LEXICO Guide No. 4

PREPARING AND ENTERING TEXTS

Richard L. Venezky
Nathan Relles
Lynne A. Price

Computer Sciences Technical Report #285
December, 1976

LEXICO Guide No. 4

PREPARING AND ENTERING TEXTS

Richard L. Venezky
Nathan Relles
Lynne A. Price

Computer Sciences Department
University of Wisconsin

December, 1976

TABLE OF CONTENTS

TABLE OF CONTENTS cont.

1. Introduction

This guide describes the preparation of texts for processing by the LEXICO system and the commands used for entering texts into a collection. Text input may be produced in machine-readable form using a keypunch, optical scanner, or remote terminal. However, before a text can be processed by LEXICO, it must be transferred to a disk file. Section 2 below defines the units into which a text is divided for processing by LEXICO. Section 3 describes the ADD block used to schedule the addition of one or more texts into a collection and the ADDCONCORD block used to concord texts as they are added. Section 4 covers the commands used to specify text input parameters. Each command is presented with situations where it might be used. Values for these parameters may be specified for an entire collection or an individual text. A complete discussion of default levels is given in Guide 3, Section 2.

The actions LEXICO takes when processing text input are described in Section 5. The following section mentions encoding text data into machine readable form. Suggestions for preparing texts for processing are made in Section 7. The concluding sections list synonyms and abbreviations that may be used when entering text input specifications, the commands used to enter these specifications, and the words that are reserved in ADD and ADDCONCORD blocks.

2. The Parts of a Text

2.1 General Description

This section discusses the units into which a text must be divided for processing by LEXICO. In general, the user must mark these subdivisions with special characters, or two-character sequences, called delimiters. Different types of delimiters are used to border different units. LEXICO provides system default values for all delimiters, but the user may replace these with characters of his choice.

2.2 Citations

Every text in a collection must be divided into user-defined syntactic/ semantic units called citations. These may be phrases, physical lines, sentences or sentence sequences. In a concordance, every keyword in the text is listed with each of the citations in which it occurs. A citation may be up to 1,530 characters long (including any notes, see Section 2.3). When a text is entered, the user must ensure that the boundary of each citation is unambiguously marked by a terminating character called a citation delimiter. Various characters may be used to terminate different citations in the same text. Citation delimiters may be conventional punctuation (.?!) or any user-defined characters.

2.3 Notes

Material which is not part of the text itself may be associated with a citation. Such a note might be, for example, a translation, source description, or variant reading.

One or more notes may appear before, after, or within any citation. Notes must be enclosed by note delimiters; if a note occurs after the citation proper, the citation delimiter must be placed after the terminating note delimiter. Citation delimiters and IDs will not be recognized within notes.

Words within a note are not entered into a concordance. However, notes are printed with the associated citation in a concordance, text listing, and on slips.

2.4 Identifiers

Finally, associated with every citation must be an identifier or ID. Typical identifiers may be page and line numbers or psalm and verse numbers. Identifiers may have up to three levels and each level may be alphabetic or numeric. The number of levels and the type (alphabetic or numeric) of each level must be consistent for all identifiers in a text, although different identifier schemes may be used for different texts in a collection. It is not necessary for each citation to have a unique ID. IDs, and individual ID levels, may be omitted in text input; they will be constructed by LEXICO as discussed in Sections 4.11 and 5.2.

Alphabetic ID levels may contain letters and numbers but must begin with a letter; an alphabetic level may not contain embedded blanks nor may it contain only blanks. Numeric ID levels may be positive numbers or zero but at least one level must be non-zero. A one-level alphabetic ID may be up to six characters long; a one-level numeric ID may be as large as 68, 719, 476, 735 (numbers must be entered without commas). An alphabetic level of a two-level

ID may be as long as three characters; a numeric level may be as high as 262,143. Finally, an alphabetic level of a three-level ID may be one or two characters, and a numeric level of a three-level ID may be no larger than 4,095.

Some examples of IDs are:

256	(1 level numeric)
1.6.28	(3 levels numeric)
a-106	(2 levels mixed)
ONE	(1 level alphabetic)
R3-4095-A	(3 levels mixed)

2.5 Terminology

The word *citation* is sometimes used to refer to a segment of a text along with the associated ID and any associated notes. In this case, the term *text portion* is used to indicate only the textual material. It should be clear from the context whether *citation* refers only to the text portion, or to the accompanying ID and notes as well.

2.6 Word Delimiters,Types, and Tokens

The text portion of a citation is composed of a sequence of words or *tokens*. These are separated by *word delimiters* such as the symbols blank, comma, colon, and semicolon. When a text is concorded, a list is made of all the unique words, or types, that appear within it. Thus, five occurrences of the same word in a text represent five tokens but only one type.

2.7 Text Names and Text Delimiters

Every text must be given a *text name,* which is a string of up to twelve non-blank characters. When a text is entered into a collection, the text name, enclosed in single quotes, must precede the citations that form the body of the text. The end of the text must be marked by a special one- or two-character symbol called a text delimiter.

2.8 Example of a Coded Text

Suppose a text called 'Dictionary' is to be composed of entries from *The Doubleday Dictionary* (Garden City, N. Y., 1975). An entry taken from page 273, column 2, is shown below:

fluro-*combining form* 1 *Chem*--Indicating the presence of flourine in a compound: *fluorite*. 2 Fluorescence: *fluoroscope*. Also fluor-

Using parentheses as ID delimiters, square brackets for notes, and a virgule (/) as a citation delimiter, the corresponding citation might be coded as

```
(273.2)FLUOR-[COMBINING FORM] 1 CHEM--INDICATING THE PRESENCE OF
FLUORINE IN A COMPOUND:  FLUORITE.  2  FLUORESCENCE:  FLUOROSCOPE.
[ALSO FLUOR-] /
```

With '^^' as the text delimiter, the entire text would be prepared as 'DICTIONARY'

```
                .
                .
                .
(273.2)FLUOR-[COMBINING FORM] 1 CHEM--INDICATING THE PRESENCE OF
                .
                .
                .
^^
```

where the ellipses represent other citations. A further example of a coded text is given in Guide 1, Section 3.6.

3. The ADD Block

To add texts to a collection, a user enters LEXICO commands at a terminal. However, LEXICO performs the actual addition of texts off-line, at a later time. The user specifies text input parameters, such as the values of various delimiters and the name of the file containing text input, in an ADD block. Of course, before interacting with LEXICO, the user must have the text properly coded on a file (see Section 6). This file may contain one or more texts, each preceded by its text name, enclosed in single quotes, and terminated by a text delimiter.

The texts on the file are added to the current collection with a block of the form

```
ADD text ;
  command 1 ;
  command 2 ;
      .
      .
      .
END ;
```

where text is the name of the first text on the file and the commands are text input specifications which differ from the collection defaults. As mentioned in Section 2.7, the text name may be up to twelve characters in length, but may not contain blanks. An alternate block header is

```
ADD ;
```

When this short form is used, LEXICO prompts TEXT NAME, PLEASE? The name of the first text to be added should be entered in response.

When the block header is enterd, LEXICO responds by assigning a numeric text code to the first text being added. Subsequently, the user may identify the text by either its name or code. If more than one text is added, succeeding texts are assigned text codes as they are added to the collection.

The command

```
IGNORE ;
```

may be entered instead of the END statement to terminate the block, cancelling all declarations made within it, without scheduling an off-line job.

When the END statement is entered, LEXICO checks for possible omissions and inconsistencies in the specifications made within the block. If such an error is made (e.g., if the name of the file containing the text input has not been entered, or the same symbol has been declared both a note delimiter and a citation delimiter, the END statement is rejected and an appropriate error message is displayed. The user may then continue as though the END statement had not been entered.

After the END statement is accepted, LEXICO asks the user if a backup of the collection should be created by the resulting off-line job (see Guide 3, Section 6) before the texts are added. The system then prompts for a run priority (see Guide 2, Section 7). Cancelling the run at this point is equivalent to IGNORing the block.

Texts may be concorded in the same off-line job in which they are added to a collection with the block

```
ADDCONCORD text ;
    command 1 ;
    command 2 ;
        .
        .
        .
END ;
```

Again, <u>text</u> must be the name of the first text on the input file, but here the commands may be either text input or concordance specifications.

The latter are described in Guide 6. This block also has an alternate block header:

ADDCONCORD ;

It may also be IGNOREd.

If a collection already contains a text with the name specified in an ADD or ADDCONCORD block, LEXICO displays a warning message to that effect. If the block is not IGNOREd, the old text will be replaced by the one being ADDed.

4. Text Input Specifications

4.1 Delimiters

All delimiters (and sequencing characters, see Section 4.11) may be one-or two- character symbols. However, all two-character delimiters (of all types) must begin with the same character (e.g., $A, $B, $C). Word, citation,ID, note, and text delimiters may not overlap. All delimiters may be entered as collection defaults in a CREATE or UPDATE block. Text defaults and temporary text values may be entered as discussed below. When defining delimiters, the reserved words COMMA, SEMI, QUOTE, BLANK, MINUS, and COLON may be used to enter the symbols they denote (see Guide 2, Section 3.4). It should be noted that LEXICO does not differentiate among the delimiters of one type. Thus, if parentheses are declared as ID delimiters, it is perfectly acceptable to enter a text using only right parentheses, or with right parentheses to the left of each identifier and left parentheses on the right. The delimiters in effect in any block are included in the display produced by the command

```
SPECS ;
```

Entering delimiters is not cumulative; the symbols specified in one of the commands described in the following sections replace the previous delimiters of the appropriate class.

An error (e.g., an attempt to enter a three-character delimiter) may occur while delimiters are being defined. In this case, no portion of the command is processed, even if it contains some correct delimiters. LEXICO does check for consistency among delimiters (e.g., the same symbol cannot be both a word and a citation delimiter). However, this is done at the end of an ADD, CONCORD, or ADDCONCORD block, not when the delimiters are entered.

4.2 Citation Delimiters

The system default citation delimiters are ., ?, and !. This may be overridden by entering up to six citation delimiters with the command

```
CITATION DELIMITERS d1 d2... ;
```

Citation delimiters may be declared as temporary text values in an ADD, ADDCONCORD, or EDIT block. Users should be aware that single-character citation delimiters are stored with a text; two-character citation delimiters are removed when a text is entered into a collection. A citation delimiter will not be recognized as such within a note.

4.3 Word Delimiters

The system default word delimiters are the blank symbol ,, :, and ;. This may be overridden by entering up to six word delimiters with the command

```
WORD DELIMITERS d1 d2 ... ;
```

Word delimiters may be entered as text defaults in an ADD, CONCORD, ADDCONCORD, or UPDATE TEXT block.

4.4 ID Delimiters

The system default ID delimiters are (and). This may be overridden by entering one or two ID delimiters in the command

```
ID DELIMITERS d1 ... ;
```

ID delimiters may be declared as temporary text values in an ADD or ADDCONCORD block.

4.5 ID Level Delimiter

The system default ID level delimiter is /. It may be overridden by entering one ID level delimiter with the command

```
ID LEVEL DELIMITER d1 ;
```

An ID level delimiter may be entered as a temporary text value in an ADD, ADDCONCORD, or EDIT block.

4.6 Note Delimiters

The system default note delimiters are [and]. This may be overridden by entering one or two note delimiters with the command

```
NOTE DELIMITERS d1 ... ;
```

Note delimiters may be declared as temporary text values in an ADD, CONCORD, or ADDCONCORD block.

4.7 Text Delimiter

The system default text delimiter is ^^ . This may be overridden by entering a text delimiter with the command

```
TEXT DELIMITER d1 ;
```

A text delimiter may be entered as a temporary text value in an ADD or ADDCONCORD block.

4.8 INPUT ON

All text input to LEXICO must be placed on a file (see Section 6). If all texts are not to be added in one run, groups of texts may be placed on different files or on different *elements* of a single file. Elements are portions of a file which may be accessed individually. The command for identifying the file or element containing the text input is

```
INPUT ON CARD FILE file ;
```

where *file* is a file or element name of up to twenty-three characters, including any qualifier. If *file* is a file name, it may be terminated by a period, although this is not necessary.

There is no system default for this value; it must be specified by the user. However, if the above command is entered in a CREATE or UPDATE block the specified file or element becomes a collection default. It may also be entered as a temporary text value in an ADD or ADDCONCORD block. The value in effect in any block is included in the display produced by the command

```
SPECS ;
```

4.9 Physical Length

The maximum length of a physical line in the text input is specified with the command

```
PHYSICAL LENGTH n ;
```

where _n_ is the maximum length. This refers to line length as defined by the input device and not as found in the original text. If any line contains more than _n_ characters, those characters after the _n_th are ignored. The system default for this value is 72. Physical length may be entered as a collection default in a CREATE or UPDATE block, or as a temporary text value in an ADD or ADDCONCORD block. The current value is included in the display produced by the SPECS command.

4.10 Limiting the Number of Citations

The command

```
LIMIT CITATIONS TO n ;
```

may be used to have LEXICO skip the remainder of a text after the _n_th citation is encountered in text input. Thus, if the text has fewer than _n_ citations, the entire text will be added. If the text has more than _n_ citations, only the first _n_ will be added; any further processing will proceed as though these citations formed an entire text. This facility is designed for the user who wishes to experiment with a portion of a large text. The system default limit is 10,000 citations. A citation limit may be entered in a CREATE or UPDATE block as a collection default and as a temporary text value in an ADD or ADDCONCORD block. The citation limit is displayed by the SPECS command.

4.11 Automatic Sequencing of Identifiers

Every citation must have an ID, and all IDs in a text must have the same number of levels. However, all IDs need not be overtly entered (see also Section 5.2). In particular, if the rightmost ID level in a text is numeric, LEXICO will generate omitted IDs from the specified ones, using a sequencing scheme selected by the user. These options may also be used when the rightmost ID level is alphabetic, but the results in such cases cannot be easily described.

The sequencing method may be entered as a collection default in a CREATE or UPDATE block. It may be entered as a temporary text value in an ADD or ADDCONCORD block. The scheme in effect in any block is included in the display produced by the command

```
SPECS ;
```

In all sequencing schemes, an identifier is constructed from the previous identifier, whether the latter was explicitly entered with the text or was constructed by LEXICO. The rightmost level is computed by incrementing the corresponding level of the previous ID; any other levels are unchanged from the previous identifier.

The system default method of sequencing on citation delimiters is useful for texts, such as biblical or legal texts, which are divided into chapters and verses. Here, the rightmost level of a system-generated ID is set to one greater than the value of this level in the ID of the immediately preceding citation. This scheme may be selected with the command

```
SEQUENCE ON CITATION ;
```

Sequencing on a special character was developed for situations when IDs are based on page and line, but where the length of each line of the original text may vary in the text input. This occurs, for example, when optical scanners place an end-of-line mark after the last non-blank character on a line and then eliminate all trailing blanks. The command

```
SEQUENCE ON character ;
```

(where _character_ is a one-or two-character symbol) causes the rightmost level of the previous ID to be incremented each time _character_ is encountered. For example, if _character_ occurs three times in a citation whose ID is A/6, under this scheme, the ID for the following citation, if omitted, would be computed as A/9. All occurrences of _character_ are removed when the text is stored in the collection. Only one sequencing symbol may be specified.

A slight problem occurs when the original page line will not fit on a single optical scanner (or keypunch) line. For these situations, a user-defined override character may be placed at the beginning of the next line to signal that the preceding end-of-line character should be ignored. This method of sequencing is selected with the command

```
SEQUENCE ON character1 EXCEPT BEFORE character2 ;
```

Both _character1_ and _character2_ may be one- or two-character symbols. When this sequencing scheme is selected, the rightmost level of the most recent ID is incremented whenever *character1* is encountered and decremented each time *character2* occurs. Both *character1* and *charcter2* are removed from the text when it is stored in the collection.

If a fixed number of characters is used to enter each citation, sequencing may be based on that number. This is done with the command

```
SEQUENCE ON n CHARACTERS ;
```

where *n* is the number of characters containing a citation.

Finally, if the command

```
DONOT SEQUENCE ;
```

is entered, an omitted ID is set equal to the ID of the preceding citation. If the rightmost ID level is alphabetic, this option might be preferred.

4.12 Text Output

When a text is added to a collection, LEXICO generates a text listing showing each citation with its ID, any associated notes, and a sequence number assigned to it. These sequence numbers begin with 1 and increase by one throughout the entire text. They are used to identify specific citations in an EDIT block.

Normally, this listing is printed. However, it may be suppressed or directed to a file or tape. The disposal of text output may be entered as a collection default in a CREATE or UPDATE block or as a temporary text value in an ADD or ADDCONCORD block. The system default condition is selected with the command

```
TEXT OUTPUT ON PRINTER ;
```

The listing may be suppressed with the command

```
DONOT LIST TEXT ;
```

It is requested with

```
LIST TEXT ;
```

The listing may be directed to a file with the command

```
TEXT OUTPUT ON FILE ;
```

where file is a file name of up to twelve characters, including any qualifier, optionally including a terminal period, and perhaps enclosed in quotes. This specifies that the text listing should be directed to an 1110 symbiont file name file. This file may be printed, transcribed onto microfiche, or written onto magnetic tape. It must be created outside of LEXICO (see Guide 2, Section 8.2).

Finally, the listing may be directed to magnetic tape with the command

```
TEXT OUTPUT ON TAPE ;
```

LEXICO will prompt for the MACC tape number of the tape to be used and the number of files that already exist on that tape (see Guide 2, Section 8.3). The user must be careful to supply this information correctly as the system does not check for possible errors. If the user enters this data incorrectly, a repeated TEXT OUTPUT ON TAPE command will cause LEXICO to prompt for this information again. If an error is noticed after the END statement has been entered, when LEXICO prompts for a run priority (see Guide 2, Section 7), the user may indicate that the run should be cancelled.

4.13 Text Memo

The command

```
MEMO 'note about text' ;
```

may be entered in any text-specific block, where note about text is a string of up to 100 characters. This memo will be displayed whenever the status of the text is displayed (see Guide 3, Section 4.2). One use of this statement is to enter text titles which are longer than the 12 characters allowed in text names.

4.14 Specs

All text input and concordance specifications in effect in any block are displayed by the command

```
SPECS ;
```

5. Processing Text Input

5.1 Introduction

This section describes the actions taken by LEXICO as it adds texts to a collection. It should be read carefully before text input is prepared, as typing or keypunching may be simplified by utilizing some of the conventions discussed here.

5.2 Identifiers

To allow typing or keypunching directly from printed editions where identifiers commonly occur in margins, an ID may occur anywhere between a text name and a text delimiter except in a note. An ID is always associated with the first complete citation following it.

The number of levels and type of each level in all identifiers in a text must correspond to the identifier of the first citation. If this identifier is omitted, LEXICO assumes three-level numeric IDs are to be used, and sets the first identifier to 0/0/1. If an entered ID has fewer levels than the initial identifier, it is assumed that the specified levels should not be changed from the previous ID. For example, if IDs represent chapter and verse numbers, the chapter number need be given only with the first verse of each chapter. If an entire ID is omitted, LEXICO constructs it as discussed in Section 4.11.

Whenever an error occurs in an ID (e.g., a letter in a numeric level), that level is set equal to the previous correct value. If all levels of a numeric ID are 0, LEXICO sets the rightmost level to 1.

5.3 Blanks

LEXICO ignores all blanks within the quotes surrounding a text name in the text input. However, blanks are not ignored in the text name given on-line in an ADD or ADDCONCORD block. Since these two text names must match, blanks should not occur in the text name given interactively.

In the body of the text, a group of blanks is equivalent to a single blank. Thus, the typist or keypuncher may leave several spaces at the end of a card or line, but this will appear as a single blank after the text has been added to a collection.

5.4 Line Boundaries

If _n_ is the physical length (see Section 4.9) of the text input, the character in the _n_th column of one line may be thought of as being immediately followed by the first character of the next line. Therefore, a word may be split across two lines, and it may be necessary to start a line with a blank.

5.5 Omitted Characters

Certain characters that occur in the text input will be removed as the text is added to a collection. These will not appear in the text listing or when citations are listed in a concordance or slips. These characters include two-character citation delimiters (see Section 4.2) and sequencing characters (see Section 4.11). One-character citation delimiters are not removed.

5.6 Errors

Various errors may occur in text input, e.g., identifiers may have the wrong number of levels or levels of the wrong type, or citations may be too long. LEXICO processes the first ten citations of a text regardless of the number of errors detected. It then continues unless four errors are found in any twenty consecutive citations after the first ten. If this occurs, the system skips the remainder of this text, although any additional texts found are added. The portion of a text that has been added in this fashion may be processed as though it were an entire text.

6. Transcribing Texts Into Machine-Readable Form

A text can be transferred to machine-readable form in various ways. Many methods involve the generation of an input tape from cards, optical scanner sheets, or directly from a keyboard. However, before the text can be processed by LEXICO, it must be placed on a file. If magnetic tape is to be used as on intermediate medium, it is recommended that card images be blocked with at least 100 card images per physical block. The transfer of data from media such as tape or cards to files and the editing of files is discussed in the MACC Computing Handbook; MACC consultants can provide advice in particular cases.

Text input to LEXICO may be placed on a file or on an element of a file. The creation and naming of files is discussed in Guide 2, Section 8.2. Files may be divided into units called elements; each element may be processed separately. An element is identified by a file name and an element name, separated by a period. For example, an element called TEXT of a file named TEST*ING is referred to as TEST*ING.TEXT

The following runstream may be used to copy a deck of cards to a file or element

```
@RUN name,proj,user,1M
@PASS password
@DATA,I file.              or      @ELT,I file,element

  deck of cards containing
    LEXICO text input
  (beginning with text name,
  ending with text delimiter)

@END
@FIN
```

Here name is the user's name, proj is his project number, user is his user number, password is his password, file is the file name, including the qualifier if the latter is different from his user number, and element is the element name if an element is being written. The @DATA,I statement is used to write a file; @ELT,I to write an element. The @END statement must not be omitted. The control statements to create and save the file (see Guide 2, Section 8.2) may be inserted after the @PASS statement. Of course, this is only an example of how text input may be transferred to a file. The MACC Computing Handbook provides a more thorough description of alternatives.

7. Preparation Procedures

Users of computing systems such as LEXICO sometimes transfer their data to a machine-readable form without considering the details of the eventual processing to be performed. Although this may be unavoidable, the frequent result is that either the data must be copied, with a slight change of format, from one machine-readable form to another, or that features which, with slightly different preparation, could have proven convenient, become unusable. For this reason, it is recommended that prospective users of LEXICO select the options they wish to use as soon as possible. Although a thorough understanding of any system's operation may be difficult to obtain without personal experience, these decisions should ideally be made before any typing, keypunching, or optical scanning is done. Of course, plans for processing data are likely to be modified as soon as they are put into execution.

The first step in text preparation should be a thorough reading of this document (Guide 4) and of Guide 6. After the user is familiar with such options as notes, automatic sequencing of identifiers, removable characters and stop characters, he should decide which system features he wishes to use. Then he must determine how the text is to be broken into citations. The next step is the selection of an identifier scheme--the number of levels and the type of each level.

After these basic decisions have been made, the user must consider how individual characters are to be used. The available characters are @[]#‸b̸ABCDEFGHIJKLMNOPQRSTUVWXYZ)-+<=>&$*(%:?!,\0123456789';/." where b̸ denotes the blank character.

where ƀ denotes the blank symbol. On some terminals, keypunches, and printers, a few of these characters (e.g.,^,[,]) may vary.

From the above characters, the user must select a set of delimiters. If he wishes to use sequencing characters, removable characters or stop characters, these must also be chosen. Most of these characters (or two-character) symbols cannot be used for more than one purpose. However, the ID level delimiter may also be used as a word delimiter, citation delimiter, or note delimiter. This means, for example, if a period is to be used as a word delimiter, periods cannot also occur in ellipses or abbreviations.

Finally, using the chosen delimiters, the text can be encoded into a machine-readable form as discussed in Sections 2 and 6.

8. Synonyms and Abbreviations

Members of the following groups of expressions may be used interchangeably in the commands discussed in this guide:

BEFORE, BY, EXCEPT, FOR, FROM, INTO, OF, ON, OR, OUT, THRU, TO, WITH;
CARD, CARDS ;
CHARACTER, CHARACTERS ;
CITATION, CITATIONS, CIT ;
DELIMITER, DELIMITERS, DELIM ;
DONOT, NO, NOT ;
TEXT, TEXTS ;
WORD, WORDS ;

9. Summary of Commands

9.1 The ADD Block

The following commands may be entered in an ADD block:

```
SPECS ;

WORD DELIMITERS d1 d2 ... d6 ;
ID DELIMITERS d1 d2 ;
ID LEVEL DELIMITER d1 ;
CITATION DELIMITERS d1 d2 ... d6 ;
NOTE DELIMITERS d1 d2 ;
TEXT DELIMITER d1 ;

SEQUENCE ON CITATION ;
SEQUENCE ON character ;
SEQUENCE ON character1 EXCEPT BEFORE character 2 ;
SEQUENCE ON n CITATIONS ;
DONOT SEQUENCE ;

INPUT ON CARD FILE file ;
PHYSICAL LENGTH n ;
LIMIT CITATIONS TO n ;

SQUEEZE OUT characters ;
FRONTSTRIP characters ;
ENDSTRIP characters ;
DONOT SQUEEZE OUT ;
DONOT FRONTSTRIP ;
DONOT ENDSTRIP ;

LIST TEXT ;
DONOT LIST TEXT ;
TEXT OUTPUT ON PRINTER ;
TEXT OUTPUT ON TAPE ;
TEXT OUTPUT ON file ;

MEMO 'note about the text' ;

END ;
IGNORE ;
```

9.2 The ADDCONCORD Block

The following commands may be entered in an ADDCONCORD block:

```
SPECS ;

WORD DELIMITERS d1 d2 ... d6 ;
ID DELIMITERS d1 d2 ;
ID LEVEL DELIMITER d1 ;
CITATION DELIMITERS d1 d2 ... d6 ;
NOTE DELIMITERS d1 d2 ;
TEXT DELIMITER d1 ;
```

```
SEQUENCE ON CITATION ;
SEQUENCE ON character1 ;
SEQUENCE ON character1 EXCEPT BEFORE character2 ;
SEQUENCE ON n citations ;
DONOT SEQUENCE ;

INPUT ON CARD FILE file ;
PHYSICAL LENGTH n ;
LIMIT CITATIONS TO n ;

LIST TEXT ;
DONOT LIST TEXT ;
TEXT OUTPUT ON PRINTER ;
TEXT OUTPUT ON TAPE ;
TEXT OUTPUT ON file ;

CONCORDANCE OUTPUT ON PRINTER ;
CONCORDANCE OUTPUT ON TAPE ;
CONCORDANCE OUTPUT ON file ;

SQUEEZE OUT characters ;
FRONTSTRIP characters ;
ENDSTRIP characters ;
DONOT SQUEEZE OUT ;
DONOT FRONTSTRIP ;
DONOT ENDSTRIP ;

DELETE ALL STOPWORDS ;
ADD STOPWORDS word1 word2 ... ;
DELETE STOPWORDS word1 word2 ... ;
STANDARD STOPWORDS ;
SHOW STOPWORDS ;
STOP LENGTH n ;
STOP LENGTH n OR LESS ;
AUTOSTOP characters ;
DONOT AUTOSTOP ;
REVERSE ;

COLLATE OE ;
COLLATE STANDARD ;
COLLATE NEW sequence ;

LIST TYPES BY FREQUENCY ;
LIST TYPES REVERSED ;
DONOT LIST TYPES BY FREQUENCY ;
DONOT LIST TYPES REVERSED ;

MEMO 'note about the text' ;

END ;
IGNORE ;
```

10. Reserved Words

10.1 The ADD Block

The following words are reserved in an ADD block:

ADD
ADDCONCORD
ALL
AUTOSTOP
BACKUP
BASETYPE
BEFORE
BLANK
BY
CARD
CARDS
CHARACTER
CHARACTERS
CIT
CITATION
CITATIONS
CLEAR
CLEANUP
COLLATE
COLLECTION
COLON
COMMA
CONCORD
CONCORDANCE
CONCORDANCES
COPY
CREATE
D
DELETE
DELIM
DELIMITER
DELIMITERS
DIR
DIRECTORY
DISPLAY
DO
DONOT
EDIT
END
ENDSTRIP
EXCEPT
FILE
FOR
FREQUENCY
FROM
FRONTSTRIP
ID
IGNORE
INPUT
INTO
LENGTH
LESS
LEVEL
LIMIT
LIST
LISTS
LOOKUP
MEMO
MINUS
NEW
NO
NOT
NOTE
NOTES
OE
OF
ON
OPTION
OR
OUT
OUTPUT
PACK
PHYSICAL
RENAME
RESPELL
RESPELLED
RESTORE
REVERSE
REVERSED
ROUTE
RULE
RULES
S
SEMI
SEQUENCE
SH
SHOW
SLIPS
SPECS
SPELLING
SQUEEZE
STATUS
STANDARD
STOP
STOPWORD
STOPWORDS
TEXT
TEXTS
THRU
TO
TYPE
TYPES
UPDATE
WITH
WORD
WORDS

10.2 The ADDCONCORD Block

The following words are reserved in an ADDCONCORD Block:

ADD
ADDCONCORD
ALL
AUTOSTOP
BACKUP
BASETYPE
BEFORE
BLANK
BY
CARD
CARDS
CHARACTER
CHARACTERS
CIT
CITATION
CITATIONS
CLEAR
CLEANUP
COLLATE
COLLECTION
COLON
COMMA
CONCORD
CONCORDANCE
CONCORDANCES
CREATE
D
DELETE
DELIM
DELIMITER
DELIMITERS
DIR
DIRECTORY
DISPLAY
DO
DONOT
EDIT
END
ENDSTRIP
EXCEPT
FILE
FOR
FREQUENCY
FROM
FRONTSTRIP
ID
IGNORE
INPUT
INTO
LENGTH
LESS
LEVEL
LIMIT
LIST
LISTS
LOOKUP
MEMO
MINUS
NEW
NO
NOT
NOTE
NOTES
OE
OF
ON
OPTION
OR
OUT
OUTPUT
PACK
PHYSICAL
RENAME
RESPELL
RESPELLED
RESTORE
REVERSE
REVERSED
ROUTE
RULE
RULES
S
SEMI
SEQUENCE
SH
SHOW
SLIPS
SPECS
SPELLING
SQUEEZE
STATUS
STANDARD
STOP
STOPWORD
STOPWORDS
TEXT
TEXTS
THRU
TO
TYPE
TYPES
UPDATE
WITH
WORD
WORDS