AN INFORMATION STORAGE AND RETRIEVAL CAPABILITY

Preliminary Manual

JEANNE M. FINKE and EARL B. HUNT

AFOSR 69-1792TR

This research was sponsored by the Air Force Office of Scientific Research, Office of Aerospace Research, United States Air Force, under AFOSR Grants Numbers AFOSR-13-1167 and AFOSR-69-1701. Distribution of this document is unlimited.

1. This document has been approved for public release and sale; its distribution is unlimited.

Computer Science Group and
Department of Psychology
University of Washington--Seattle
Computer Science Technical Report No. 69-04-06
April 9, 1969

1.0 General Information

This report describes two computer programs, STORE and RETRIEVE. The programs written in B5500 Extended Algol, provide a limited storage and retrieval capability. Sections 2.0 and 3.0 of this report describe the respective programs in detail. Section 4.0 describes operating information for the current versions of the programs.

The programs described here are preliminary versions of an information management system being developed as part of a study of computer aids to human problem solving. A more general system will be announced and documented subsequently. This document is being issued to aid research groups who can make use of the preliminary system.

It is hoped that the data files prepared under the preliminary system will be compatible with the file management routines of the final system, but this cannot now be guaranteed

The manual is written at two levels. Starred sections (*) are concerned with the internal mechanics of the system. Unstarred sections describe how to use the programs. A user need be familiar only with the unstarred sections.

Both this system and the projected generalized system make use of the programming techniques and procedures developed by Kildall (1969). Familiarity with this reference is assumed in the discussion of the starred sections.

2.0 The STORE Program

STORE accepts input data on-line directly from a remote terminal or in card image form queued as part of a B5500 packet. This input consists of:

A. file specification

B. sets of data elements (character strings) and their delimiters.

STORE initializes the specified disk file for storage, interprets the bounds of each input set, and interprets the delimiters which identify each data element within a set. The data elements of each set are stored temporarily in a variable length array. When all the elements of an input set have been recognized, the enitre data set is placed permanently in disk storage as a variable length character string. In addition, other fixed length and variable length character strings, forming indexes to the data sets, are also stored permanently in disk storage.

2.1 Input to STORE

There are no column restrictions for data input to the STORE program.

2.1.1 File Specification

The following delimiter identifies the name of a disk file into which data sets will be stored:

ex: /FILE filename

filename is a series of alphanumeric characters whose length is less than or equal to 7; filename normally will be the user's job number.

2.1.2. Data Sets

The following delimiters identify the bounds of a data set and the elements

within a set. Although the example below corresponds to data sets describing books or documents, a data set maybe any user defined entity. In the generalized version to be released later, the user will be able to specify his own delimiters.

Delimiter	Function
/DSET	identifies beginning of a new data set
$R	identifies the reference, i.e., the name of the person who input the data set
$T	identifies the title of a document
$A	identifies the author of a document
$D	identifies the date of a doucment
$S	identifies the source of a doucment, e.g., journal source
$C	identifies the contents or abstract of a document
/END	identifies the end of the last data set input to STORE

2.1.3 Input Limitations

A. The file specification must precede any data set input.

B. Each delimiter must be followed by at least one space, except /END. This is necessary to distinguish a delimiter from the actual character string.

C. Each data set must contain at least one data element and its delimiter.

D. Each line of input must be terminated by a group mark, i.e., "←", when input is entered via remote terminal. The group mark should only be typed after at least one character "grouping" of the element is typed, e.g.,

/DSET ←	is incorrect
/DSET $T ←	is incorrect
/DSET $D NOVEMBER ←	is acceptable
/DSET $D NOVEMBER 10 ←	is acceptable
/DSET $D NOVEMBER 10, ←	is acceptable
/DSET $D NOVEMBER 10, 1968 ←	is acceptable

E. The size of a single data set should not exceed 54 (full) lines.

F. The number of data sets in which a unique $R or $A element may appear is equal to 500.

2.2 The Storage Process

The following information describes the manner in which data is stored and manipulated by the computing system.

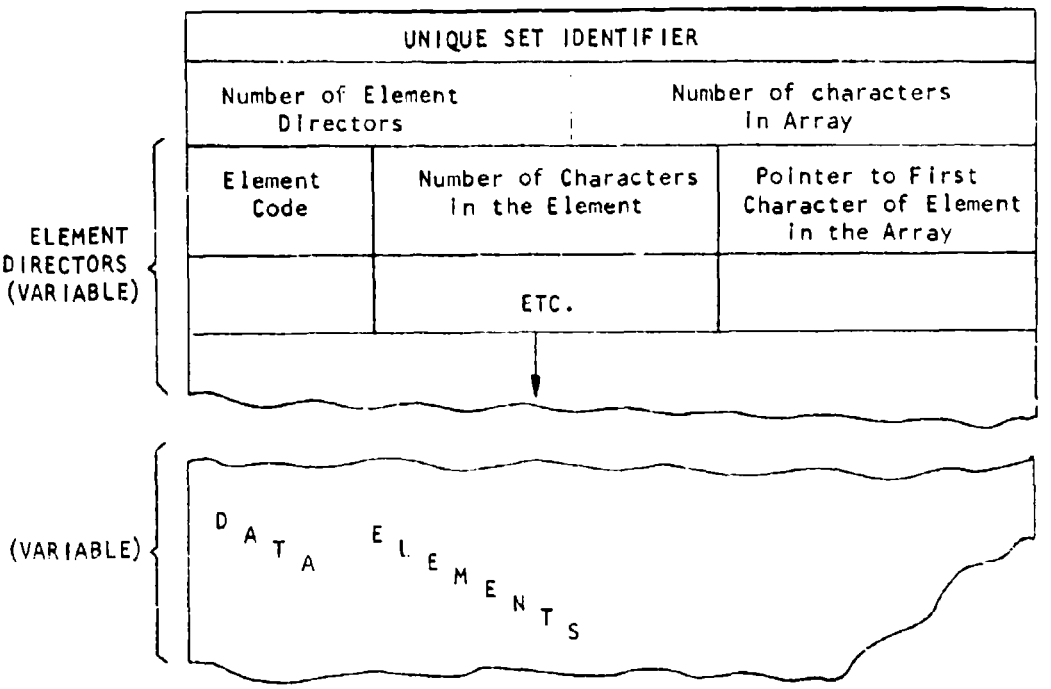
There are two types of disk output from STORE: sequential and ordered. Disk storage Units 1 and 2 are designated as sequential storage. Units 3, 4, and 5 are designated as ordered storage. These units are described below in a logical order rather than numerical.

2.2.1 Sequential Storage Unit 1

Unit 1 contains variable length character strings which contain all the elements input for a particular data set. As each data set is placed in Unit 1, it is assigned a unique set identifier of 8 characters. The unique set identifier is a sequentially assigned integer.

The array format, from which the variable length character string is written, is described below. The maximum array size is currently 4000 characters.

2.2.2 Relation of Set Identifiers to Data Elements



When the first /DSET delimiter is encountered in the input stream, an array is initialized for the data set.

Element directors are formed and stored in the same order as the particular element and its delimiter are encountered in the input stream.

The element codes corresponding to the external delimiters are:

Internal Element Code	External Delimiter
01	$R
02	$T
03	$A
04	$D
05	$S
06	$C

The pointers are relative to character location 0 of the array.

When successive /DSET delimiters are encountered, the STORE program completes its work in the current array for the previous data set and the variable length character string is placed in sequential storage Unit 1. An array is then initialized for the new data set. This process continues until the /END delimiter is encountered. The STORE program then completes its work in the current array for the last data set and places the variable length character string in Unit 1. The STORE program then proceeds with its final "housekeeping".

Each time a data set is placed in Unit 1, the unique set identifier for the data set and the relative location of the data set in Unit 1 are stored in ordered storage Unit 3. In addition, before the program terminates, the last unique set identifier assigned to a data set is stored in Unit 1, and its

relative location is stored in ordered storage Unit 3 together with a special identifier key of all zeroes. See section 2.2.2.

2.2.3 Ordered Storage Unit 3

Unit 3 contains unique indexes of length 16 characters, each called a Set Identifier Index.

Format

UNIQUE SET IDENTIFIER	RELATIVE LOCATION OF DATA SET IN UNIT 1

characters → 0 7 8 15

As mentioned above in section 2.2.1, there is a 1 to 1 correspondence for data sets stored in Unit 1 and set identifier indexes.

These indexes are maintained in ascending order on the key characters 0 through 7.

The set identifier index with key characters equal all zeroes contains in its right half (characters 8-15) the relative location of a character string in Unit 1 containing the last unique set identifier used. This index is always accessed during initialization of the STORE program.

2.2.4 Ordered Storage Unit 4

Unit 4 contains unique indexes of length 16 characters, each called a Reference Index, corresponding to $R elements.

Format

First 8 characters of a Reference Element	Relative location of Set Identifier character string in Unit 2

characters → 0 7 8 15

As elements with $R delimiters are encountered in the input stream, the first 8 characters of the element (packed, left-adjusted), constituting the key characters, are binary searched through the Reference Indexes.

If the key characters of the reference element are not found in the Reference Indexes, the set identifier for the data set (to which the reference element belongs) is stored as a character string in sequential storage Unit 2. The key characters of the reference element and the relative location of the set identifier string in Unit 2 are then placed in ordered storage Unit 4.

If the key characters of the reference element are found in the Reference Indexes, the set identifier character string is retrieved from Unit 2, and the new set identifier is added to the character string. The old string is deleted from Unit 2 and the new consolidated string is added to Unit 2. The relative location of the new string in Unit 2 is stored with the key characters for the reference element in Unit 3.

These indexes are maintained in ascending order on the key characters 0 through 7.

2.2.5 Ordered Storage Unit 5

Unit 5 contains unique indexes of length 16 characters, each called an Author Index, corresponding to $A elements.

Format

	First 8 characters of Author Element	Relative location of Set Identifier character string in Unit 2
characters →	0 7	8 15

As elements with $A delimiters are encountered in the input stream, the first 8 characters of the element (packed, left-adjusted), constituting the key characters, are binary searched through the Author Indexes.

The same storage procedure, as described above in section 2.2.4, is followed.

2.2.6 Ordered Storage Unit 2

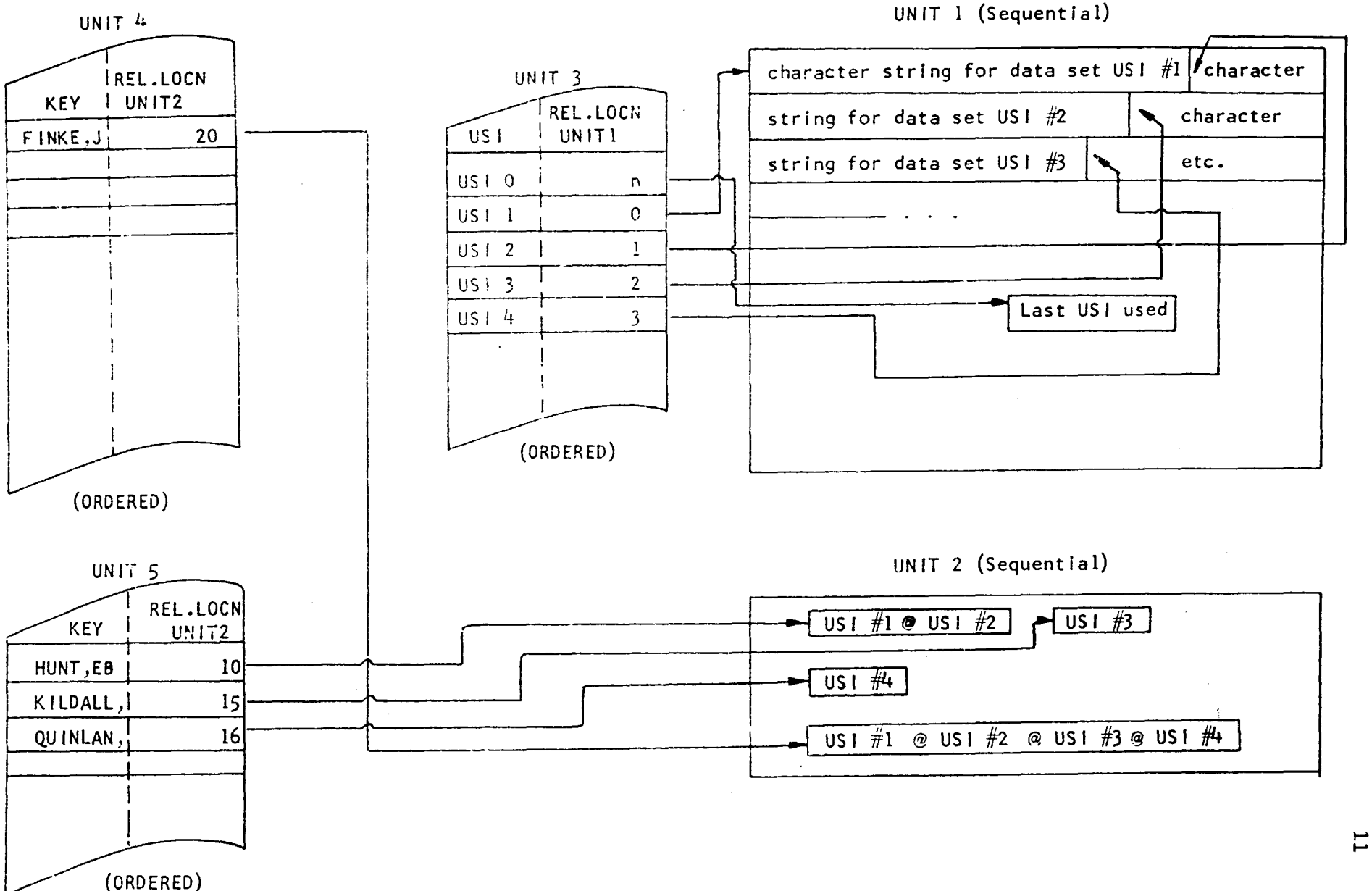
Unit 2 contains character strings of set identifiers associated with unique Reference or Author keys. These character strings are of variable length but the length is a multiple of 8 characters, i.e., each set identifier is 8 characters.

Format

	unique set identifier		unique set identifier	...	unique set indentifier
character 0	7	8	15		etc.

Character strings are added to or deleted from Unit 2 depending on conditions described in sections 2.2.3 and 2.2.4.

The current limit of the number of set identifiers per character string is 500.



2.2.7 Diagram of Interaction Between Sequential and Ordered Storage

3.0 The RETRIEVE Program

RETRIEVE accepts input data on-line directly from a remote terminal or in card image from queued as part of a B5500 packet. This input consists of sets of:

A. file specification

B. query elements (character strings) and their delimiters.

RETRIEVE interprets the bounds of each query set and identifies the query element in the set. RETRIEVE processes each query sequentially. The indexes corresponding to the element delimiter specified are searched for the data element which matches the query element. Set identifiers retrieved for the matched query elements are used to access the actual data sets. The data sets found are then output to the remote terminal or printer.

If no data sets are found for the query element, an appropriate message is output.

3.1 Input to RETRIEVE

There are no column restrictions for input to the RETRIEVE program.

3.1.1 File Specification

The following delimiter identifies the name of a disk file from which data sets will be retrieved:

ex: /FILE filemane

filename is a series of alphanumeric characters whose length is less than or equal to 7; filename would correspond to same file previously created by the STORE program.

3.1.2 Query Sets

The following delimiters identify the bounds of a query set and the query elements within the set.

Delimiter	Function
/QSET	identifies beginning of a new query set
$R	identifies the reference element for which data sets are to be retrieved
$A	identifies the author element for which data sets are to be retrieved
/END	identifies the end of the last query element input

3.1.3 Input Limitations

A. The file specification must precede any query set input

B. Each delimiter must be followed by at least one space, except /END. This is necessary to distinguish a delimiter from the actual character string.

C. Each query set must contain one and only one query element and its delimiter

D. Each line of input must be terminated by a group mark, i.e., "←", when input is entered via the remote terminal. The group mark should only be typed after at least one character "grouping" of the element is typed, e.g

/QSET ←	is incorrect
/QSET $R ←	is incorrect
/QSET $R JONES ←	is acceptable
or $R JONES ←	is acceptable
or $R JONES, A ←	is acceptable
or $R JONES, A B ←	is acceptable

*3.2 The Retrieval Process

When /QSET delimiters are encountered in the input stream, i itialization for a query takes place. When the query element is encountered, a query element key is constructed identical to the method described for data element keys in sections 2.2.4 and 2.2.5. This key is then binary searched through the ordered storage Unit 4 (Reference Indexes) or through the ordered storage Unit 5 (Author Indexes) depending upon the delimiter of the query element, i.e., $R or $A.

If a match does not occur, the message "NO DOCUMENTS FOR THIS REQUEST" and the query element are output.

If no indexes are found in the ordered storage, the message "NO ELEMENTS OF THIS TYPE IN STORAGE" and the query element are output.

If a match does occur, the set identifier character string, whose relative location in Unit 2 is specified in the matched Reference or Author Index, is retrieved. Each set identifier in the string is binary searched through the ordered storage of Unit 3. If no match occurs, the message "SYSTEM ERROR ELEMENT REFERENCE TO DOCUMENT INCOMPLETE" and the query element are output. If a match does occur, the data set character string, whose relative location in Unit 1 is specified in the matched Set Identifier Index, is retrieved.

At this time a full character compare of the query element and the data element is made.

If a complete match does not occur, the next set identifier in the string is selected and the process begins anew. If a complete match occurs, the unique set identifier and the elements of the data set are output. Each element begins printing on a new line. Sets retrieved are separated by a single blank line.

This process is repeated until all the data sets are processed for set identifiers in the character string. If no hits were detected for any of these data sets after the full character compare was performed, the message "NO DOCUMENTS FOR THIS REQUEST" and the query element are output. The processing of the next query is then begun.

When the /END delimiter is encountered RETRIEVE completes any final "housekeeping" and terminates processing.

4.0 Operating Instructions

There are currently two versions of the STORE program described separately in sections 4.1 and 4.2. Also there are currently two versions of the RETRIEVE program described separately in sections 4.3 and 4.4.

It is assumed that users of these programs be familiar with the B5500 Teletype Users' Manual and the General Information Manual of the University of Washington Computer Center.

4.1 Packet Version of the STORE Program

The control cards and input organization required for execution of the STORE program queued from a packet are as follows:

```
?EXLIBE    <user job number>/STPACKX FROM 1110020
?PROCESS = <time estimate>
?DATA      REMFLIN
 /FILE     <user job number>
 /DSET       etc.
data sets
 /DSET       etc.
  ...
 /END
 ?END
```

Note: Data in card image form is assumed to be contained in cols. 1-72.

4.2 On-line Version of the STORE Program

The control cards and input organization required for execution of the STORE program in an on-line interactive mode are:

```
??EXLIBE   <user job number>/STOREMX FROM 1110020; PROCESS = <time
           estimate> (the beginning of job message will be followed
           by a message to the user; NOW INPUT SETS)
 /FILE     <user job number>
 /DSET     etc.
data sets
 /DSET     etc.
 /END
```

Note: the system messages are halted until the /END statement is typed to terminate processing. The program message RUN COMPLETED SUCCESSFULLY will precede subsequent system messages.

4.3 Packet Version of the RETRIEVE Program

Described below are the control cards and input organization required for execution of the RETRIEVE program queued from a packet.

```
?EXLIBE   <user job number>/RTPACKX FROM 1110020
?PROCESS = <time estimate>
?LINES = <line estimate>
?DATA     REMFLIN
/FILE     <user job number>
/QSET       etc.
query sets
/QSET       etc.
 ...
/END
?END
```

Note: Data in card image form is assumed to be contained in cols. 1-72.

4.4 On-line Version of the RETRIEVE Program

Described below are the control cards and input organization required for execution of the RETRIEVE program in an on-line interactive mode.

```
??EXLIBE    <user job number>/RETRMX FROM 1110020; PROCESS = <time
            estimate> (the beginning of job message will be followed
            by a message to the user:  NOW INPUT SETS)

/FILE    <user job number>
/QSET        etc.
query sets
/QSET        etc.
 ...
/END
```

Note: the system messages are halted by the program. To restart system messages type ?SM

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)

University of Washington
Department of Psychology
Seattle, Washington

2a. REPORT SECURITY CLASSIFICATION

UNCLASSIFIED

2b. GROUP

3. REPORT TITLE

AN INFORMATION STORAGE AND RETRIEVAL CAPABILITY

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Scientific Interim

5. AUTHOR(S) (First name, middle initial, last name)

Jeanne M. Finke
Earl B. Hunt

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
9 April 1969	18	

8a. CONTRACT OR GRANT NO.	AF-AFOSR-1311-67	9a. ORIGINATOR'S REPORT NUMBER(S)	Technical Report #69-04-06
b. PROJECT NO.	9778-01		
c.	6144501F	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	AFOSR 69-1792TR
d.	681313		

10. DISTRIBUTION STATEMENT

1. This document has been approved for public release and sale; its distribution is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
TECH, OTHER	Air Force Office of Scientific Research (SRLB) 1400 Wilson Boulevard Arlington, Virginia 22209

13. ABSTRACT

The programs described here are preliminary versions of an information management system being developed as part of a study of computer aids to human problem solving. A more general sysgem will be announced and documented subsequently.

DD FORM 1 NOV 68 1473

UNCLASSIFIED

14 KEY WORDS	LINK A		LINK B			
	ROLE	WT	ROLE	WT		
COMPUTER PROGRAMS						
EXTENDED ALGOL						
STORAGE						
RETRIEVAL						