THE UNIT RECORD AND THE SEARCH STATEMENT. MARC MEANS INFORMATION IS PRINTED BY ORDER OF MARC DATE, WITH MOST RECENT INFORMATION PRINTED FIRST.
<COMPACT> OR <INDENTED>	COMPACT MEANS THE PRINTOUT TAKES AS LITTLE PAPER AS PRACTICABLE. INDENTED PRODUCES A STANDARD PRINTOUT IN MORE PRESENTABLE FORM.

<ON-LINE> OR <OFF-LINE>

ON-LINE MEANS PRINTING ON THE TELETYPE. OFF-LINE MEANS THAT THE OUTPUT GOES ONTO A MAGNETIC TAPE, WHICH CAN LATER BE FED TO A HIGH-SPEED PRINTER. THE LATTER OPTION IS USEFUL FOR PRINTING LARGE VOLUMES OF MATERIAL.

<INCLUDE> OR <EXCLUDE> FOLLOWED BY ::CATEGORY DESIGNATORS FOR THE UNIT RECORDS::

INCLUDE MEANS THAT THE PROGRAM IS TO PRINT INFORMATION FROM ONLY THE UNIT RECORD CATEGORIES YOU SPECIFY. EXCLUDE MEANS PRINT FROM CATEGORIES EXCEPT THOSE YOU SPECIFY. IF THIS OPTION IS DELETED FROM THE COMMAND, ALL CATEGORIES ARE PRINTED. IF NEITHER THE WORD INCLUDE NOR EXCLUDE IS SPECIFIED IN THE COMMAND, BUT CATEGORY DESIGNATORS ARE SPECIFIED, THE PROGRAM ASSUMES BY DEFAULT THAT THE WORD INCLUDE WAS SPECIFIED. IF YOU WISH TO SPECIFY MORE CATEGORIES THAN WILL FIT ON THE FIRST LINE OF YOUR PRINT COMMAND MESSAGE, YOU ENTER THE TERM "INCLUDE" OR "EXCLUDE", BUT DO NOT SPECIFY ANY CATEGORY DESIGNATORS. WHEN THE COMMAND IS INSERTED, THE PROGRAM WILL RESPOND BY PROMPTING YOU AS FOLLOWS:

PROG: LIST CATEGORIES TO BE (INCLUDED, EXCLUDED), USING COMMAS.
USER:

IF YOU THEN LIST YOUR CATEGORIES SEPARATED BY COMMAS AND END THE FIRST LINE WITH A COMMA, THE ABOVE USE WILL BE REPEATED AFTER YOU STRIKE THE CARRIAGE RETURN KEY. YOU CAN THEN CONTINUE LISTING CATEGORIES ON THE NEW LINE. (IF NINE OR MORE CATEGORIES ARE DESIRED, IT IS EASIER TO EXCLUDE THE UNDESIRED CATEGORIES.)

THE CATEGORY NAMES, OR THEIR TWO-LETTER ABBREVIATIONS, ARE ENTERED WITH COMMAS IN BETWEEN EACH. THE 16 CATEGORIES AND THEIR DESIGNATORS ARE LISTED BELOW:

LC CARD NUMBER	CN	SHORT TITLE	ST
MARC DATE	MD	TITLE	TI
LC CLASS NO.	LC	SERIES NOTE	SE
DEWEY CLASS NO.	DC	PLACE OF PUBLICATION	PL

LC PRICE	PR	LEVEL	LV
EDITION	ED	PUBLISHER	PB
CONFERENCE INDICATOR	CO	FORM OF CONTENTS	FO
AUTHOR-CONFERENCE	AU		

IF ANY OF THE ABOVE SIX OPTIONS (NUMBER OF ITEMS, SEARCH STATEMENT NUMBER, HITS OR MARC, COMPACT OR INDENTED, ON-LINE OR OFF-LINE INCLUDE OR EXCLUDE CATEGORIES) ARE NOT SPECIFIED IN THE COMMAND, THE PRINTOUT AUTOMATICALLY WILL BE IN THE DEFAULT MODE FOR THAT OPTION. THE DEFAULT MODE PROVIDES FIVE ITEMS FROM THE MOST RECENT SEARCH STATEMENT, ORDERED BY MARC-DATE IN COMPACT FORMAT, PRINTED ON-LINE, WITH ALL CATEGORIES PRINTED.
HERE ARE SOME EXAMPLES OF PRINT COMMAND MESSAGES:

"PRINT SSN 4, 20, ON-LINE, EXCLUDE, LEVEL"
"PRINT INCLUDE, CN, PR, AN, PB"
"PRINT"
"PRINT 10, HITS, STANDARD"

THESE EXAMPLES ILLUSTRATE TWO POINTS. FIRST, PRINT OPTIONS CAN BE SPECIFIED IN ANY ORDER IN THE PRINT COMMAND MESSAGE. SECOND, WORDS SUCH AS (STANDARD), WHICH ARE NOT ACCEPTABLE TERMS FOR THE PRINT OPTIONS, ARE SIMPLY IGNORED AND THE PROGRAM PRINTS THE DEFAULT MODE.

THE RENAME COMMAND:
A RENAME COMMAND IS USED WHEN YOU FIND YOURSELF UNCOMFORTABLE WITH USING THE NAMES THAT ARE PROVIDED FOR DESCRIBING PROGRAM ACTIONS. (USERS WHO ARE HABITUATED TO USING OTHER COMMAND LANGUAGES MAY FIND THIS ACTION ESPECIALLY VALUABLE.) ANY OF THE COMMAND NAMES, THE LINKING WORDS "AND" AND "OR," THE NEGATION WORD "NOT," AND COMMAND MESSAGE SPECIFICATION TERMS CAN BE RENAMED FOR YOUR PURPOSES. OTHER USERS OF THE PROGRAM WILL NOT BE AFFECTED BY YOUR CHANGES. A RENAMED NAME CAN BE ANY ARBITRARILY CHOSEN WORD, OR ANY SYMBOL ON THE KEYBOARD THAT IS NOT ALREADY USED FOR SIGNALLING PURPOSES.

THE ACCEPTABLE VARIANTS OF THE COMMAND NAME ARE:
<RENAME> OR <RNM>

EXAMPLES OF THE COMMAND MESSAGE FORM ARE:
<"><RNM>< >::AND::< ><TO>< >::*::<"><CRK>
<"><RENAME>< >::OR::< ><TO>< >::+::<"><CRK>
<"><RNM>< >::FIND::< ><TO>< >::SEARCH::<"><CRK>

THE VARIABLE CONTENTS OF THE COMMAND MESSAGE ARE:
1. THE WORD THAT IS TO BE RENAMED.
2. THE WORD OR SYMBOL THAT IS TO REPLACE THE RENAMED WORD.

THE VERSION COMMAND.
YOU USE A VERSION COMMAND TO TELL THE PROGRAM HOW MUCH DETAIL TO INCLUDE IN THE ROUTINE MESSAGES IT SENDS TO YOU. MANY MESSAGES FROM THE PROGRAM TO YOU ARE SENT REPEATEDLY (THE EIGHT MAIN TYPES OF SUCH ROUTINE PROGRAM MESSAGES WILL BE DESCRIBED IN A LATER SECTION OF THIS EXPLANATION). UNLESS COMMANDED OTHERWISE, THE PROGRAM PRINTS SUCH ROUTINE MESSAGES IN WHAT IS CALLED A COMPOSITE FORM, WHICH IS DESIGNED TO AID YOU IN QUICKLY BECOMING FAMILIAR WITH THE MESSAGES. IN THE COMPOSITE FORM, THE MESSAGE IS REPEATED THREE CONSECUTIVE TIMES, FIRST IN A SYMBOLIC VERSION, THEN A SHORT VERSION AND FINALLY A LONG VERSION.

THE ACCEPTABLE VARIANTS OF THE COMMAND NAME ARE:
<VERSION> OR <VERS>

AN EXAMPLE OF THE COMMAND MESSAGE FORM IS:
<"><VERS>< >::<SHORT><READINESS CUE>::<"><CRK>
THE VARIABLE CONTENTS OF THE COMMAND MESSAGE ARE:

1. THE FIRST WORD IN THE VARIABLE CONTENTS MUST BE EITHER THE WORD SYMBOLIC, SHORT, OR LONG.
2. THE REMAINDER OF THE VARIABLE CONTENTS CAN CONSIST OF ANY ONE OF THE NAMES OF THE EIGHT ROUTINE TYPES OF PROGRAM MESSAGES (DESCRIBED LATER); CAN BE LEFT BLANK, OR; CAN BE THE WORD "ALL."

BEFORE YOU ISSUE ANY VERSION COMMAND, THE PROGRAM PRINTS OUT ALL ITS ROUTINE MESSAGES TO YOU IN THE FULL COMPOSITE FORM. THE WORD LONG APPEARING IN THE VARIABLE CONTENTS PORTION OF A VERSION COMMAND MESSAGE WILL ALSO CAUSE THE FULL COMPOSITE FORM TO BE PRINTED. THE WORD SHORT CAUSES THE LONG VERSION TO BE DELETED, AND ONLY THE SYMBOLIC AND SHORT VERSIONS OF THE MESSAGES TO BE PRINTED. THE WORD SYMBOLIC CAUSES THE LONG AND SHORT VERSIONS TO BE DELETED, AND ONLY THE SYMBOLIC VERSION TO BE PRINTED.

IF YOU SPECIFY THE NAME OF A ROUTINE PROGRAM MESSAGE IN YOUR VERSION COMMAND MESSAGE, THE PROGRAM WILL THEREAFTER MAKE THE SPECIFIED DELETIONS ONLY FOR THE TYPES OF MESSAGE YOU NAMED. IF YOU SPECIFY THE WORD ALL, THE SPECIFIED DELETIONS WILL BE PERFORMED ON ALL TYPES OF ROUTINE MESSAGES. IF YOU LEAVE THE MESSAGE TYPE PORTION OF VERSION COMMAND MESSAGE BLANK, THE DELETION YOU SPECIFY WILL BE PERFORMED THEREAFTER ON THE TYPE OF ROUTINE MESSAGE THAT THE PROGRAM SENT TO YOU JUST BEFORE YOU ISSUED THAT VERSION COMMAND.

THE EXPLAIN COMMAND:
YOU MAY USE AN EXPLAIN COMMAND WHENEVER THERE IS ANYTHING ABOUT THE SYSTEM'S MESSAGES OR ITS COMMANDS THAT HAS SLIPPED YOUR MIND. WHEN YOU SPECIFY THE NAME OF A PROGRAM MESSAGE, OR A COMMAND, OR OF A TERM OR PHRASE DEFINED IN THE PRESENT EXPLANATION, THE PROGRAM WILL SELECT OUT THE APPROPRIATE PORTION OF THE PRESENT EXPLANATION AND PRINT IT FOR YOU.

THE ACCEPTABLE VARIANTS OF THE COMMAND NAME ARE:
<EXPLAIN><EX><HELP> <DESCRIBE>

EXAMPLES OF THE COMMAND MESSAGE FORM ARE:
<"><EXPLAIN>< >::<PRINT>::<"><CRK>
<"><EX>< >::<COMMANDS>::<"><CRK>

THE VARIABLE CONTENTS OF THE COMMAND MESSAGE ARE:

1. THE ACCEPTABLE VARIANTS OF THE NAMES OF ALL COMMANDS, THE NAMES AND THE SYMBOLIC VERSIONS OF ALL ROUTINE PROGRAM MESSAGES, AND ANY TERMS OR PHRASES DEFINED IN THIS FULL EXPLANATION. (WHEN ANY OF THESE ARE SPECIFIED, THE PROGRAM PRINTS OUT AN APPROPRIATE EXPLANATION.)
2. THE WORD COMMANDS. (WHEN THE WORD COMMANDS IS SPECIFIED, THE PROGRAM PRINTS OUT A LIST OF THE NAMES OF THE COMMANDS.)
3. A BLANK. (WHEN THE VARIABLE CONTENT PORTION OF THE COMMAND MESSAGE IS LEFT BLANK, THE PROGRAM PRINTS A LIST OF PERMISSIBLE ACTIONS YOU MIGHT TAKE (CONSIDERING THE STATUS OF THE PROGRAM AT THAT MOMENT), TO ACHIEVE FURTHER PROGRESS IN YOUR INTERACTION WITH THE PROGRAM.

THE STOP COMMAND:
YOU ISSUE A STOP COMMAND WHEN YOU WISH TO CLOSE THE BIRCH PROGRAM AND ITS FILES, WITH RESPECT TO THE TERMINAL THAT YOU ARE USING. WHEN YOU ISSUE A STOP COMMAND, THE PROGRAM RESPONDS BY REQUIRING YOU TO REAFFIRM YOUR COMMAND BY TYPING THE WORD "YES," FIRST REMINDING YOU THAT THE STOP ACTION WILL CLOSE THE FILES AND SHUT DOWN THE PROGRAM FOR YOUR TERMINAL. IF YOU THEN TYPE THE WORD "YES" AND STRIKE THE CARRIAGE RETURN KEY, THE PROGRAM ACCOMPLISHES THE STOP ACTION.

THE ACCEPTABLE VARIANTS OF THE COMMAND NAME ARE:
<STOP>

AN EXAMPLE OF THE COMMAND MESSAGE IS:
<"><STOP><"><CRK>

THE VARIABLE CONTENTS OF THE COMMAND MESSAGE ARE:
NONE

FORM OF PROGRAM MESSAGES:
PROGRAM MESSAGES ARE ROUTINE, OFTEN-REPEATED STANDARD MESSAGES THAT THE PROGRAM SENDS TO YOU TO INFORM YOU ABOUT ITS STATUS, OR ABOUT ITS FINDINGS IN SEARCHING ITS FILES IN RESPONSE TO YOUR COMMANDS. EACH PROGRAM MESSAGE IS PRINTED IN WHAT IS CALLED A COMPOSITE FORM, WHICH CONSISTS OF THREE CONSECUTIVELY REPEATED VERSIONS OF THE MESSAGE: A SYMBOLIC VERSION, A SHORT VERSION, AND A LONG VERSION. THE SYMBOLIC

VERSION CONSISTS OF ONE WORD OR SHORT GROUP OF LETTERS. THE SHORT VERSION CONSISTS OF A FEW WORDS OR PHRASES AND IS EASIER TO RECOGNIZE. THE LONG VERSION CONSISTS OF A FULLY EXPLANATORY SENTENCE. AS YOU BECOME FAMILIAR WITH THE PROGRAM MESSAGES, YOU CAN, BY USING A VERSION COMMAND (EXPLAINED EARLIER) PROGRESSIVELY DELETE THE LONG AND SHORT VERSIONS OF EACH MESSAGE.

EIGHT TYPES OF ROUTINE PROGRAM MESSAGES (PM) ARE DESCRIBED BELOW. IN THE INDENTED EXAMPLE OF THE FULL COMPOSITE FORM OF EACH PM MESSAGE, CURVED PARENTHESES ARE USED TO SET OFF MESSAGE CONTENTS THAT ARE VARIABLE.

THE READINESS CUE (PM):
A READINESS CUE IS PRINTED WHENEVER THE PROGRAM CAN ACCEPT A SEARCH STATEMENT OR ANY COMMAND. THE FULL COMPOSITE FORM OF THE READINESS CUE IS:

SS (NUMERAL)/C?---SEARCH STATEMENT (NUMERAL) OR COMMAND?---
ENTER SEARCH STATEMENT NUMBER (NUMERAL) OR ANY COMMAND.

FOR EXAMPLE, IF SEARCH STATEMENTS 1 AND 2 HAVE ALREADY BEEN FORMULATED, THEN:

THE SYMBOLIC FORM OF THE READINESS CUE IS:
SS 3/C?

THE SHORT FORM OF THE READINESS CUE IS:
SS 3/C?---SEARCH STATEMENT 3 OR COMMAND?

THE LONG FORM OF THE READINESS CUE IS:
SS 3/C?---SEARCH STATEMENT 3 OR COMMAND?---ENTER SEARCH STATEMENT NUMBER 3 OR ANY COMMAND.

THE CONTINUATION ACKNOWLEDGEMENT (PM):
THE PROGRAM PRINTS A CONTINUATION ACKNOWLEDGEMENT WHENEVER YOU END A TELETYPE LINE WITH AN "AND" OR AN "OR." THE ACKNOWLEDGMENT SERVES AS A SIGNAL THAT THE PROGRAM UNDERSTANDS THAT YOU HAVE NOT COMPLETED YOUR SEARCH STATEMENT. THE FULL COMPOSITE FORM OF THE CONTINUATION ACKNOWLEDGEMENT IS:

CNT (NUMERAL)---CONTINUE SSN (NUMERAL)---CONTINUE WITH THE NEXT ENTRY IN SEARCH STATEMENT (NUMERAL), OR ENTER SPACE TO TERMINATE.

FOR EXAMPLE, IF YOUR FORMULATION OF SEARCH STATEMENT 3 TOOK MORE THAN ONE LINE, AND YOU WERE USING THE SYMBOLIC FORM OF PROGRAM MESSAGE, THE CONTINUATION ACKNOWLEDGEMENT MESSAGE WOULD APPEAR:

CNT 3

THE POSTINGS NUMBER (PM):
WHEN YOU INSERT A SEARCH STATEMENT INTO THE PORGRAM, THE PROGRAM IMMEDIATELY ACKNOWLEDGES THE INSERTION BY PRINTING THE WORD "PROG:" WHEN IT HAS COMPLETED THE PROCESSING IT PRINTS OUT A MESSAGE AFTER THE WORD "PROG:." THE MESSAGE TELLS YOU HOW MANY UNIT RECORDS IN THE FILES MATCH YOUR SEARCH STATEMENT. THE COMPOSITE FORM OF THE MESSAGE IS:

PSTG (NUMERAL)---NUMBER POSTINGS (NUMERAL)---THE NUMBER OF UNIT RECORDS MATCHING THE SEARCH STATEMENT JUST ENTERED IS (NUMERAL).

THE NO POSTINGS (PM)
THE PROGRAM PRINTS A NO POSTINGS MESSAGE WHEN YOU INSERT A SEARCH STATEMENT THAT HAS ONE OR MORE SEARCH ENTRIES FOR WHICH THE PROGRAM CANNOT FIND POSTINGS. THIS CAN RESULT FROM SEVERAL CAUSES. YOU MAY HAVE:

1. CORRECTLY SPELLED A TERM NOT EXISTING IN THE INDEX FILE.
2. MISSPELLED A TERM WHICH DOES NOT EXIST IN THE INDEX.
3. MISSPELLED A TERM WHICH EXISTS, AND HAS POSTINGS.

OFTEN YOU CAN FIND OUT THE CAUSE BY ISSUING A "NEIGHBOR" COMMAND.

IF THE NON-POSTED TERM OR TERMS HAS BEEN LINKED BY "AND" TO OTHER TERMS IN THE SEARCH STATEMENT, THE PROGRAM WILL AUTOMATICALLY DELETE THEM. IF A PRINT COMMAND IS ISSUED FOR A SEARCH STATEMENT CONTAINING NON-POSTED TERMS, THE PROGRAM IGNORES THESE TERMS AND ALL OTHER TERMS LINKED TO THEM BY "AND." THE FULL COMPOSITE FORM OF THE NO-POSTINGS MESSAGE IS:

NP (TERM)---NO POSTINGS (TERM)---THERE IS NO POSTING FOR THE TERM (TERM), IT AND ALL OTHER TERMS LINKED TO IT BY THE WORD "AND" HAVE BEEN DELETED FROM THE SEARCH STATEMENT YOU JUST ENTERED.

THE MULTI-MEANING (PM)
A MULTI-MEANING PM IS PRINTED WHEN THE PROGRAM DISCOVERS THAT, IN FORMULATING A SEARCH STATEMENT, YOU HAVE ENTERED A TERM WHICH HAS MULTIPLE MEANINGS, AND HAVE NOT SPECIFIED WHICH MEANING YOU INTENDED. (A TERM WITH MULTIPLE MEANINGS APPEARS MORE THAN ONCE IN THE INDEX, UNDER DIFFERENT HEADINGS; IT MAY APPEAR IN MORE THAN ONE CATEGORY OF THE ITEM UNIT RECORD,--E.G., THE TERM BRAIN MIGHT APPEAR AS BOTH A SUBJECT HEADING AND AS AN AUTHOR'S NAME.) IN THE MULTI-MEANING PM, THE DIFFERENT CONTEXTS OF OCCURRENCE OF THE MULTI-MEANING TERM ARE PRINTED OUT IN A NUMBERED LIST. YOU ARE TO ANSWER BY INSERTING THE NUMBER OF YOUR INTENDED MEANING. THE COMPOSITE FORM OF THE MULTI-MEANING PM IS:

PROG: MM (TERM), SPFY (NUMBERED CONTEXTS)---MULTI-MEANING (TERM), SPECIFY (NUMBERED CONTEXTS)---TERM (TERM) HAS MULTIPLE MEANINGS, SPECIFY BY NUMBER WHICH YOU INTEND (NUMBERED CONTEXTS).

FOR EXAMPLE, IN THE SYMBOLIC VERSION, A MULTI-MEANING PM MIGHT READ:

MM BRAIN, SPFY 1. AU, 2. SH.

THIS WOULD MEAN THAT FOR THE SEARCH STATEMENT THAT YOU HAD JUST ENTERED, THE PROGRAM HAS FOUND THE TERM (BRAIN) APPEARING IN THE LIST OF AUTHORS' LAST NAMES (AU), AND THE SUBJECT HEADINGS (SH). IF YOUR INTENDED MEANING WAS AUTHOR, YOUR RESPONSE WOULD BE:

<1><CRK>

ANOTHER SITUATION WHICH MAY RESULT IN A PM CONCERNING MULTIPLE MEANING IS THE USE OF TRUNCATION SYMBOL (#) TO INDICATE A TRUNCATED TERM. FOR EXAMPLE, ONE MIGHT PUT IN THE FIRST FOUR LETTERS OF AN AUTHOR'S NAME FOLLOWED BY #. THE SYSTEM WOULD THEN SEARCH IN ITS INDEXES POSSIBLY FINDING A NUMBER OF AUTHORS WHOSE LAST NAMES ALL BEGIN WITH THE FOUR LETTERS WHICH WERE INPUT. THE PROGRAM WILL GIVE YOU A LISTING OF THE POSSIBILITIES INHERENT IN YOUR REQUEST, JUST AS IN THE PREVIOUS MULTI-MEANING PM. YOU WOULD THEN SLEECT THE PARTICULAR MEANING, AS IN THE PREVIOUS EXAMPLE, BY TYPING THE APPROPRIATE NUMERAL, FOLLOWED BY A CARRIAGE RETURN.

THE OVERFLOW (PM)

AN OVERFLOW PM IS PRINTED WHEN THE PROGRAM'S ACTIVE SEARCH REGISTER IS COMPLETELY FILLED BY THE REQUIREMENTS OF A SEARCH. THIS CONDITION CAN RESULT FROM EITHER OF TWO CAUSES:

1. THERE MIGHT BE TOO MANY SEARCH ENTRIES IN THE STATEMENT (THE ABSOLUTE UPPER LIMIT IS 99 ENTRIES, AND THE EFFECTIVE LIMIT IS USUALLY SOMEWHAT LESS, DEPENDING ON THE KINDS OF LINKINGS SPECIFIED BETWEEN ENTRIES).
2. THERE MIGHT BE TOO MANY UNIT RECORDS IN THE FILE THAT ARE POSTED TO (I.E., CONTAIN) THE COMBINATION OF ENTRIES YOU SPECIFIED IN YOUR SEARCH STATEMENT.

THE COMPOSITE FORM OF THE OVERFLOW MESSAGE IS:

OVFLW, (ENT OR REC)---OVERFLOW, (ENTRIES OR RECORDS)---OVERFLOW OF THE SEARCH REGISTER HAS OCCURRED AS A RESULT OF TOO MANY (ENTRIES OR RECORDS).

THE SERIES FULL (PM)

THE PROGRAM PRINTS A SERIES FULL PM WHEN THE MAXIMUM OF 7 SEARCH STATEMENTS HAS BEEN INSERTED FOR ONE SEARCH SERIES, AND NO MORE SEARCH STATEMENTS CAN BE ACCEPTED FOR THAT SERIES. AT THIS POINT YOU MAY ISSUE ANY COMMAND, BUT CANNOT INSERT ANY MORE SEARCH STATEMENTS. THE COMPOSITE FORM OF THE SERIES FULL MESSAGE IS:

SF,C?---SERIES FULL, COMMAND?---YOUR PRESENT SEARCH SERIES IS FILLED TO MAXIMUM WITH SEVEN SEARCH STATEMENTS, BUT YOU MAY ISSUE ANY COMMAND.

USER ERROR (PM)
THERE ARE MANY DIFFERENT USER ERROR MESSAGES, EACH OF WHICH IS PRINTED OUT FOR THE APPROPRIATE ERROR CONDITION. THE ERROR MESSAGES ARE SELF-EXPLANATORY AND WILL NOT BE DESCRIBED HERE. THE INTRODUCTION TO ALL USER ERROR MESSAGES IS THE SAME, AND IN ITS COMPOSITE FORM IT APPEARS:

PROG: UE;---USER ERROR; (COMPUTER PRINTS EXPLANATION OF ERROR).

REFRESHER SUMMARY

HERE IS A REFRESHER SUMMARY OF THE OPERATION OF BIRCH2.

THE PROGRAM'S EXPLAIN COMMAND CAN BE USED TO OBTAIN AN EXPLANATION OF EACH OF THE ITEMS LISTED BELOW. TO ISSUE AN EXPLAIN COMMAND YOU:

1. TYPE A DOUBLE QUOTE MARK <">.
2. TYPE THE WORD <EXPLAIN>, AND ONE OR MORE SPACES < >.
3. TYPE AN <ACCEPTABLE VARIANT OF THE NAME OF ANY COMMAND> OR, THE <NAME OR SYMBOLIC VERSION OF ANY PORGRAM MESSAGE>, OR, ANY OF THE <TERMS OR PHRASES LISTED BELOW AS HAVING EXPLANATIONS>.
4. TYPE ANOTHER DOUBLE QUOTE MARK <">.
5. INSERT THE COMMAND BY STRIKING CARRIAGE RETURN KEY <CRK>.

BIRCH2 HAS ELEVEN COMMANDS. THEIR NAMES ARE:

FIND, NEIGHBOR, DIAGRAM, ERASE, ABORT, RESTART, PRINT, RENAME, VERSION, EXPLAIN, AND STOP.

THE PROGRAM HAS EIGHT ROUTINE PROGRAM MESSAGES. THEIR NAMES ARE:

READINESS CUE, CONTINUATION ACKNOWLEDGMENT, POSTING NUMBER, NO POSTINGS, MULTI-MEANING, OVERFLOW, SERIES FULL, AND USER ERROR.

THE FOLLOWING TERMS OR PHRASES CAN BE EXPLAINED:

TYPING ERRORS, UNIT RECORD, OPERATING OVERVIEW, PACING YOURSELF, TWO INPUT FORMS, FORMULATION OF COMMANDS, SEARCH STATEMENT, BOOLEAN OPERATORS, LINKING, SEARCH ENTRIES, SEARCH STATEMENT NUMBER, SEARCH SERIES, NEGATION, QUALIFICATION, LONG SEARCH STATEMENTS, FORM OF PROGRAM MESSAGES

ERROR MESSAGES:

MORE THAN SIX SYNTAX ERRORS ENCOUNTERED -- STATEMENT IGNORED.
RETRIEVAL TERM LONGER THAN 36 CHARACTERS.

TWO CONSECUTIVE LINKING WORDS NOT PERMITTED.
LEFT PARENTHESIS OUT OF PLACE.
SUPERFLUOUS RIGHT PARENTHESIS.
IMPROPER USE OF HYPHEN
MORE THAN ONE CONSECUTIVE 'NOT' FOUND.
SLASH MARK NOT BETWEEN MAIN HEADING SUBHEADING PAIR.
HASH MARK OUT OF PLACE
'NOT' FOUND OUT OF PLACE.
QUALIFIER NOT FOUND BETWEEN LEFT AND RIGHT PARENTHESES.
TWO CONSECUTIVE LEFT PARENTHESES FOUND.
RIGHT PARENTHESIS MISSING FROM QUALIFIER EXPRESSION.
SECOND DATE OF WEEK-WEEK PAIR MISSING.
QUALIFIER NAME NOT LEGAL FOR THIS USAGE.
'NOT' IS NOT PERMITTED BEFORE A QUALIFIER.
SYNTAX FRROR ENCOUNTERED.
ILLEGAL TERM FOUND AFTER QUALIFIER.
SSN SPECIFIED HAS NOT BEEN FORMULATED YET.
TOO MANY TERMS SPECIFIED FOR THIS SEARCH FORMULATION.
ONLY ONE HASH MARK IS PERMITTED IN A SEARCH TERM.
ILLEGAL USE OF SLASH.
ILLEGAL PARAMETER.
NO SEARCH STATEMENTS ENTERED YET.
SSN SPECIFIED HAS FEWER POSTINGS THAN REQUESTED TO PRINT.
SSN SPECIFIED HAS FEWER POSTINGS THAN SPECIFIED FOR.
SUM OF SKIP AND PRINT NUMBERS EXCEEDS NUMBER OF POSTINGS IN SSN.
NON-EXISTENT SSN SEPCIFIED.
SSN SPECIFIED NOT PRINTABLE BECAUSE STORED IN INVERSE FORM.
ILLEGAL COMMAND SPECIFICATION ENCOUNTERED.
SPECIFIED SSN HAS NO DOCUMENTS IN IT.
ILLEGAL ELEMENT TYPE FOUND.
ILLEGAL PARAMETERS.
RENAMED PRIMITIVE DOES NOT EXIST.
RENAMED PRIMITIVE HAS MORE THAN ONE SPECIAL CHARACTER.
'IN' ARGUMENT NOT A COMMAND NAME.
NEW NAME IS ALREADY A PRIMITIVE.
RENAMED PRIMITIVE DOES NOT EXIST.
MISSING DOUBLE-QUOTE MARK.
COMMAND NAME MISSING OR INCORRECT.
ILLEGAL SSN SPECIFIED.
TOO MANY PARAMETERS SPECIFIED IN CALL.
INCORRECT PARAMETERS.

THE ELEMENTS OF THE INPROS FILE

Element Number	Retrieval	Qualifying	Description
1	Yes	Yes	LC CARD NUMBER
*2	No	Yes	ORDER DATE
3	Yes	Yes	LC CLASS NO.
4	Yes	Yes	DEWEY CLASS NO.
5	No	No	LC PRICE
6	No	No	EDITION
*7	Yes	Yes	LIBRARY
8	Yes	Yes	AUTHOR-CONFERENCE
9	No	No	SHORT TITLE
10	No	No	TITLE STATEMENT
11	No	No	SERIES NOTE
12	No	No	PLACE OF PUBLICATION
13	Yes	Yes	PUBLICATION DATE
14	Yes	Yes	LEVEL
15	No	No	PUBLISHER
16	Yes	Yes	BOOK ORDER NO.
*17	Yes	Yes	ORDER STATUS
18	Yes	Yes	VENDOR, STANDARD
19	No	No	VENDOR, UNUSUAL
20	Yes	Yes	REQUESTOR
21	Yes	Yes	FUND
22	No	No	NUMBER OF COPIES
23	No	No	VOLUMES
24	No	No	BINDING
25	Yes	Yes	ARRIVAL DATE
26	No	No	NET PRICE
27	No	No	CALL NUMBER
28	Yes	Yes	PURCHASE ORDER NO.
29	Yes	Yes	ORDER CODE
30	No	No	VOUCHER NO.
31	No	No	ITEMS RECEIVED

Figure 1. The Elements of the INPROS File

LISTS SYSTEM

USER'S MANUAL

for

CIRCULATION CONTROL SUBSYSTEM

at

FULLERTON JUNIOR COLLEGE LIBRARY

Chapter 1 - Introduction

1.1 GENERAL INFORMATION

This manual explains how to use the Fullerton Junior College Library circulation control subsystem. Each chapter includes an explanation of a particular function as well as instructions for and examples of its use. The manual contains four chapters and an Appendix. This chapter, an introduction to the design concepts of the entire system, is intended to give the reader a general idea of the purpose and various functions of the system. Chapter 2 discusses the machine-readable book cards, which are the basis for identifying a particular book and recording any transaction about it. The actual recording of transactions and preparation of special requests are explained in Chapter 3. Chapter 4 describes the data processing functions that maintain the circulation files and prepare the various reports used by the circulation staff in the library. The Appendix is an alphabetical listing of terms that are keywords within the circulation system, with the numbers of figures and sections in the manual in which they are discussed. It is hoped that this appendix will be useful as a quick reference for guiding a user to the proper section of the manual.

1.2 OVERALL SYSTEM DESIGN

The circulation function maintains records of library materials used in the Fullerton Junior College Library. The system is centered around the use of an IBM 357 data-collection system that records each circulation transaction by accepting a machine-readable book card and producing a machine-readable transaction card. Two computer programs then process the transaction cards each night and produce various reports on the circulation status of library materials. These reports are returned early the next morning to the library staff for their use in maintaining control over the circulation of materials.

The three distinct parts of the circulation system are illustrated in Figure 1.1. The first is the preparation of machine-readable book cards for all the books in the collection. The second is the daily gathering of information representing circulation transactions or special requests. The third is the nightly processing of the data and the preparation of the circulation reports.

1.3 OPERATIONAL DESIGN FOR THE PREPARATION OF MACHINE-READABLE BOOK CARDS

The purpose of creating machine-readable book cards for the collection at Fullerton Junior College Library is to use these cards as input to the IBM 357 data-collection system in order to automatically record circulation transactions in the library. There are five general steps in the preparation of the book cards:

a. Catalog entries must be typed in an optical-character-recognition (OCR) font according to the specifications outlined in Optical Scanning Manual for Library Catalog Conversion, TM-3918.

b. The typed sheets of catalog entries must then be optically scanned and the data output on magnetic tapes.

c. The tapes containing the catalog entries are processed to produce a file of records, representing book cards for each entry on the master catalog file that is found to be in the correct form.

d. The records on the book card file are punched into 80-column cards, which are then interpreted and inserted into the back pocket of the proper book.

e. Catalog entries on the master file having errors that prevented the production of a book card record must be corrected by means of an update routine.

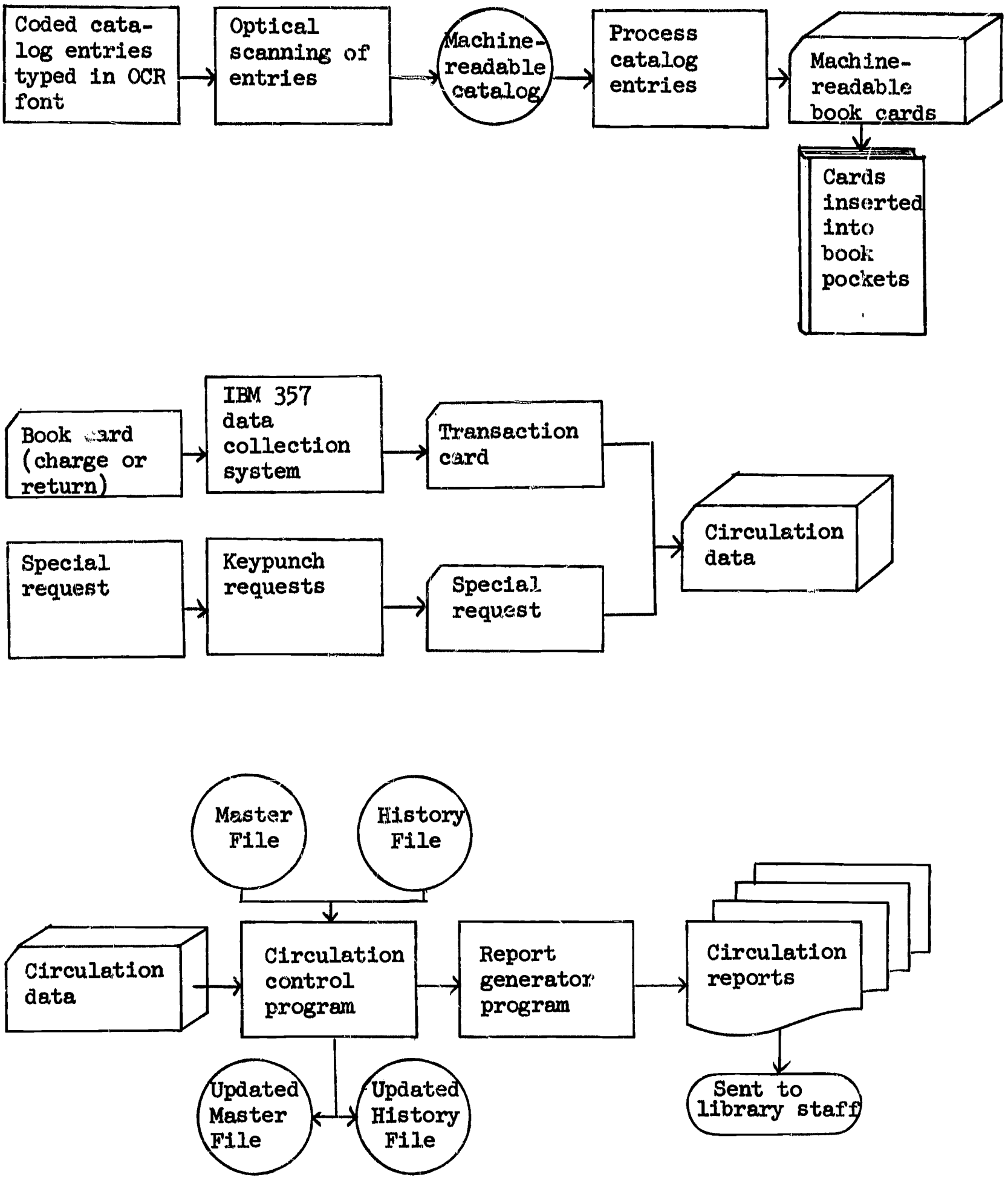


Figure 1.1. Overall System Design for the Circulation Function at Fullerton Junior College

1.4 OPERATIONAL DESIGN FOR THE COLLECTION OF CIRCULATION DATA

Two types of information must be collected in this automated circulation system. The first is that representing a transaction, which is either the charging out or discharging of a book. The second is that representing special requests such as renewals, holds, or borrower search requests. Each charge and discharge is recorded in machine-readable form, at the time the transaction occurs, by an IBM 357 data-collection system. This system automatically records the transaction for the book by accepting the book's prepunched book card as input and producing as output a punched transaction card that contains information identifying the book, the user, and the due date. Special requests are prepared by the library staff and keypunched into 80-column cards according to a preset format.

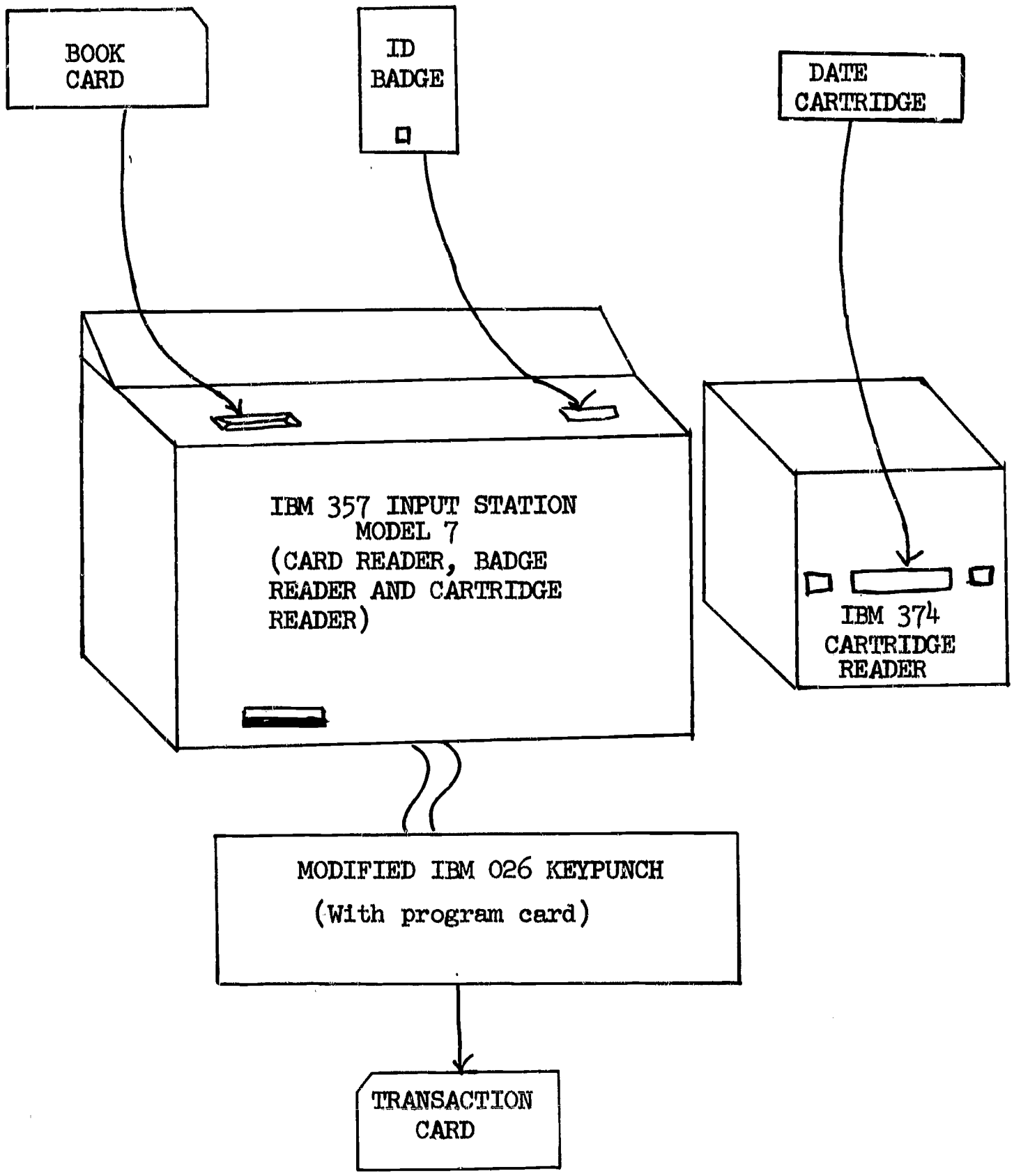
In order to charge out a book, the borrower submits the book card and his plastic identification (ID) card to the circulation attendant, who inserts the two items into the 357 input station, as shown in Figure 1.2; in addition, the proper date cartridge, reflecting either a two-week or a two-day loan period, is inserted in the 374 Cartridge Reader. There are two ways to renew a book. The first is for the borrower to submit the book card and his ID badge to the circulation attendant, who will handle the transaction in the same manner as a regular charge transaction. The second is to prepare a specially punched card containing the code R (for renewal) and the user's ID number. In order to return a book, the library attendant must insert the book card for the returned book, and a special plastic badge with the six-digit number 999999, into the 357 input station.

1.5 OPERATIONAL DESIGN FOR PROCESSING THE CIRCULATION DATA

The last phase of the circulation system is the nightly processing of the output from the data collection phase. Two computer programs form this batch-processing system for circulation control. The first processes all the transaction cards and special-request cards, updates the master circulation file and history file, and outputs data for the second program, which generates print files containing the circulation reports. These print files are then printed out in the form of reports, which are returned to the library staff.

The data processing system performs the following functions:

a. Maintains a master circulation file for all material that is currently out on loan and records the following data on each book: call number, author, title, volume, copy, date charged out, date due, borrower ID, number of overdues, number of renewals, and number of holds placed on the book while it has been charged out to the current user.



NOTE: Refer to IBM document A24-1027-3, File No. 357-09 for a complete description of the 357 Data Collection System.

Figure 1.2. IBM 357 Data Collection System

b. Maintains a history file containing a record for each completed transaction involving the charge and discharge of a book.

c. Produces the following reports (see chapter 4 for descriptions of the reports):

1) Daily Circulation Report, which lists all items out on loan in call-number order.

2) Returned-Book Report, which is a list, in call-number order, of books returned that day.

3) Overdue Notices for nonfaculty users, prepared upon request with two notices per page, which are then separated and mailed out by the library staff in window envelopes.

4) Delinquent Overdue Report, prepared at the same time as the Overdue Notices and sent to the Head Librarian.

5) Borrower Search Report, which is produced only upon request.

1.6 FUTURE DESIGN CAPABILITIES

The circulation control system is designed so that it can be expanded to allow for such capabilities as the production of a bookform catalog, the statistical processing of the history file, and borrower searching by user class.

Since all catalog entries were typed in machine-readable (OCR) form for the production of the book cards, a book catalog can easily be produced by data processing techniques from this machine-readable catalog file. All elements in the catalog are uniquely identified, so that a program could be written to search for the elements within the entry and output them in a format desirable for the book catalog. Since the entries on the master files are not in any logical order, they would need to be processed and sorted into some reasonable order--for example by call number, author, title, or subject. The entries are also coded to allow for uppercase and lowercase printing.

The history file that is maintained in the data processing phase of the system contains information on all completed transactions. A computer program could be designed and written to produce statistical results about the circulation system, using the history tape as a data base; however, another solution might be to convert the history tape to the format used by LISTS for on-line retrieval and use the facilities within LISTS for querying the data base. The contents

of the history tape would allow the following questions to be answered:

a. What is the average number of days that material is out on loan?

b. What is the average overdue period?

c. What are the averages for the number of holds, overdues, and renewals for all books?

d. How does the volume of circulation transactions vary throughout the year?

e. What is the breakdown of usage by LC classification? (This can be used by the acquisitions department in strengthening the collection in areas of dense use and for weeding the collection in areas of light use.)

f. By using the student grade records, what correlation is there, if any, between a student's use of the library and his grades?

g. Are there any subject areas that have a large number of lost books? (Lost books have a special borrower number within the class of numbers assigned to the library staff.)

h. What percentage of the students and/or faculty and/or community library members use the library?

Every library user must have a six-digit ID number assigned to him for the purpose of identifying materials he checks out. These numbers are assigned by user class so that all faculty have one group of numbers, students have another group, and library staff have yet another group. Because of this division of user groups within predetermined numerical ranges, various statistical analyses can be performed on the library users.

Chapter 2 - Producing Machine-Readable Book Cards

2.1 INTRODUCTION

The purpose of the machine-readable book card is to identify a particular book in the library collection and to allow the automatic recording of any transaction that involves the use of that book. This chapter discusses the two ways to produce a book card--from a catalog entry in machine-readable form and by manually keypunching a card. All catalog entries are initially typed in coded form, using an optical character-recognition (OCR) font. These entries are then transferred to magnetic tape, where they are processed by a computer program and checked for errors. If no errors are found in the entry that prevent making a book card, the program produces a book card (or cards, if there is more than one volume or copy) for that entry. If any errors are found, a book card is not made and the catalog entry must be corrected. Each time a new book is added to the library's collection, the catalog entry for that book must be coded, typed, and processed to get a book card for the new book. If a book card is damaged or lost, a new one must be made by manually keypunching a card that contains the information in the original card.

2.2 PREPARING NEW CATALOG ENTRIES

In order to obtain a book card for a new book, its catalog entry must be coded, typed in OCR font, and computer processed. (The coding scheme used in preparing the catalog entries for optical scanning is described in detail in Optical Scanning Manual for Library Catalog Conversion, TM-3918.) Figure 2.1 shows six sample catalog entries that have been typed according to the specifications in the optical scanning manual.

After the coded catalog entries are processed, four items are returned to the library:

a. A deck of book cards for entries that were processed correctly.

b. A list of entries found to have errors.

c. A printout of all the catalog entries and their sequence numbers on the master file.

d. A printout of all the book cards, in call-number order, showing the sequence number of the book card's catalog entry on the master file.

```
=ODK17.M49=1/MILLER, /WRIGHT /WATTS|COMP.=3/RUSSIA; A PERSONAL          (1)
ANTHOLOGY.=3LONDON|/NEWNES|{1965}%1RUSSIA↓

=OE169.L.M635=1/MILLER, /PERRY|1905-1963=2/NATURE'S NATION.=3           (2)
CAMBRIDGE|/HARVARD UNIV.|1967.%1U. S.--INTELLECTUAL LIFE%2U. S
.--RELIGION%3PHILOSOPHY, AMERICAN$7.50↓

=OTN24.C2A3 NO.169=1/NATIONAL /CONFERENCE ON /CLAYS AND /CLAY           (3)
/MINERALS=2/CLAYS AND CLAY MINERALS; PROCEEDINGS.=3WASHINGTON
|1952-%1CLAY=51/CALIFORNIA. /DIVISION OF /MINES AND /GEOLOGY↓

=OML111.N4B4=1/NATIONAL /ASSOCIATION OF /SCHOOLS OF /MUSIC. =          (4)
2/A BASIC MUSIC LIBRARY FOR SCHOOLS OFFERING UNDERGRADUATE DE
GREES IN MUCIC.=3WASHINGTON|{1967}%1MUSIC LIBRARIES%2MUSIC--
BIBL.$2.00*2↓

=OHB7293.N23=1/NASH, /WILLIAM /W=2RESIDENTIAL REHABILITATION:          (5)
PRIVATE PROFITS AND PUBLIC PURPOSES.=3NEW YORK|/MC/GRAW-/HILL
|1959.%BUILDINGS--REPAIR AND RECONSTRUCTION%2HOUSING↓

=ONC1429.N3V5=1/NAST, /THOMAS|1840-1902=1/THOMAS /NAST, POLIT          (6)
ICAL CARTOONIST.=3/ATHENS|/UNIV. OF /GEORGIA|{1967}%1NAST, TH
OMAS|1840-1902%2CARICATURES AND CARTOONS↓
```

Figure 2.1. Sample Coded Catalog Entries

COLUMNS	NO. OF CHARACTERS	CONTENTS
1	1	Number "0" - required by the 357 data collection system
2-26	25	Call number
27-28	2	Volume number (01-99)
29	1	Copy number (0-9, A-Z)
30-44	15	Author
45-62	18	Title
63	1	12-11-3 punch
64	1	12-11-1 punch
65	1	12 punch
66	1	11 punch
67	1	0 punch
68	1	9 punch
69	1	8 punch
70	1	7 punch
71	1	12-11-3 punch
72	1	12-11-0 punch
73	1	1 punch
74	1	2 punch
75	1	3 punch
76	1	4 punch
77	1	5 punch
78	1	6 punch
79	1	12-11-4 punch
80	1	(blank)

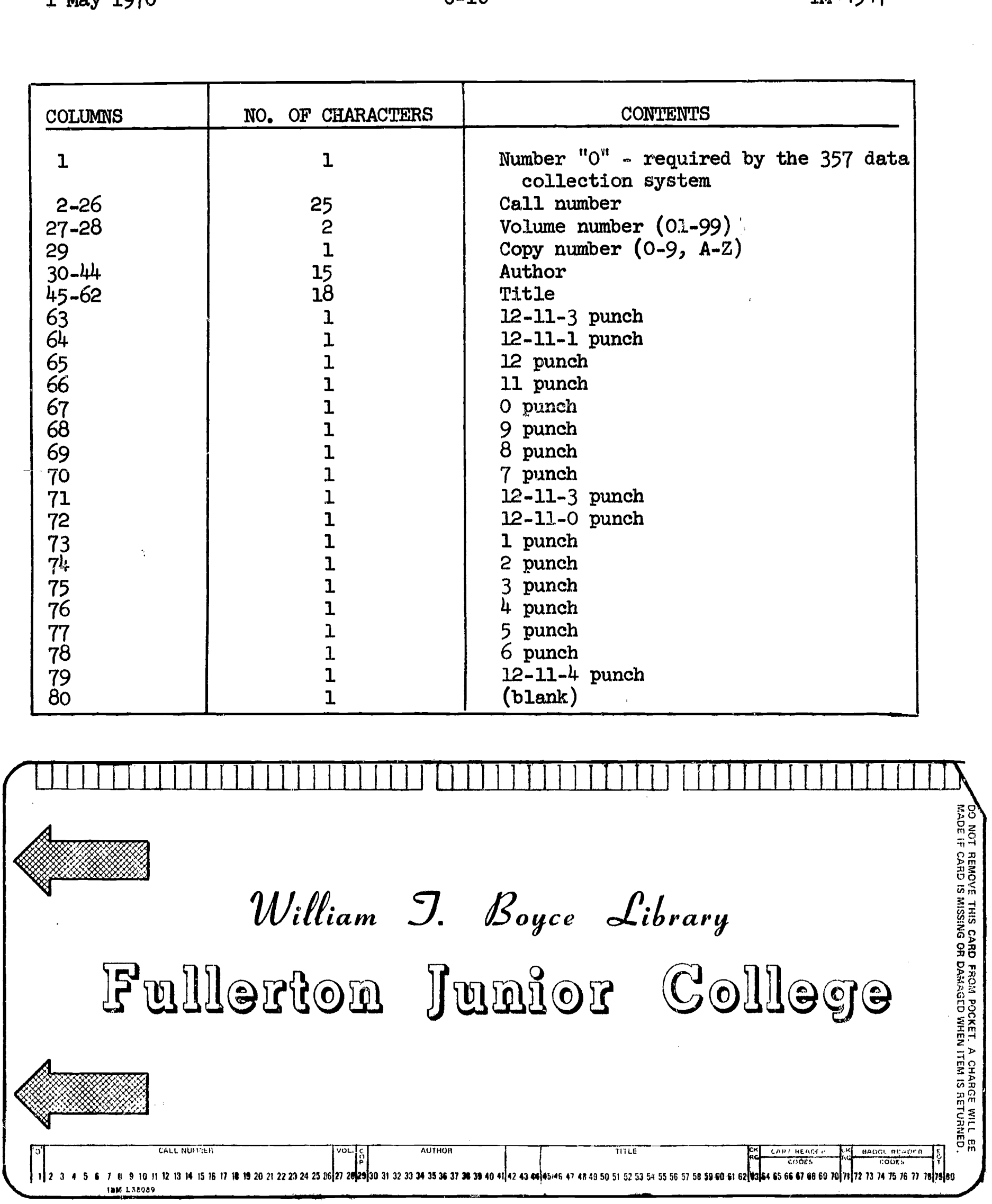


Figure 2.2. Book Card for Fullerton Junior College

Book Cards

Figure 2.2 illustrates the book card, its format, and its contents. When the book cards are interpreted, only 60 characters can be printed across the top line of the card. Figure 2.3 shows the information in the punched book card that is printed at the top of the card. In all cases, the book card must be keypunched according to the format and contents shown in Figure 2.2. Colume 80 in the book card must be left blank; all other spaces not filled with a number, letter, or special character must be filled with dashes because of the requirements of the IBM 357 data collection system.

Punched Columns	Printed Columns
2-23	1-22
27-29	23-25
30-44	27-41
45-62	43-60

Figure 2.3. Format for Interpreting Book Cards

Error Listing

The error listing output by the computer program that produces the book cards shows the complete catalog entry, its sequence number on the master file, and a message identifying the error. (Spelling mistakes and errors not affecting the production of the book card are not noted by the computer program.) In processing the sample entries in Figure 2.1, for example, the following entries would be listed as containing errors preventing the production of a book card:

- No title in entry number 1
- No title in entry number 6

Printout of Master Catalog File

The printout of all the catalog entries shows the complete entry and its sequence number on the master file. The entries on this printout are in the same order as they were optically scanned and are not in order by such entry

types as call number, author, or title. By examining the printout for the sample entries in Figure 2.1, the cataloging staff should notice the following errors, in addition to those noted by the computer program:

- The word UNIV in entry number 2 should be coded so that the U will be capitalized.
- The word MUCIC in entry number 4 should be changed to MUSIC.

Book Card Listing

The printout of book cards shows the information on the book cards and the sequence numbers of their catalog entries in the master file. In looking at this printout, the library staff can check for possible duplicate cards, which occur with fiction books whose authors and titles are so similar that the abbreviated information in the book cards does not distinguish the books. (see Figure 2.4.)

2.3 PREPARING CORRECTIONS FOR CATALOG ENTRIES

In order to correct the catalog entries that are in machine-readable form on the master tapes, the cataloging staff must prepare coded corrections. Two types of corrections may be submitted for automatic processing--the changing of information in the entry, or the deletion of the entire entry from the master file. Corrections to the master file are typed in OCR font and have the following format:

&NXyy.....y&&zz.....z ↓

where & indicates the beginning of the correction entry

N is a one-to-five-digit number corresponding to the sequence number of the catalog entry on the master file

X is a one-letter change code, "C" if the master entry is to be changed, "D" if the master entry is to be deleted

yy....y is a character string in the master catalog entry that uniquely identifies the data to be changed

&& indicates the end of the data to be changed and the beginning of the new data to be inserted into the master catalog entry

zz....zz is the character string of new data that will replace the character string yy....y

↓ indicates the end of the correction entry

Entry #1: Short Stories. Best short stories of 1968 . . . .

Entry #2: Short Stories. Best short stories of 1969 . . . .

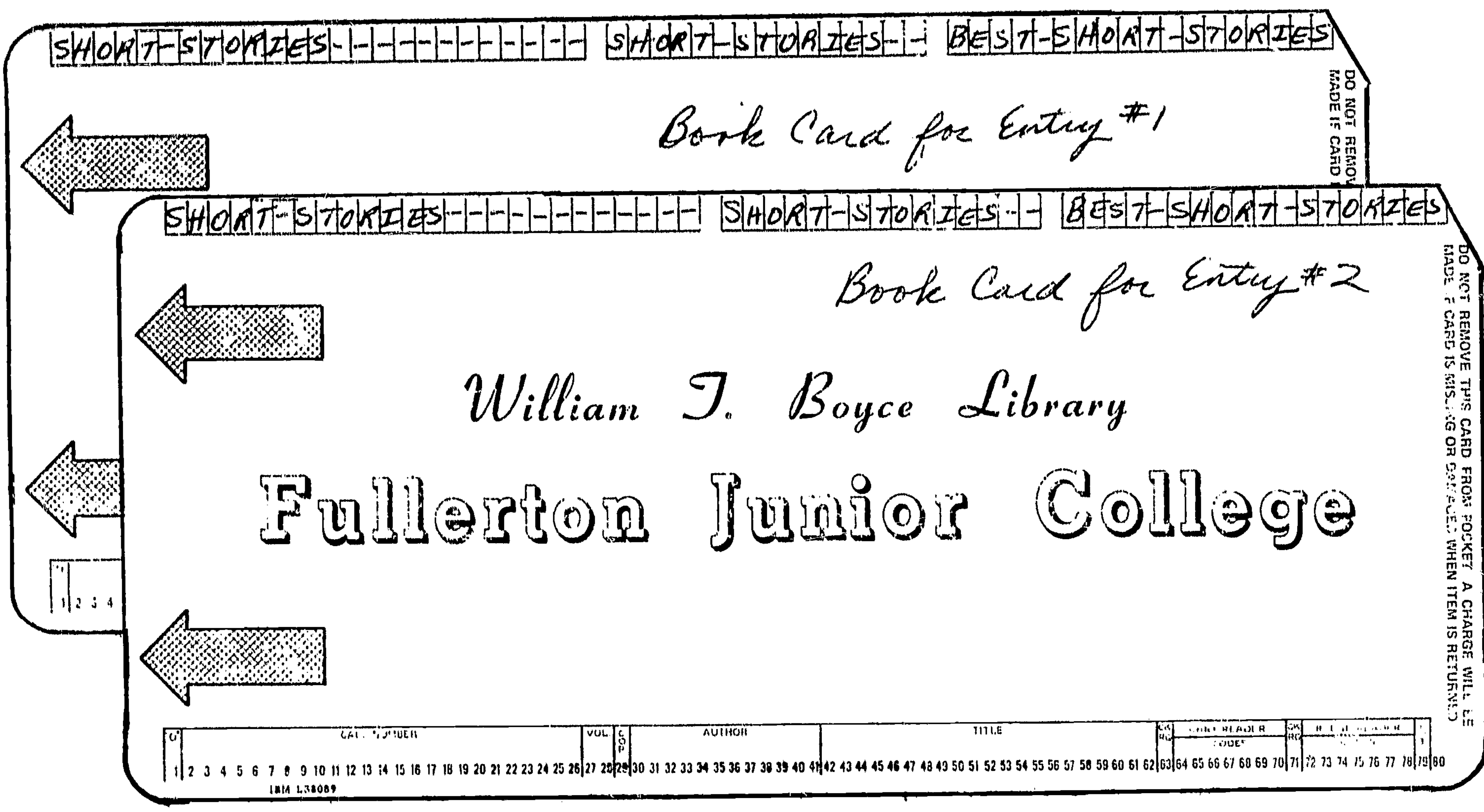


Figure 2.4. Example of Duplicate Book Cards

The corrected entry must begin at the left margin of a new line on the typed sheets and must be typed according to the same rules used in typing the original catalog entries. The sequence number of the master catalog entry can be found in two ways. First, if there was an error in the entry that prevented the book card program from making a book card record, the entry and its sequence number are printed out. The second way is to find the sequence number from a printout of the master catalog file. When all corrections for a file have been typed, they are optically scanned, output on magnetic tapes, and processed so that a new master catalog file is created reflecting the changes.

Figure 2.5 shows the coded changes that would correct the errors in the entries shown in Figure 2.1. In addition, the third line shows how the library staff could indicate there were four copies of entry number 3. If any of the original entries had been deleted, the sequencing for the entries would have changed. Therefore, if the library staff wanted to make further changes after the first set of changes had already been processed, they would have to look at a printout of the new master catalog file in order to see the new sequence numbers for the entries.

```
&1C=3/RUSSIA&&=2/RUSSIA↓                    (1)
&2CUNIV&&/UNIV↓                             (2)
&3C=1&&*4=1↓                                (3)
&4CMUCIC&&MUSIC↓                            (4)
&6C=1/THOMAS&&=2/THOMAS↓                    (5)
```

Figure 2.5. Sample Coded Changes

2.3 MANUALLY PREPARING A BOOK CARD

There are four situations in which a book card must be keypunched manually, as opposed to being produced automatically:

- Identical book cards exist for different catalog entries.
- Book cards are damaged.
- Book cards are lost.
- Book cards require special identification (e.g., 4 copies of volume 3, but 3 copies of volumes 1, 2, and 4).

The printing at the top of manually keypunched book cards will not be the same as that on cards that are computer produced and then interpreted. The manual keypunching method allows the characters in all 80 columns of the card to be printed on one line at the top of the card.

Identical or Non Unique Book Cards

In the case of identical book cards for different entries, as illustrated in Figure 2.4, the staff must manually keypunch new book cards with information that sufficiently identifies each book. For example, the information in the title field for the cards shown in Figure 2.4 could be changed from BEST-SHORT-STORIES to BEST-SH-STOR.-1968 and BEST-SH-STOR.-1969.

Damaged Book Cards

A new book card must be prepared when an existing card becomes damaged by folding, tearing, or excessive wear. If possible, the new card should be prepared from the old one by keypunching the information exactly as it appears in the old card. If the information on the card cannot be recognized, the new card must be prepared as though the old one had been lost.

Lost Book Cards

If a book card is lost, the information for a new card must be found from the master printout of book cards. This listing is in call-number order so that the person making the new card can look up the book by its call number and punch the information for that book into the new card.

Special Preparation of Book Cards

A book card that requires special punching to identify the book can be prepared by a procedure similar to that used for duplicate book cards. For example, if there are four copies of volumes 1 through 5 but five copies of

volume 6, the computer program will produce four book cards for each volume, but the book card for copy 5 of volume 6 will have to be manually keypunched. When any additional information, such as a copy-number, is to be placed in the call-number field, a slash must be punched after the last character of the call number and before the additional information. (For example, to add the information COP. 3-Vol. 6 to the call number HB10.5, punch HB10.5/COP.3-VOL.6---------.)

Chapter 3 - Collecting the Data for the Circulation Transactions and Special Requests

3.1 INTRODUCTION

Whenever someone checks out or returns a book, certain information about the user and the book must be recorded. Likewise, when a library user wants to place a hold on a book currently on loan, information about the requestor and the desired book must be saved in some manner so that when the book is returned by the present borrower, the requestor can be notified that the book is available. The process of recording and gathering this information for later processing is referred to as data collection. This chapter discusses the data collection phase of the circulation control system and explains what data are to be collected, and in what form.

The six functions that are handled by the circulation-control data processing system are discussed in each of the remaining sections of this chapter:

1. Charging a book out to a user
2. Renewing a book for a user
3. Returning a book
4. Requesting that a hold be placed on a book currently out on loan
5. Searching the circulation files for the holdings of a user
6. Requesting that overdue notices be prepared for all books currently overdue

Throughout this chapter, the term "book card" refers to the white IBM punched book card whereas "blue card" refers to the blue book card used in maintaining circulation control under the manual system.

If there is any machine trouble with the IBM 357 data collection system, contact one of the following IBM representatives:

IBM Keypunch Service Department	835-0674
Dennis Taylor Educational Institutional Consultant	521-8050
Robert Levor IBM Systems Engineer	(213) 620-1830

If there are any questions or problems regarding the circulation control system, contact the SDC LISTS monitor at (213) 393-9411, Ext. 6133 or Ext. 6933.

3.2 CHECKING OUT A BOOK

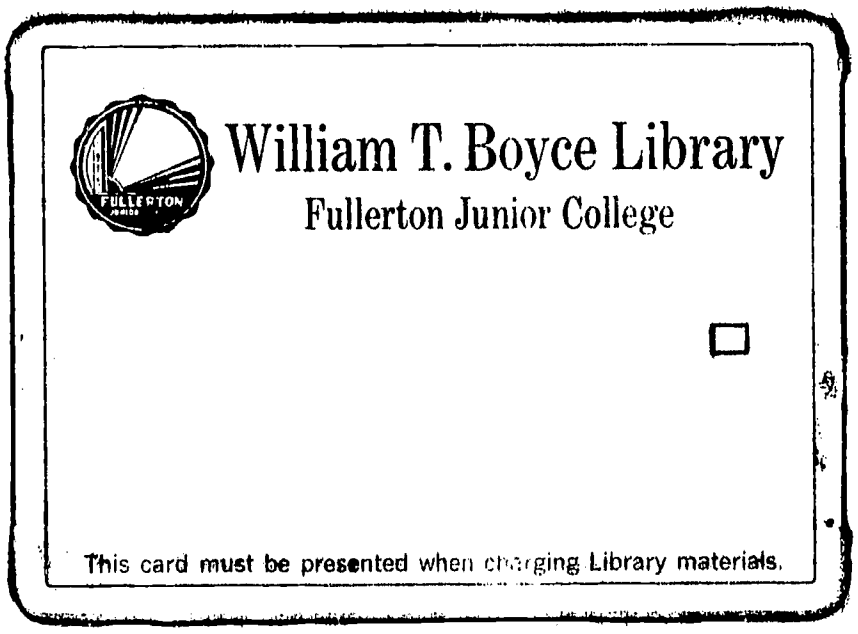
The process for checking out a book involves transferring information from the book card, the user's identification badge, and the date cartridge to a transaction card. This information is recorded automatically by the IBM 357 data collection system at the main circulation desk. The transaction card is created automatically at the output station, which is a modified keypunch machine. The user can be a regular borrower, a library staff member, or a library department. This definition of the user allows the library staff to remove books temporarily from the collection for technical processing functions such as rebinding.

In the normal checkout of a book, the borrower gives his plastic identification badge (illustrated in Figure 3.1) and books to the circulation attendant, who charges out each book in the following manner:

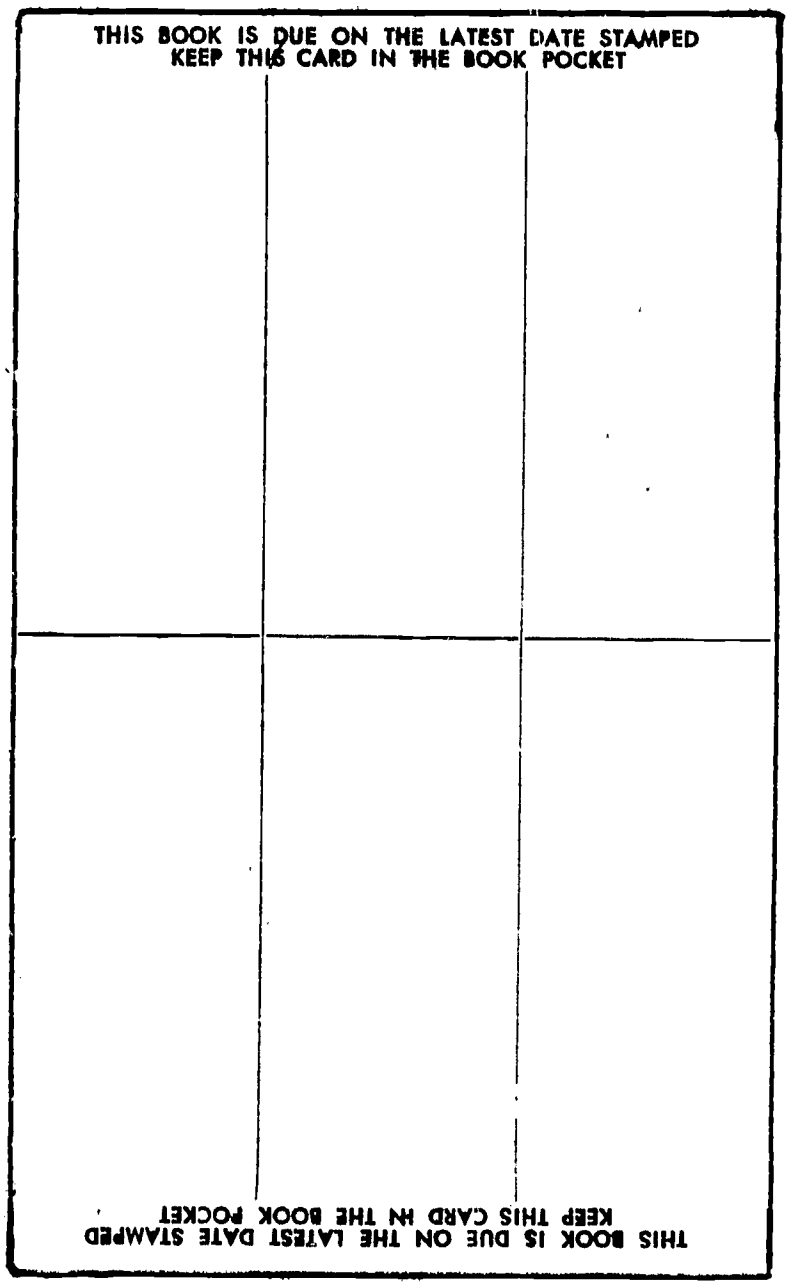
1. The book card (shown in Figure 3.2) is removed from the front pocket of the book, and the call number, volume, and copy number are checked against the book to see that the book has the correct card. (Minor spelling differences in the author and title can be disregarded in this check.) If the card reflects the proper identification for the book, the attendant can continue checking out the book by following steps 2 and 3.

   If the book does not appear to have the proper book card, the attendant should file the book card into the temporary book card file by call number. This temporary file is used to hold all book cards which have not yet been placed in the proper books. The attendant should look through this file for the book card that would correctly identify the book being checked out. If the correct book card is found, it should be removed from the file and used for checking out the book as outlined below. If the correct book card cannot be found in this temporary file, the book should be checked out using manual procedures.

2. The book card, the user's badge, and the proper date cartridge are inserted into the 357 data collection unit and kept there until the book card has been completely read by the machine and released. The book card is fed into the left-hand slot on top of the 357 input unit so that the two arrows on the left of the card face the attendant and enter the machine first. The user badge is firmly inserted in the right-hand slot of the machine

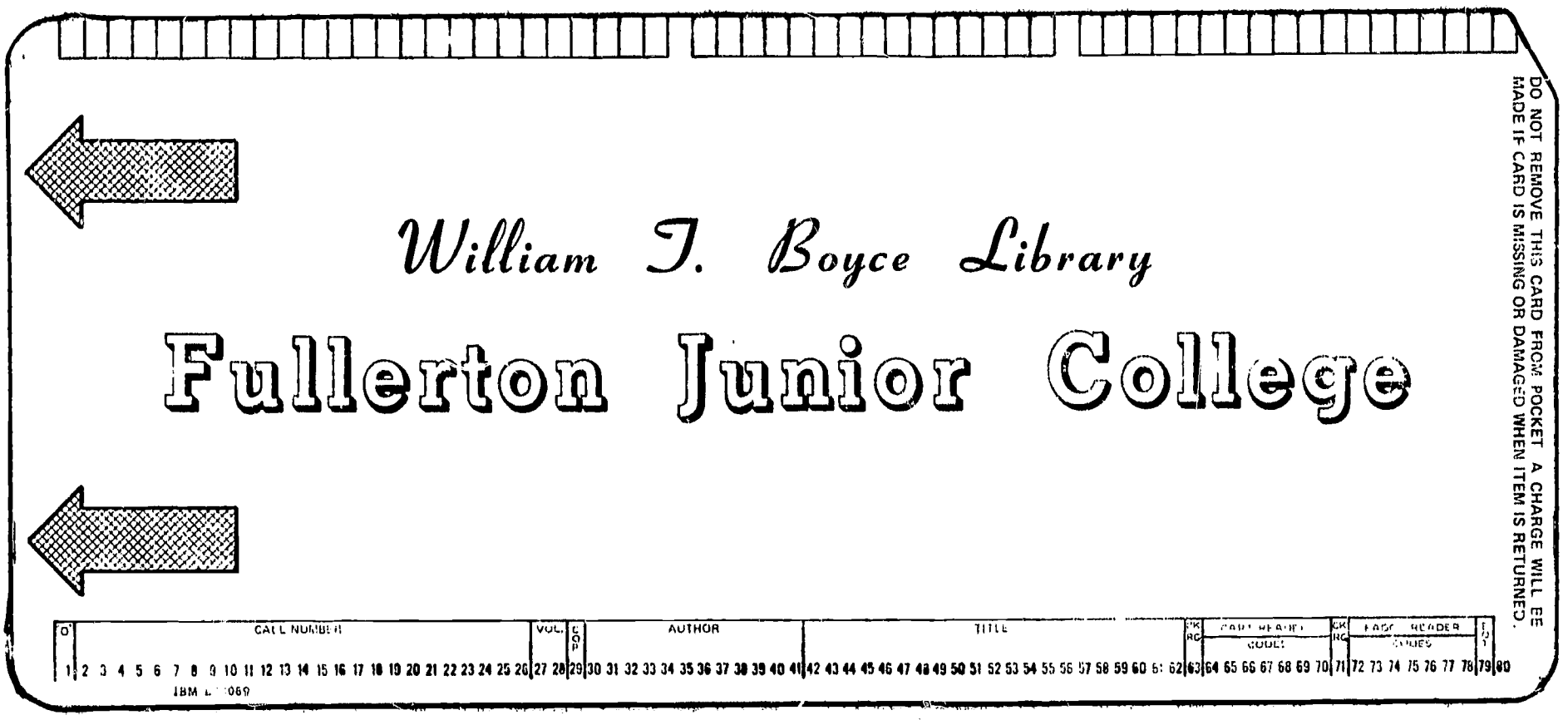


User Badge

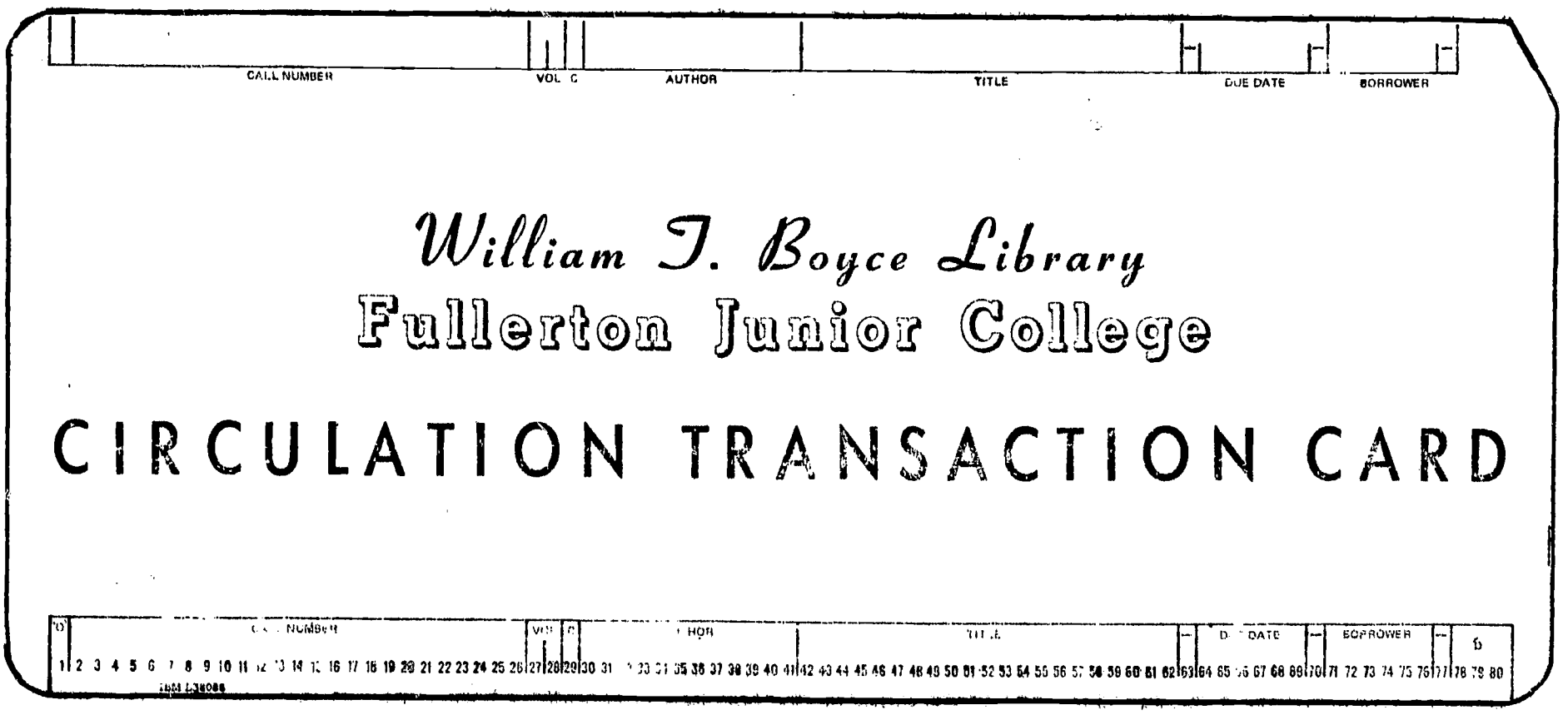


Predated-Due Card

Figure 3.1. User Badge and Predated-Due Card



Book Card



Transaction Card

Figure 3.2. Book Card and Transaction Card

with the square hole placed downward and the printed side of the card facing the attendant. There will be a "click" sound when the badge is properly inserted. If the "repeat" light on the 357 input station turns on during the transmission of the data from the book card into the transaction card, the checking out process must be repeated and the incorrect transaction card removed from the card deck. (Refer to pages 16 and 17 in the manual on the IBM 357 data collection system, Form A24-1027-3, for a full discussion of the panel lights on the input unit.)

3. The proper predated due-card (shown in Figure 3.1) is inserted into the book's pocket along with the book card, and the book is returned to the user. The orange due-cards are used for books having a two-day loan period; white due-cards are used for books having a two-week loan period. The badge is returned to the user unless he is a member of the faculty or library staff, in which case the badge is filed at the circulation desk.

If the user does not have his badge with him when checking out books, the circulation attendant should follow the steps outlined below for issuing him an identification badge.

1. Determine the user's identification number. If he has never been assigned a badge and is a student, the circulation attendant should use the identification number on the student's registration card (i.e. #6 card) as the number for the badge. If the borrower is not a student (therefore not having a matriculation number from Fullerton Junior College), he must be assigned a number according to his user class. Figure 3.4 lists the possible user classes and shows the ranges of six-digit numbers for the different types of library users (e.g., faculty, library staff, community users, etc.).

2. Prepare the badge and punch the semester identification symbol in it. The borrower's identification number must be punched into the plastic badge by using the special badge-punching machine at the circulation desk. Place the badge in the holder on the left-hand side of the machine with the large prepunched hole facing down. Push the badge into the machine as far as it will go and punch the ID number. After removing the badge from the machine, verify that the correct ID number has been punched by placing the badge over the numbered IBM card at the front desk. Then, using a special symbol for each semester, the

attendant must hand-punch the badge beyond the square hole on the right-hand side in the position designated for that particular semester. At the same time, if the user is a student, his registration card should be punched with the same punch symbol. If the borrower's registration card has already been punched for that semester, this indicates that the user has already been issued a badge. In this case, the user should be issued a 25¢ fine for the duplicate badge.

3. If the user is being issued a badge for the first time and is a nonstudent borrower, the circulation attendant must obtain his name, address, phone number, and the identification number punched on his badge, as this information is used when preparing overdue notices, delinquent reports, and borrower search reports. At the end of the day, this information should be keypunched into an IBM card according to the format shown in Figure 3.3 and kept with the other user address cards. At the end of the month, all new punched address cards would be given to the LISTS project monitor at SDC so that they can be merged with the other cards in the user address file.

In keypunching the user address card, the borrower ID number must be punched so that the units position is in column 6, the tens in column 5, etc. For example, the user number 1234 would be keypunched as 001234, beginning in the first column. However, the name, address, city, and telephone are keypunched in the card beginning at the leftmost column of the field. For example, the name HARRY ADAMS is punched so that the H is in column 7 of the card and the S is in column 17, with blanks in columns 18 through 31.

Columns	Number of Characters in the field	Contents of the Field
1-6	6	Borrower ID Number
7-31	25	Name (Forename Surname order)
32-56	25	Address (or department)
57-70	14	City (or division)
71-80	10	Phone

Figure 3.3. Format for User Address Cards

RANGE OF ID NUMBERS	MAJOR CLASS	MINOR CLASS	NUMBER OF POSSIBLE USERS
000,000-899,999	Students		900,000
900,000-929,999	Fullerton Faculty		30,000
	900,000-902,999	Humanities	3000
	903,000-905,999	Social Sciences	3000
	906,000-908,999	Fine Arts	3000
	909,000-911,999	Physical Sciences	3000
	912,000-914,999	Life Sciences	3000
	915,000-917,999	Physical Education	3000
	918,000-920,999	Business Education	3000
	921,000-923,999	Technical Education	3000
	924,000-926,999	Home Economics	3000
	927,000-928,499	Nursing Education	1500
	928,500-929,999	Mathematics & Eng.	1500
930,000-979,999	Community Users		50,000
	930,000-959,999	Adult Education Users	30,000
	960,000-979,999	Outside Community Users	20,000
980,000-984,999	Library Staff		5000
	980,000-980,999	Librarians	1000
	981,000-981,999	Library Classified Stf.	1000
	982,000-982,999	Department work	1000
	983,000-983,999	-	1000
	984,000-984,999	-	1000
985,000-989,999	Administration and Classified Staff		5000
	985,000-986,999	Fullerton Jr. College Administrators	2000
	970,000-989,999	Fullerton Classified Staff	3000
990,000-994,999	Cypress Faculty		5000
995,000-999,998	Cypress Students		5000

Note: The only preset codes are the following:

999,999	Discharge (return a book)
982,997	Unknown borrower
982,998	Temporarily missing
982,999	Lost and permanently removed from files
982,996	Two hour reserve

Figure 3.4. Ranges of Identification Numbers for Library Users at Fullerton Junior College

The date cartridge used to transmit the due date into the transaction cards is a removable cartridge the size of an 80-column card. It contains twelve data columns, each of which can be set to one of eleven positions (0 through 9, and blank). The first six leftmost columns are set to represent the due date for a charged book and must be in the form YYMMDD (e.g., 690125). There are two cartridges, one containing a date reflecting a two-day loan period and the other containing a date reflecting a two-week period. The cartridge used in checking out a book must have a due date corresponding to the loan period for that particular book.

The transaction card, shown in Figure 3.2, is automatically produced for each book card that is inserted into the data collection unit of the 357 system. This transaction card contains data from each of the three input elements--the book card, ID badge, and date cartridge. If all the input data have been transmitted correctly, the transaction card will have an x-punch in column 81 (i.e., upper right corner of the card); if any errors occur in the transmission of the data, the x-punch in column 81 will be missing. Therefore, all transaction cards without this punch in column 81 must be pulled out and corrected before being sent on for processing at the end of the day. If the user ID is missing in one of the incorrect transaction cards, the number 982997 should be put in the ID field of the new card. In this case, it is not known who has the book and the library staff will have to hope the book is returned by the unknown user.

3.3 RENEWING A BOOK

A book can be renewed in three ways. The first two depend on the book's being charged out through the IBM data collection system; the third is used when the book was charged out manually. The first is for the original borrower to check out the book through the circulation desk as though he were checking it out for the first time. The circulation attendant must verify that the number on the user's identification badge is the same number used to check out the book the first time. This can be done by looking up the call number of the book in the daily circulation report and finding the ID number for the book in question. The reason for this verification is to ensure that the original user is properly charged and credited with books he borrows. For example, if John Jones borrows a book and his friend Sam Smith tries to renew the book with his badge, and not Jones's badge, the computer program will reject the renewal request since it appears that someone new is trying to charge out a book that has not yet been returned.

The second method of renewing a book does not require using the borrower's badge or the book card of the book to be renewed. The circulation attendant looks up the book to be renewed in the current daily circulation report. The transaction number and user number for that book are then manuscripted onto the coding sheet illustrated in Figure 3.5. The letter R is placed in the first position under the heading CODE. The new due date, in the form YYMMDD (e.g., 690531), is written in the six positions under the heading RENEWED DATE.

To renew a book that was checked out manually, the book's blue card is updated to reflect the new due date.

3.4 RETURNING A BOOK

There are three cases to be considered in processing a returned book: first, the book was checked out automatically and has its book card in the front pocket; second, the book was checked out automatically but its book card has either been lost or damaged beyond use; and the book was checked out manually because it did not yet have a book card. The following steps discuss what action should be taken by the circulation attendant in each of these cases.

1. If the returned book has a book card in its front pocket, the card and the return badge (i.e., a plastic badge with the number 999999) should be inserted into the IBM 357 data collection system. This produces a transaction card with the number 999999 in the identification field of the card. The original book card is put back in the book, which can then be shelved.

2. If the book card is lost or so damaged that it cannot be used again, a new book card must be made before the book is checked back in through the data collection system.

   If the returned book does not have a book card, the attendant should check the daily circulation report to see if the book was originally checked out automatically. If the report contains the book's call number, author, and title, this indicates that the book had a book card when it was checked out, and that the card has evidently been lost since then. A new book card must be made and processed by the 357 machine, as described in Step 1, before the book is reshelved. (See Figure 2.2 for the contents of the book card.)

3. If the book was checked out manually, the attendant should pull the book's blue card, and put it in the book's pocket. The temporary book card file should be checked to see if the proper book card is in that file; if found, the card is inserted in the front pocket and the book reshelved. If the proper book card cannot be found, the attendant should take the book to the cataloging department. There, the cataloging staff should check the book against the master book card list to see if a book card was ever made for the book. If the master shelf list (i.e., master book card listing) shows a book card for the book, a new card should be punched using the information on the master shelf list. If a book card has never been made, the official catalog card should be pulled and prepared in optical scanning form as a new entry.

FULLERTON JUNIOR COLLEGE LIBRARY
CIRCULATION DEPARTMENT

CODING SHEET FOR CIRCULATION REQUESTS

Instructions: Any request for a hold, renewal, or borrower search is to be made on this sheet in the following manner.

a. Insert the proper code in the column labelled CODE. The codes are H for hold, R for renewal, and B for borrower search.
b. The transaction number is found in the daily circulation report next to the author of the book requested. This number is required only for requesting a hold or renewal.
c. The user ID is the six digit number from the borrower's plastic badge. It is needed for all three types of requests.
d. The renewal date is required only for renewals and represents the new due date. It must be in the form YYMMDD.

CODE	TRANSACTION NUMBER	USER ID	RENEWED DATE

Figure 3.5. Sample Coding Sheet for Circulation Requests

3.5 PREPARING SPECIAL REQUESTS

The following five types of requests require that the circulation attendant manuscript certain information on a coding sheet from which cards can be keypunched at the end of the day:

1. Renewing a book.
2. Requesting that a hold be placed on a loaned book.
3. Searching the master circulation files for the holdings of a particular borrower.
4. Removing a book from the collection for library staff use.
5. Requesting that overdue notices be prepared.
6. Submitting the initial set of transactions (e.g., at the start of a semester).

3.5.1 Renewing a Book

One way to renew a book is to prepare a punched card such as the one shown in Figure 3.7 entitled "SAMPLE RENEWAL CARD." On the coding sheet for circulation requests (Figure 3.5), manuscript the letter R in column 1 followed by the transaction number of the book to be renewed, the user's ID number, and the new due date in the form YYMMDD. This information is then keypunched in an 80-column card according to the format in Figure 3.6. Each time a book is renewed, the renewal indicator for the book, shown in the daily circulation report, is increased by one. Thus, it is up to the circulation assistant to look at the daily printout of the circulation files and determine whether a renewal can be made for a book. (For example, a library's policy may be to allow no more than two renewals on a book.)

Columns	Contents
1	The letter "R"
2-6	5-digit transaction number taken from Daily Circulation Report
7-12	ID number of borrower
13-18	6-digit due date in the form YYMMDD

Figure 3.6. Format for a Renewal Request Card

*INITIAL RUN — SAMPLE CONTROL CARD FOR INITIAL RUN

*PREPARE OVERDUE NOTICES 690710 X Ø

SAMPLE OVERDUE CONTROL CARD

B 100234

SAMPLE BORROWER SEARCH CARD

H12345200000

SAMPLE HOLD CARD

R00200E50000690715

SAMPLE RENEWAL CARD

Figure 3.7. Sample Special Request Cards

For example, if a library user whose ID number was 550000 wanted to renew a book appearing on the daily circulation list with the transaction number 00200 and the new due date was July 15, 1969, the attendant should manuscript the information so that it would appear on the sample renewal card as shown in Figure 3.7.

3.5.2 Placing a Hold on a Book

The hold request is used to set the hold indicator shown under the first column in the daily circulation report. A hold request is made by writing the transaction number of the requested book and the ID badge number of the requestor on the coding sheet and having this information keypunched according to the format in Figure 3.8 in order to get a card such as the sample hold card in Figure 3.7. In addition, the attendant must keep on file the name, address, and telephone number of the requestor.

Columns	Contents
1	Letter "H"
2-7	Transaction number of the book to be tagged with a hold request
8-12	ID number of requestor

Figure 3.8. Format for a Hold Request Card

Each time a hold request is made, the hold indicator in the daily circulation report for the requested transaction number is increased by one. Before reshelving any books, the circulation attendant should look at the returned-book list to see if there are any holds on returned books.

3.5.3 Requesting a Borrower Search

If the library staff wants to know what a particular borrower has out on loan, a keypunched card should be prepared according to the format shown in Figure 3.9. The computer program that processes this card will prepare a borrower search report, whose format is illustrated in Chapter 4, Figure 4.3.

Columns	Contents
1	The letter "B"
2-6	(Blanks)
7-12	ID number of borrower

Figure 3.9. Format of the Borrower Search Card

3.5.4 Library Staff Requests

The library staff and its departments can act as regular borrowers in the circulation system. Each department should have its own badge number so that a book can be removed from the shelves and checked out to a particular department. For example, if a book needs to be sent to the bindery, it can be checked out with a badge whose number is assigned to the bindery.

Two badge numbers can be used to show that a book is lost or temporarily misplaced; these codes are 982,999 for a lost book and 982,998 for a temporarily misplaced book. Thus, if a user reports that he has lost a book, the library staff should prepare a transaction card for the lost book with the identification number 982999. This will cause the record in the circulation file for that book to be removed and placed on the history file to show that the book was lost. Then, at the end of the semester, the history file can be processed to show the number of books lost during that period.

3.5.5 Requesting Overdue Notices

In order to cause the production of overdue notices and delinquent reports, the library staff must prepare a card with the format shown in Figure 3.10. This control card should be submitted only when the library staff wants overdue notices to be prepared. When submitted, it must be placed at the beginning of the deck of card input of the daily computer run.

Columns	Contents
1	Asterisk (*)
2-24	The three words--PREPARE OVERDUE NOTICES
30-35	6-digit date of the form YYMMDD
40	"X" if overdue notices for faculty are required
45	Digit for the number of overdues before adding the book to the delinquent list. If both overdue notices and addition to the delinquent list is required (e.g., at the end of a semester), then use the digit "ø."
51-55	PUNCH, if the delinquent list is to be placed on punched cards also

Figure 3.10. Format for Overdue Control Card

The date input in columns 30-35 of the control card will be used to determine whether materials are to be considered overdue. Any book whose due date is less than the date in the control card will be considered overdue and will be output on an overdue notice for the user. Thus, the library staff can decide to produce overdue notices any day of the week (rather than every Friday, for example) and has leeway in its definition of what consitutes overdue material.

3.5.6 Initial Run

For the initial run, there will be no history file or file of books charged out. Use of the *INITIAL RUN control card is necessary to forestall the program's attempting to use non-existing files. One job control card in the run deck must also be changed (DISP for the GO.HISTORY file in job step 4 should be (NEW, KEEP) rather than (MOD, KEEP)).

3.6 SUBMITTING THE DATA FOR COMPUTER PROCESSING

At the end of each day, the circulation staff must collect the transaction cards and keypunch all special requests before sending the data out to be machine processed. The following steps provide a checklist for ensuring that the data are in proper form:

1. Check all the transaction cards for the X-punch in column 81. If a card does not have this punch, remove it from the deck and look at the following card. If the following card is the correct version of the incorrect card, discard the incorrect card. If the following card is completely different from the incorrect card, then a correct card will have to be key-punched and inserted in the deck in place of the original incorrect card. If a transaction card has been produced with blanks in the borrower ID field, punch the number 982997 in columns 71 through 76 of the card. This number is used to indicate that the borrower is unknown but that the book will still be recorded as being out on loan. If a transaction card has blanks in the due-date field, punch the date in the two-week date cartridge into the date field on the card. For example, if the date on the two-week cartridge is 690815, punch 690815 into columns 64 through 69 of the transaction card.

2. Keep all cards in the order in which they were produced. Because a book can be charged out and returned by one user and then charged out the same day to another user, it is essential that the cards be kept in order so that the first user is properly credited with the return of the book.

3. If overdue notices are to be produced, the overdue control card must be placed at the front of the deck of cards.

4. Make sure that no cards are upside down or backwards in the deck.

Chapter 4: Using the Output from the Circulation Control System Programs

4.1 INTRODUCTION

At the end of each day, all the punched cards representing circulation transactions and special requests are sent to the computer center for processing. The result of this processing is a series of reports and a list of messages, which are sent back to the library. This chapter discusses these reports and messages by showing samples of their format and explaining what they mean. The five possible reports that can be output by the system are the Daily Circulation Report, the Returned Book List, Overdue Notices, the Borrower Search Report, and the Delinquent Book Report. Only the first two reports are always produced; the remaining three must be requested by means of the overdue control card (see Figure 3.11) and the borrower search request cards (see Figure 3.10). The list of messages from the computer programs contain error messages related to specific transactions and general comments related to the overall operation for that day's processing.

4.2 DAILY CIRCULATION REPORT

The purpose of the daily circulation report is to list in call-number order all materials that are currently out on loan. The format of the report is shown in Figure 4.1. The date following the title is of the form MM/DD/YY (e.g., 06/30/69) and reflects the day on which the circulation information was processed. The printed lines beneath the subheadings on each page contain information identifying a particular book that is out on loan, each line representing one book. Each page, except the first, is numbered in the upper right corner.

The various parts of the subheading are discussed below:

HOLD: The number beneath the subheading HOLD indicates the number of requests that have been made for this book by library users other than the current borrower. This position will be blank if no hold requests have been made for the book. The circulation staff should accept no more than 9 hold requests for a particular book, as there is only one digit allowed for indicating the number of requests.

CALL NUMBER: There are 25 characters under this subheading reserved for the call number of the book. This information should correspond exactly to the call number on the book card.

VOL: Under this subheading is the two-digit volume number of the book. This number should correspond exactly to the volume number on the book card.

COPY: Under this subheading is the one-character copy number of the book. This copy number should be the same as that on the book card.

TRANS NO: The transaction number is a one-to-five digit number assigned by the computer program to the circulation transactions. The only purpose of the numbers, which are reassigned each day in sequential order, is to use them in identifying particular transactions when making special requests. Since the transaction numbers change each day, it is necessary to reference the most current report to get the correct number for a circulation transaction.

AUTHOR: The 15-character author name under this subheading is the same as that in the author field of the book card.

TITLE: Under this subheading are 18 characters from the title. This title is the same as that in the title field of the book card.

BORROWER ID: The six-digit number under this subheading is the identification number from the plastic badge of the person who borrowed the book.

DUE DATE: This date, in the form month/day/year, represents the date on which the book is due back to the library.

RENEW: Under this subheading is a one-digit number that indicates the number of times the book has been renewed by its current borrower, whose ID number is under the column heading BORROWER ID. This position will be blank if the book has not been renewed by the borrower.

OVERDUES: This one-digit number represents the number of times the loaned book has been found to be overdue. This position will be blank if the book has not been found to be overdue. If the book has been found to be overdue more than 9 times, an asterisk will be in this field.

4.3 RETURNED-BOOK LIST

This report lists all books that had been on loan and were returned to the library on the day shown in the title. The report is in call-number order and has the same format as the Daily Circulation Report, as illustrated in

Figure 4.1. If a book has been checked out and returned on the same day, it will be listed in the Returned Book List, but not on the Daily Circulation Report, for that borrower; the number '00000' will be listed in the TRANS NO. column.

4.4 OVERDUE NOTICES

The overdue notices are prepared upon the request of the circulation staff and have the format illustrated in Figure 4.2. There are two notices to a page; each page must be cut in half before it can be mailed out in a left-window envelope. The dotted line, a third of the way down the page, indicates where the page should be folded. The address is placed above this dotted line, on the left, so that it will show through the window of the envelope. Below the dotted line are the call numbers and titles of the books that are overdue for the borrower. A maximum of 14 overdue books can be listed on a page; a borrower who has more than 14 overdue books will receive more than one notice.

The various parts of the notice are discussed below according to the sub-heading on the form:

TO XXXXXX:	This six-digit number is the borrower's identification number. Below this are three lines reserved for the borrower's full address. If these lines are blank or filled with 3 asterisks, the address for this user could not be found in the master address file, and the notice should be brought to the attention of the librarian in charge of circulation so that the address for the user can be looked up.
NOTICE NO:	The number in this column tells how many notices have been sent out for this particular book.
CALL NUMBER:	The 21 characters in this field are taken from the call number in the book card of the overdue book.
VOL:	Under this heading is the volume number of the overdue book.
COPY:	Under this heading is the copy number from the book card for the overdue book.
DUE MM/DD:	The number in this column represents the date on which the book was due.

4.5 BORROWER SEARCH REPORT

The Borrower Search Report is produced whenever the library staff has submitted a special request to inquire about the materials that a particular user has on loan. The format of this report is shown at the top of Figure 4.3. The contents, which are in ID order, include the names and addresses of all users for whom a borrower search was requested, plus a list of all materials that the user currently has out on loan.

The title of the report appears only on the first page; the two lines of subheadings appear at the top of every page. The list of materials lent to a user are double-spaced, and there is a triple space between the last book for one user and the name-address information for the next user. The meaning of the output below the various subheadings is as follows:

ID: This six-digit number is the user's identification number, which is taken from the plastic badge when the book is checked out.

PHONE: The telephone number for the user is taken from the registrar's file, along with the user's name and address. If the proper record for the user could not be found in the registrar's file, this field will be blank.

BORROWER NAME: Under this heading is the name of the borrower, which was found in the registrar's file. If the user's identification number could not be found in the registrar's file, three asterisks and the user's ID number will appear in this field.

BORROWER ADDRESS: The user's address and city are printed out under this heading. Blanks will be output in this field if the address information could not be found for this user's ID number in the registrar's file.

CALL NUMBER: Under this heading are the call numbers of books lent to a particular user. These call numbers are originally taken from the book cards.

VOL: The volume number from the book card of the overdue book is in this column.

COPY: The copy number from the book card of the overdue book is under this heading.

AUTHOR: Under this heading is the author of the overdue book. This information is taken from the book's book card.

TITLE: The title from the book card of the overdue book is printed under this subheading.

DUE DATE: The due date for the book, in the form MM/DD/YY, is printed under this subheading.

4.6 DELINQUENT BOOK REPORT

The Delinquent Book Report contains the name, address, and list of books for every user who has materials that have been found to be overdue more than three times. This report is produced only when overdue notices are prepared, and should be delivered to the head librarian. The format of the report is the same as that for the Borrower Search Report described in section 4.5. The report, illustrated in Figure 4.3, is in order by the user's identification number.

4.7 OUTPUT MESSAGES

The two computer programs that process circulation transactions and special requests can output informative and error messages, the latter requiring some action on the part of either the library staff or the programmer. The messages, along with their meanings and the actions they require, are listed below in alphabetical order by the computer program which prints them out.

4.7.1 Messages from the Circulation Report Program

MESSAGE	MEANING	ACTION REQUIRED
File XXXX must be printed.	Data have been output on the file XXXX and the file must therefore be printed at the end of the program.	Output the print file on the system printer by using one of the system utilities.
Following ID not found in registrar file XXXXXX YYYYYY	The ID Number YYYYYY was not found in the registrar file. The next number in the file was XXXXXX. The ID number YYYYYY, plus	The registrar file should be updated to include the name and address of this library user.

MESSAGE	MEANING	ACTION REQUIRED
	a series of asterisks, will be output in the address portion of the report.	

4.7.2 Messages from the Circulation Control Program

MESSAGE	MEANING	ACTION REQUIRED
Error in request code--not a hold or renewal XX XX	A card was included in the deck of input that did not have a zero, R, H, or B in column 1.	If the card shown by XX ..... XX represented a desired request but was coded improperly, the card should be corrected and resubmitted. If the card was not supposed to be included in the card input, nothing need be done, since the card was not processed.
Illegal transaction--cannot return a new entry XX X	The input transaction XX.....XX had the number 999999 in the borrower ID field, indicating the return of the book. However, no entry could be found on the master file for this call number, author, and title.	The card representing this transaction should be checked and resubmitted with the proper ID. The library staff should report this to the programmer so that the history of the transaction can be traced and corrected.
Illegal transaction against following record which is already out on loan XX ..... XX. Input record is YY ..... YY.	The ID in the input transaction YY .....YY was a number other than 999999, 982998, 982999, or the number of the borrower in the master record XX .....XX.	The request YY .....YY is not processed. The library staff should try to determine what action should have been taken against the master record.

MESSAGE	MEANING	ACTION REQUIRED
Too many borrower search requests made--only 100 allowed--resubmit those including and after this request XX X	The program can handle a maximum of 100 borrower search requests and therefore has rejected all those after the 100th request.	Resubmit those requests after and including the one shown in another computer run.
Too many renewal and hold requests--resubmit those including and after this request XX XX	The program can handle a maximum of 300 special renewal and hold requests and therefore has rejected all those after the 300th request.	Resubmit those requests after and including the one shown in another computer run.
Too many transaction cards--only 1000 allowed--resubmit those including and following this card XX.....XX	The system is designed to process a maximum of 1000 transactions per computer run. This limit is due to the size of the array ARRAY-LOAN.	Either resubmit those cards following and including the one shown or increase the size of ARRAY-LOAN so as to handle more than 1000 transactions at a time.
Total number of borrower search requests XXXXX	XXXXX input cards had a B in the first column.	None
Total number of renewal and holds XXXXX	XXXXX input cards had an H or R in the first position	None
Total number of transactions processed XXXXX	There were XXXXX input cards with a zero in the first column.	None

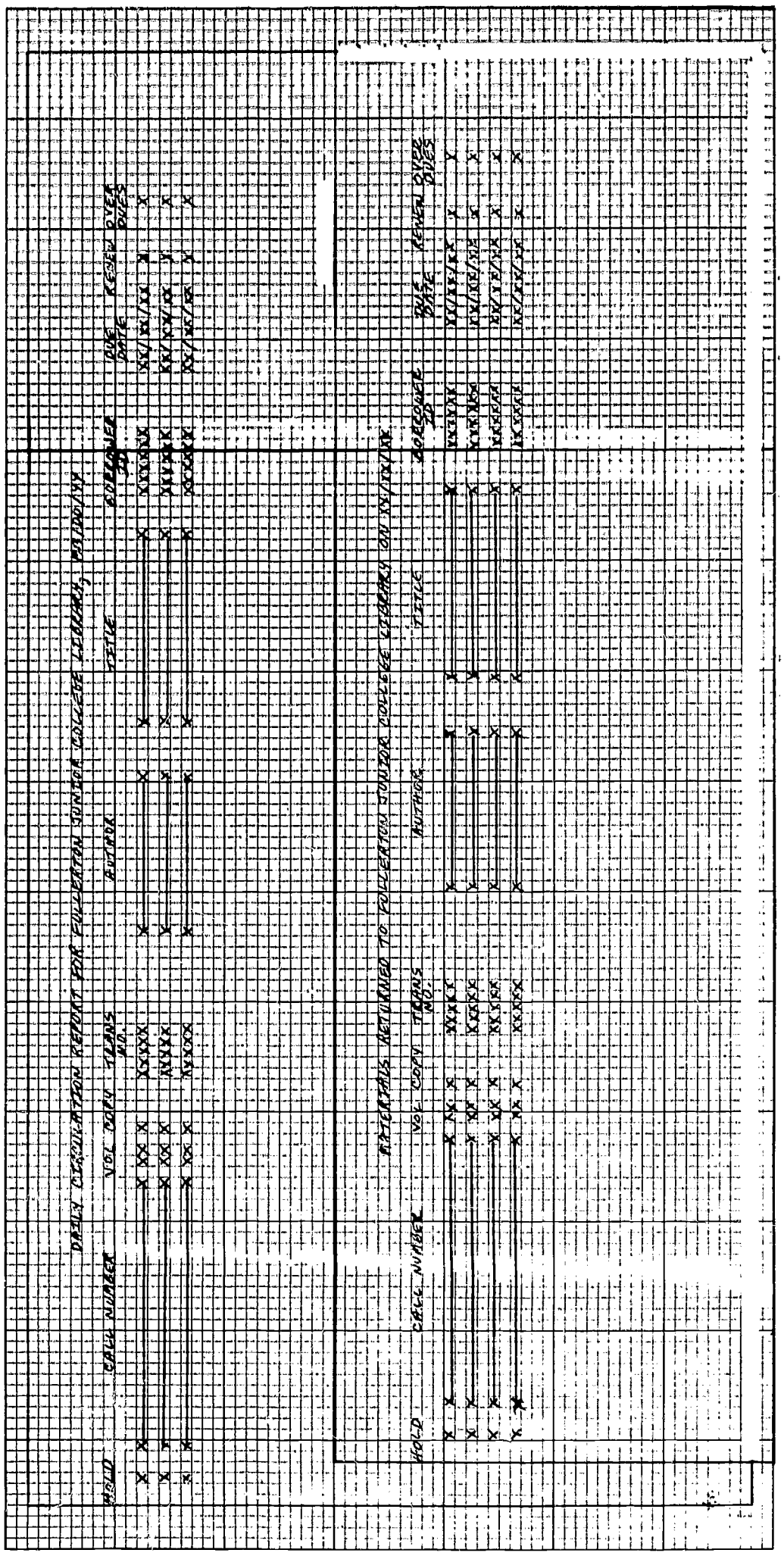


Figure 4.1. Daily Circulation Report and Returned Book List

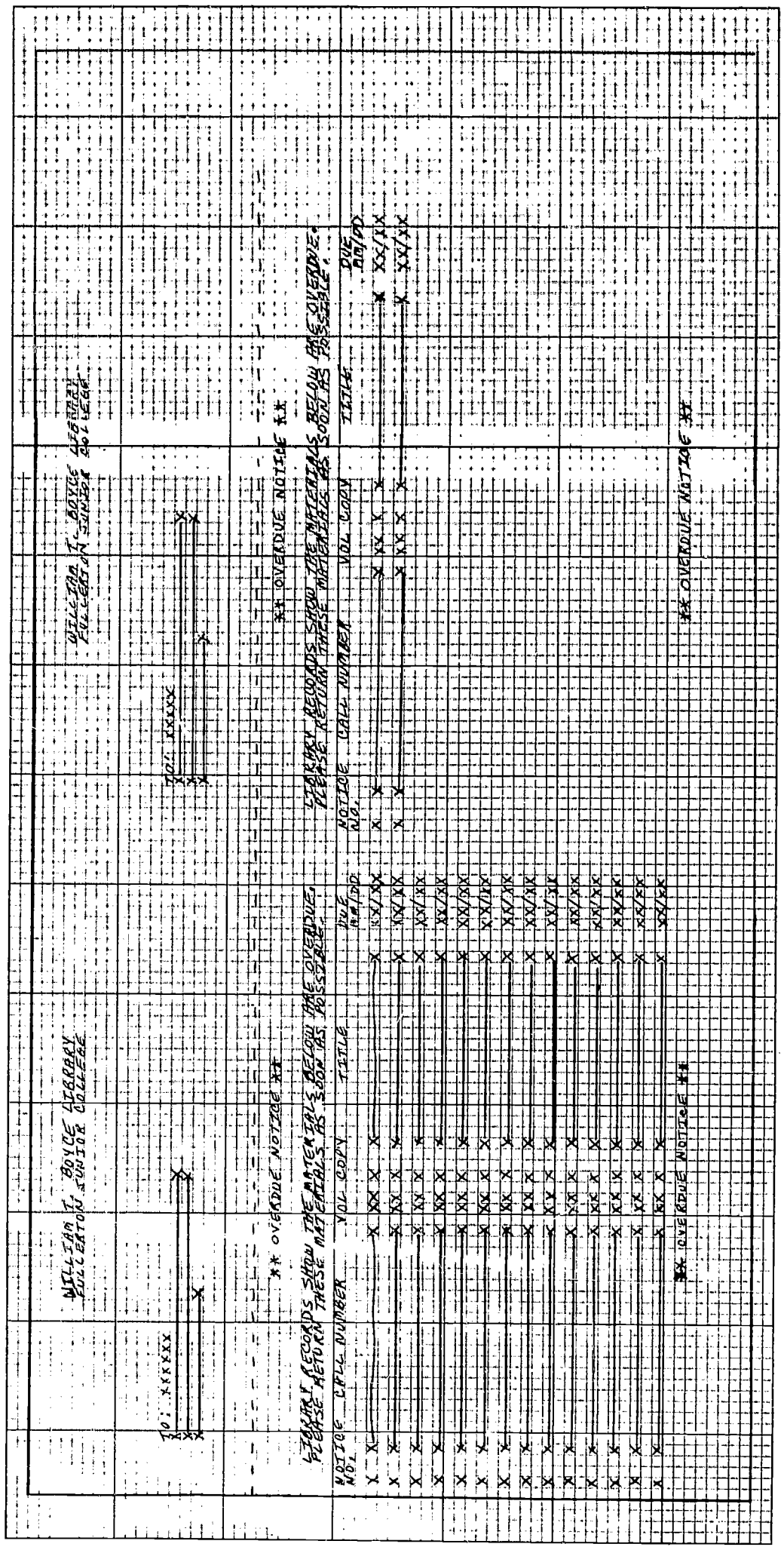


Figure 4.2. Overdue Notice

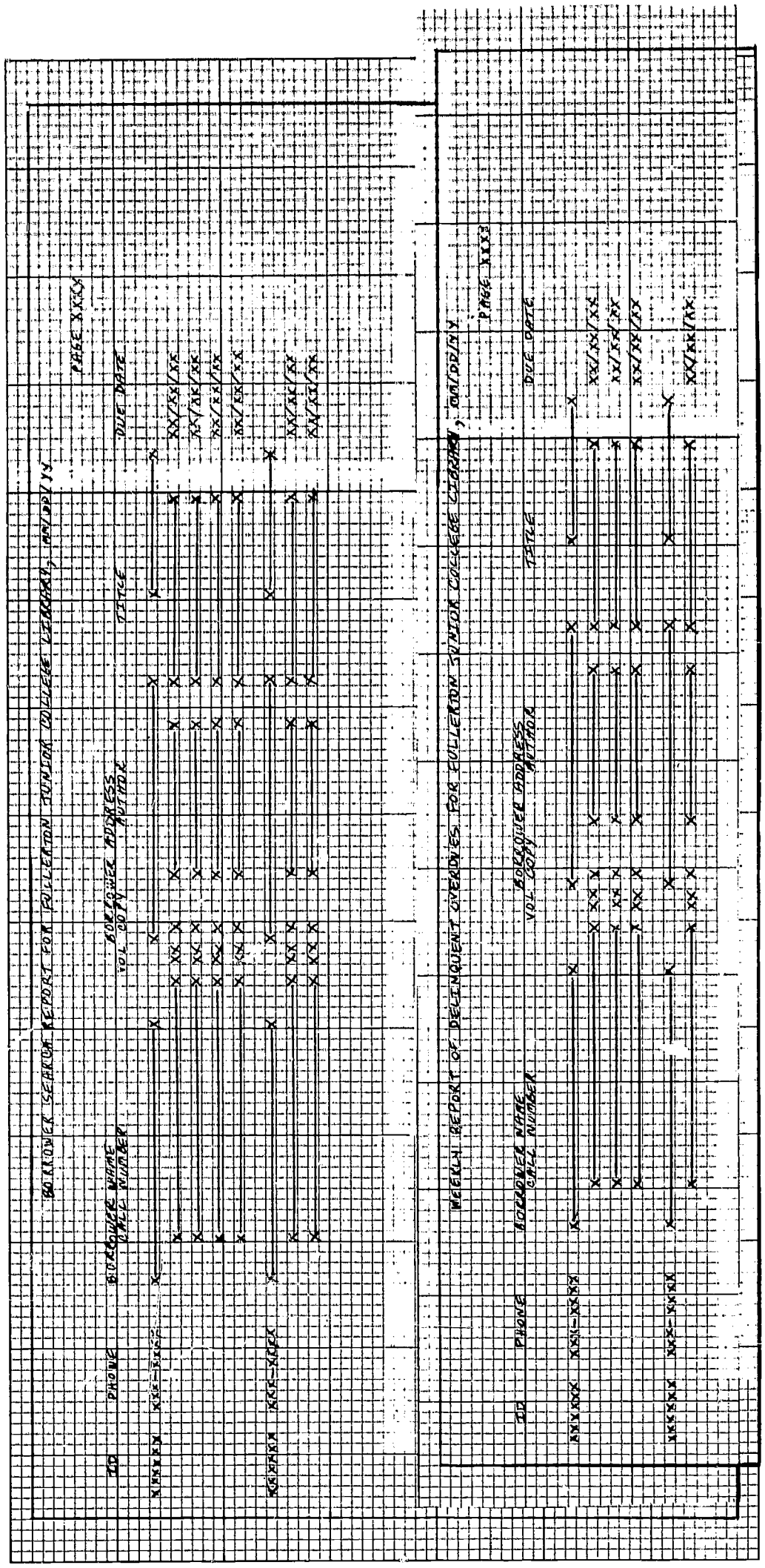


Figure 4.3. Borrower Search Report and Delinquent Report

APPENDIX:

INDEX TO KEY WORDS

APPENDIX: INDEX TO KEY WORDS

X

INSTRUCTIONS FOR OPERATING AN ON-LINE CIRCULATION SYSTEM

CHAPTER 1: NEW TRANSACTIONS

Circulation personnel who will be using this system are to use the attached experimental form for input of circulation data into the test circulation file (CIRC FILE) maintained on disc pack 0508. The information should be manuscripted on this form as the transactions occur. Once or twice a day, it should be input into the file using the GENERATE program of CDMS. A sample operation log from a terminal is included in this manual. It can be followed for the most part step by step. Particular items on the sample which are variable are so marked. They are also discussed below.

The circulation personnel should copy down pertinent information concerning loan transactions on the attached forms. The various fields of the form are explained in the following paragraphs.

FIELD 1. CALL NUMBER

This field is variable and can be any length necessary. The actual data must be preceded by 1) followed by a space. There must be a space before the next field number. In case of commas or other special characters, they must be enclosed by angle brackets (e.g., < >). It is safest to use the angle brackets wherever you are unsure as to whether a special character is restricted* or not. The left-angle bracket would be placed immediately preceding the call number and the right-angle bracket would be immediately following. The angle brackets must be preceded by a space (in the cast of the left-angle bracket) and followed by a space (in the case of the right-angle bracket). Follow the suggested formatting of the call number as exists in the sample data base (also see examples in appendix).

---

* See list of restricted characters on p. 2-21 of TM-4132/000/02A.

FIELD 3. AUTHOR

Field No. 3 also is variable. It is suggested, however, that you limit yourself to some reasonable length. Remember that commas are special characters, and to be on the safe side it is, perhaps, best to use angle brackets unless you are absolutely certain that you will not need them. It is better to enclose the entire author name in angle brackets rather than to attempt to enclose commas. (If you try the latter, there are special problems with respect to spacing.)

FIELD 4. TITLE

Field No. 4 is also variable in length. Again restrict yourself to something reasonable. It is best to leave off words such as "the", "a", and "an" at the beginning of the title. Choose descriptive terms if you shorten or rephrase the title.

FIELD 5. TOTAL TIMES BORROWED

This should be input as zero initially. When there is an existing data base, this number can be updated automatically, but it must be loaded initially with a value.

FIELD 2.

The copy identification constitutes the rest of the input record for each transaction. Please follow the spacing exactly. These fields are not variable. It is extremely important that you do not exceed the maximum length. Where an X is included, in fields 21, 22, 23, and 27, there may be either one, none, or up to the maximum number. In the remaining fields, 24, 25, and 26, there must be exactly the number of characters as indicated: no more, no less!

In the case of field 24 (due date), if the item is an indefinite loan--that is, office reference--this date, then, can become the date the item was checked out rather than due date.

It is not necessary to input information regarding the number of items renewed unless the item actually has been renewed some number of times of which you are aware.

The letter codes used in the status and location fields are explained on the form.

GENERAL CONSIDERATIONS

Please examine the examples in the appendix and observe the various formatting conventions used. Formatting is important in this program; however, once you have become accustomed to it, you will find that it is quite easy to use (I hope). These instructions will allow you to input new data only. They do not indicate how an existing record may be updated. For this information, see the chapter on Update (which will be available shortly).*

*This was never produced, because of the demise of the Technical Library.

COMMENTARY ON APPENDIX 1

Each line of the example has been numbered to aid in following. Line 1 is a command to start the program. There are other ways of starting a program that you will find in TM-4132/000/02A. At Line 5, you will make your first input after starting the program. In the example, the response is "ADDON" which will be the normal response. At Line 6, you identify the type of input which will always be "TTY." On Line 8, the system is asking you to indicate how many items you will be inputting at this particular session. Estimate a number large enough to cover whatever number of transactions you intend to input. There is no need to be cautious in estimating the number. It is better to have a larger file than to try to cut it too close and run out of space. At Line 9, you can always answer "11" for the average number of values per entry. Be careful that you do not put in two "LL's" instead of two "11's." At Line 11, input the number "1" for the average number of missing values per entry.

At Line 11, you are asked to identify the file that contains the existing data base. It is called CIRCFIL2 and it is resident on disc 0508. Disc ID is V40508, so the entire identification is as it appears on Line 11. On Line 14, the system asks for your output file identification. Simply change the digit at the end of the file name, alternating back and forth, so that the input file for one time becomes the output file the following session. That is, if you start out with input file 2 today and output file 3, tomorrow you will start with input file 3 and output file 2.

The only option necessary, as mentioned in Line 16 is the option FIX which has been input on Line 19 after the word NEXT on Line 18. The system responds with the word NEXT on Line 20, and on Line 21 the user gives the command RUN. When the system is ready for actual data input, it will print out the line number 1 followed by a double colon (1.00:) as on Line 24. Begin your data immediately following the double colon. The space following the double colon corresponds to column one on a punched card, if you were using punched cards. On Line 24, input begins. The first line of input shows a correction made. The second character should have been a right parentheses instead of the letter nine. Note how the call number is spaced without periods. Note, however, that in the next entry on line 5 of the input, i.e., Line 27 of the explanation, the call number has a decimal point which is actually part of the LC class. Note that the title field (field 4 on Line 24 of the example) would normally be "information retrieval; a critical view." It is probably just as easy to leave out punctuation, e.g., ";" and to input an extra space. In the second entry, note the angle brackets around the author. The purpose of these, of course, is to allow the comma which is a <u>restricted</u> character.

On Line 29, note that an error was made following the letter O in field 26. There was no field 27 to input in this case, so the number 2 was cancelled with a backward slash. If you are using a 2741 terminal, you simply backspace rather than using a backward slash. The backward slash is also used on Line 27 where a space was inadvertently input following QA76. The class is actually QA76.15 and there should have been no space between the 6 and the decimal point; therefore, the backward slash cancelled the space. Again, on the 2741 you would simply backspace, type the decimal point, and go right on. Note that the word "fin" for the first entry was put on a separate line, whereas in the second entry it simply followed the data by several spaces. It was not necessary to put "fin" on a separate line for the second entry since there was space at the end of the line. In the first entry there was not space within the line to allow for input. Note, however, that on a 2741, you can extend the line to 120 characters.

When you are finished with your input, type in the word DONE separated by at least one space from the word FIN. For no reason at all, the word was input on a separate line (Line 30) on the example.

At that point, if you have made no errors in your input, you have completed the process. The rest of the lines in the example are all output by the program, and there is nothing for you to do until the program has finished. However, if you have made an error that is detectable by the computer (for example, put in a zero instead of a letter o in field 26) then, at some point following Line 30 and prior to the beginning of the output on Line 32, you would have received one or more error messages depending on the errors that you made. With the FIX option operational, you could have corrected those errors. Sometimes the error messages are a little hard to understand; however, if you give the command PRINT ENTRY after the error message, the entire erroneous entry will be printed out and generally you will be able to identify the problem involved. There is also a command PRINT LINE available which will print the entire line containing the error message. If you have trouble, please call D. Black for aid.

After you have made your corrections following the rules given in TM-4132, you would input the word RUN and the program would continue if you had been successful in your error correction, the program will then proceed to run to finish. Should you be unsuccessful in correcting errors and have the program stop, or should you receive an error message which you do not understand, please call D. Black, Extension 6938 or 6933. Of course, after the program has stopped, you may log out or proceed with any other operation you wish.

APPENDIX 1

Input Example for Daily Transactions

```
 1 /GENERATE
 2 PGM.STARTED 3C000
 3 FILE LCTP9190, 2314 VOL. 9001 OPENED ON A0
 4 'GENERATE' VERSION 6.0 IS OPERATING.
 5 NEW/ADDON/RESTART:ADDON
 6 ENTER 'TTY' OR INPUT DATA FILE IDENTIFIERS:TTY
 7 FILE SMBI9190, 2314 VOL. 9002 OPENED ON A1
 8 ENTER NUMBER OF ENTRIES IN INPUT FILE(S) OR 'NO':50
 9 ENTER AVERAGE NUMBER OF VALUES PER ENTRY:11
10 ENTER AVERAGE NUMBER OF MISSING VALUES PER ENTRY:1
11 ENTER INPUT DATA BASE FILE IDENTIFIER:CIRCFIL2 V40508
12 2314 VOL 0508 TO BE MNTED ON D1
13 THE DATA BASE NAME IS 'SDC LIBRARY CIRCULATION FILE'.
14 ENTER OUTPUT DATA BASE DISC FILE IDENTIFIER:CIRCFIL3 V40508
15 FILE CIRCFIL3, 2314 VOL. 0508 OPENED ON D1
16 THE FOLLOWING OPTIONS ARE AVAILABLE BEFORE 'RUN'-
17 --PAUSE, FIX OR NOFIX, INCLUDE, ASSIGN, TRACE, AND IGNORE--
18 NEXT.
19 FIX
20 NEXT
21 RUN
22 FILE LCTP4570, 2314 VOL. 9001 OPENED ON A0
23 FILE CON34570, 2314 VOL. 9002 OPENED ON A1
24     1.00:19\) Z699 N27 1966 3) NATIONAL COLL. ON INFORMATION
25     2.00:RETRIEVAL 4) INFORMATION RETRIEVAL  A CRITICAL VIEW
26     3.00:5) 0 2) 21) 4 22) 13138 23) 6321 24) 02/03/69 25) C 26) 0
27     4.00:FIN
28     5.00:1) QA76 \.15 B84 3) <BREUER, HANS> 4) DICTIONARY FOR
29     6.00:COMPUTER LANGUAGES 5) 0 2) 21) 2 22) 13138 23) 6321
30     7.00:24) 04/05/68 25) C 26) 0 2\  FIN
31     8.00:DONE

32 FILE NAME4570, 2314 VOL. 9004 OPENED ON A3
33 FILE DATA4570, 2314 VOL. 9002 OPENED ON A1
34 FILE FAIL4570, 2314 VOL. 9004 OPENED ON A3
35 FILE CON14570, 2314 VOL. 9001 OPENED ON A0
36 FILE CON24570, 2314 VOL. 9003 OPENED ON A2
37 FILE LDTP4570, 2314 VOL. 9004 OPENED ON A3
38 FILE ERDA4570, 2314 VOL. 9003 OPENED ON A2
39 '2' ENTRIES PROCESSED.
40 INITIAL PROCESSING OF 2 ENTRIES IS COMPLETE.
41 FILE CN1T4570, 2314 VOL. 9003 OPENED ON A2
42 FILE CNT14570, 2314 VOL. 9001 OPENED ON A0
43 FILE FLST4570, 2314 VOL. 9001 OPENED ON A0
44 FILE CN2T4570, 2314 VOL. 9002 OPENED ON A1
45 FILE CNT24570, 2314 VOL. 9002 OPENED ON A1
46 FILE CN3T4570, 2314 VOL. 9004 OPENED ON A3
47 FILE CENT4570, 2314 VOL. 9003 OPENED ON A2
48 FILE VALU4570, 2314 VOL. 9004 OPENED ON A3
49 THE GENERATE OPERATION HAS FINISHED PROCESSING
50 PGM.STOPPED
```

APPENDIX 2

Circulation Input Form

Part A

CIRCULATION INPUT FORM

Call #	Author	Title
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______
1) ______ ---	2) ______ ---	4) ______

Part B

	Total Times Borrowed	Copy #	Borrower ID (Man #)	Org. Code	Due Date	Status (C, D, H, R, T)	Location (B, M, O)	Times Renewed
______ ___	5) 0 2)	21) XX	22) XXXXX	23) XXXXXX	24) XX/XX/XX	25) X	26) X	27) XXX
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___
______ ___	5) 0 2)	21) __	22) _____	23) ______	24) __/__/__	25) _	26) _	27) ___

C = Charge
D = Discharge
H = Hold
R = Renewal
T = Transfer
B = Branch
M = Main
O = Office reference

LISTS EVALUATION QUESTIONNAIRE

PART I - INFORMATION ABOUT THE USER

1. What is the title of your job?

2. Which operations of LISTS did you use?

   Searching _____

   Bibliographies_____

   Book orders_____

   Other (specify)_____

3. For how many years have you been employed

   a. at this library?_____

   b. at other libraries?_____

4. What is your level of education?

   High school_____

   Secretarial school_____

   B.A. (specify major)_____

   M.L.S._____

   Other (specify)_____

5. If you have had any previous experience with computers, please describe it.

PART II - OPERATIONS AND PRODUCTS

Instructions on answering Part II of the questionnaire.

In this part of the questionnaire, you are being asked to evaluate the LISTS, Library Information System Time-Sharing, project on the basis of the operations it performs, and the products it produces. Included below are suggestions to help formulate your evaluation or generate comments.

1. What are the strengths of this operation or product and what do I like about it?

2. What are the weaknesses of this operation or product and what do I not like about it?

3. What changes could be made to this operation or product that would make it more satisfactory to me?

4. Was each particular operation easy or too difficult to use?

5. Was the operation useful?

OPERATIONS	Never Used	Favorably Impressed	Satis-factory	Needs Im-provement	Severe Problem	No Opinion
A. General Procedures:						
1. Dialing the system						
2. Time-Sharing system commands						
a. LOGIN						
b. LOAD and GO						
3. Redialing when disconnected						
4. Terminating LISTS operation						
5. Correcting typing errors						
6. Pacing yourself						
7. Input form						
a. answer						
b. command						
Comments:						
B. Retrieval:						
1. Conversation mode						
a. symbolic version						
b. short version						
c. long version						
2. Boolean operators						
3. Multiple linking						
4. Search entries						
a. number of retrieval elements						

	Never Used	Favorably Impressed	Satis-factory	Needs Im-provement	Severe Problem	No Opinion
b. length of retrieval elements						
c. qualifying elements						
5. Search statement number						
6. Search series						
7. Long search statements						
8. Overflow						

Comments:

9. Commands						
a. Find						
b. Neighbor						
c. Diagram						
d. Erase						
e. Abort						
f. Print						
g. Explain						
h. Rename						
i. Version						
j. Restart						
k. Stop						

Comments:

	Never Used	Favorably Impressed	Satis-factory	Needs Im-provement	Severe Problem	No Opinion
C. Book Order Procedures:						
1. Long form ordering	____	____	____	____	____	____
2. Short form ordering	____	____	____	____	____	____
3. Updating INPROS file	____	____	____	____	____	____
4. Status report	____	____	____	____	____	____
Comments:						
5. Products						
a. Special bibliographies printed off-line	____	____	____	____	____	____
b. Shorter bibliographies printed on-line	____	____	____	____	____	____
c. Catalog cards	____	____	____	____	____	____
d. LC catalog card orders	____	____	____	____	____	____
e. Book order forms	____	____	____	____	____	____
f. Claim letters	____	____	____	____	____	____
g. Accounting reports	____	____	____	____	____	____

Comments:

PART III - QUESTIONS

1. LISTS has been an experiment. Would you like to use it permanently?

___yes ___no

Why?

2. Do you feel that you have a good comprehension of what the system is supposed to do?

___yes ___no

Why?

3. What more than anything else aided this comprehension?

___personal contact
___User's manual
___actual terminal experience
___other (specify)

Why?

4. Do you think it adequately does what it is supposed to do?

___yes ___no

Why?

5. Have the hours of operation, 9 to 1, 4 hours a day been sufficient?

___yes ___no

Why?

6. If you feel more hours are needed, how many more? _____

7. Would you prefer the hours of operation to be other than in the morning, and if so, when? _____

8. Have you been bothered or held up by technical failure in the following areas, and if so, to how great an extent?

   a. computer

   b. teletype terminal

   c. telephone

9. Are there any elements in the MARC record or in the INPROS file you would like to use for retrieval but cannot at present? _____

   If so, what are they?

10. Please rank the operations and products in order of greatest importance to you.

Operations:
___Book ordering
___Retrieval
___Other

Products:
___On-line bibliographies
___Off-line bibliographies
___Catalog cards
___LC catalog card orders
___Book order forms
___Claim letters
___Accounting reports

11. Are there any capabilities of the LISTS system which you do not use?

    If so, why?

12. What operations or products would you like that you do not have?

13. How long do you estimate that it took before you felt comfortable using LISTS?

14. Approximately how much time have you spent demonstrating LISTS to:

    _____Board members

    _____Faculty

    _____Patrons

    _____Visitors

    _____Others (specify)

15. Do you feel that you save time using LISTS? _____

    If so, how much and in what areas?

16. Are there certain operations that perhaps take more time using LISTS?

    _____. If so, which ones, and how?

17. How has LISTS affected other operations in your library in terms of:

    Personnel requirements

    Job Skills

    Traditional procedures

18. Additional comments: