

Lawrence Berkeley National Laboratory

Recent Work

Title

USER'S MANUAL FOR THE DOE TECHNICAL INFORMATION CENTER INTEGRATED RECORD INPUT SYSTEM (IRIS).

Permalink

<https://escholarship.org/uc/item/74f17835>

Authors

Lawrence, J.D.
Haimand, T.

Publication Date

1983-03-01



Lawrence Berkeley Laboratory

UNIVERSITY OF CALIFORNIA

Engineering & Technical Services Division

RECEIVED
LAWRENCE
BERKELEY LABORATORY
JUN 8 1983
LIBRARY AND
DOCUMENTS SECTION

USER'S MANUAL FOR THE DOE TECHNICAL INFORMATION
CENTER INTEGRATED RECORD INPUT SYSTEM (IRIS)

J. Dennis Lawrence and Todd Hammond

March 1983

TWO-WEEK LOAN COPY

*This is a Library Circulating Copy
which may be borrowed for two weeks.
For a personal retention copy, call
Tech. Info. Division, Ext. 6782.*



LBL-15731
c.2

DISCLAIMER

This document was prepared as an account of work sponsored by the United States Government. While this document is believed to contain correct information, neither the United States Government nor any agency thereof, nor the Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or the Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof or the Regents of the University of California.

DOE/TIC-11616
LBL-15731

User's Manual for the
DOE Technical Information Center
Integrated Record Input System
(IRIS)

J. Dennis Lawrence and Todd Hammond

Lawrence Berkeley Laboratory
University of California
Berkeley, California 94720

Published March 1983

This work was supported by the Director, Office of Energy Research,
Office of Basic Energy Sciences, Materials Sciences Division
of the U. S. Department of Energy under Contract No. DE-AC03-76SF00098.

ABSTRACT

This report explains how to use the Integrated Record Input System (IRIS) being developed for the DOE Technical Information Center. The system is more fully described in a companion report: *Development of an Integrated Record Input System for DOE's Technical Information Center*, DOE/TIC-11615 (LBL-15723), November 1982.

CONTENTS

1. Introduction	1
2. APPEND — Retrieving Completed Citations	3
3. BLDINV — Build Thesaurus Inverted Index	8
4. BLDLOC — Build an Empty Location File	12
5. CDUMP — Dump ASCII File on Terminal	14
6. CLNUP — Cleanup a Completed Citation File	16
7. COMPCT — Compact Scattered Files	19
8. CPYCIT — Copy Citation Records	23
9. DECODE — Dump Non-ASCII File on Terminal	27
10. LISTMT — List Thesaurus Main Terms	29
11. LISTPI — List Inverted Index Words	31
12. LOCATE — Find Citation Locations	33
13. PFILE — Print Files on Omni 820	36
14. PRTCDR — List Thesaurus Coarse Directory	38
15. ROUTE — Manual Routing of Citation Records	40
16. SEQCIT — Sequence Citation Records	44
17. SORT — Route Citations to Category Files	47
18. THSGRF — Thesaurus Display	51
19. TICEDT — Edit Citation Records	60
20. TYPCIT — Print Citations on a Terminal	76

1. Introduction

This document tells how to interact with the various programs in the DOE/TIC Integrated Record Input System (IRIS). Concentration is on specifics of using each program; more general issues are discussed in Report DOE/TIC--11615. Please read this report before trying to use the programs described here.

Documentation on each program generally includes the following topics:

Purpose.	What task is the program intended to accomplish?
Audience.	Who are the primary users of the program?
Areas.	In what DEC 10 areas (master, indexer, supervisor, developer, system) will the program operate correctly?
Terminals.	What terminal types may be used?
Files.	What input, output, recovery, and authority files are used by the program, and in what areas must they reside?
Dialog.	How is the program invoked and controlled by the user, and how does it respond to user commands?
Errors.	What error conditions can arise, what messages are displayed for them, and how does the program react?
Execution.	How does the program carry out its task? (This is included only if it appears helpful to actually using the program.)

The following terminology is used:

@	RETURN key of the terminal
Current Area	The DEC 10 area currently being used
Developer	Refers to the LBL staff who are developing the computer system
Developer Areas	The DEC 10 areas being used by the developers
Indexer	Refers to the TIC staff who index document citations
Indexer Areas	The DEC 10 areas being used by the indexers
Master Area	The DEC 10 area being used to hold files used by all indexers and supervisors
Supervisor	Refers to the TIC staff who supervise the indexers
Supervisor Areas	The DEC 10 areas being used by the supervisors
System Area	The DEC 10 area used to contain all production versions of the software
System Manager	The person who manages the files in the master area, and generally assists the user's of the system
System Programmer	Refers to the TIC programmers who provide assistance to the indexers and supervisors

These program instructions are meant to be used independently of one another. Consequently, there is a fair amount of repetition where a comment applies to several programs.

The terminal screen has 80 columns — somewhat more than is available in this document. Consequently, some displays are compressed here, usually by using two lines to show a program message that requires just one line on the terminal.

Environmental Control

When you log onto an indexer or supervisor area, several measures are taken to make sure the system knows what kind of terminal you are using. After you have entered your password, and system sign-on messages have been displayed, you will see one of the following:

.tset

*?'H1552'

Terminal type: (h1552)

.tset

*?'vt100'

Terminal type: (vt100)

If the terminal you are using is the same as is indicated in these messages, just press the *RETURN* key. Otherwise, enter a code for the terminal you are using from the following list. For example, if you are using a DEC VT 100 terminal, respond with "VT100", or "100". You may use upper or lower case, as you wish.

Terminal	Legal Codes
AJ 830/833	AJ, 830, 833
Ann Arbor	ANN, Ann Arbor
DEC VT 100	VT100, 100
DEC VT 52	VT52, 52
DEC LA 120	LA120, LA, 120
Hazeltine 1552	H1552, H52
Hazeltine 1500	H1500
Heathkit H19	H19, 19
LSI ADM 3a	ADM3A, ADM, 3A
LSI ADM 31	ADM31, 31
Tektronix 4006	TEK, 4006
TI Omni 800	OMNI 800, OMNI
TI Silent 700	S700, Silent 700

This information is saved, and is then used by most of the programs to tailor terminal displays for your specific device. Consequently, you will not be asked repeatedly for your terminal type.

In the program descriptions given below, the phrase "operates correctly on any terminal type" is used repeatedly. This means that you may use any terminal on the list just above. A few programs will run only on specific terminals; these are indicated.

Error Messages

The documentation provided here gives only those error messages that are peculiar to the program under discussion. Many other error messages can arise from other parts of the computer system: the operating system, FORTRAN library subroutines, RATFOR library subroutines, and indexing system library routines. Messages that may arise in the latter two portions are not described in this document. Thus, if you see an error message that is not described in this document, it probably comes from one of the first three sources, and you should consult their documentation. Hopefully, this will be a very rare occurrence.

2. APPEND - Retrieving Completed Citation Files

2.1. Purpose

This program will examine all indexer and supervisor areas for citation files containing completed, rejected, or rerouted records, and records with new terms. Such files are then transferred to the current (generally, the master) area. The input files are the result of running the **TICEDT** program. Records are appended to files in the current area, and the originating files are deleted.

When all records have been transferred from all indexer and supervisor areas, a report is displayed on the terminal screen, and written to a file, giving statistics on the record movement.

2.2. Audience

APPEND will normally be run by the system manager.

2.3. Areas

APPEND will operate correctly in any area, but will normally be run in the master area.

2.4. Terminals

APPEND will operate correctly on any terminal type, including both display screen terminals and hard copy (line-by-line) terminals.

2.5. Files

2.5.1. Input Files

For each indexer or supervisor area examined, the following files will be used as input if they exist. Each file is deleted after it is copied to the current area.

SAVE.DAT	contains citation records completed by the indexer or supervisor.
CATyyB.DAT	contains citations rerouted to specific categories by the indexer ('yy' is the new category number).
CATUNK.DAT	contains citations rerouted to the "unknown category" file by the indexer.
CATREJ.DAT	contains citations that have been rejected, as unwanted in RECON.
NEWTRM.DAT	contains citations with suggested new thesaurus terms.

2.5.2. Output Files

All output files are in the current area (normally, the master area):

SAVE.nnn	contains all records from file SAVE.DAT from the indexer or supervisor area identified by the initials 'nnn'.
CATxxA.DAT	contains citations copied from the CATxxB.DAT files.
CATUNK.DAT	contains citations copied from the CATUNK.DAT files in the input areas.
CATREJ.DAT	contains citations copied from the CATREJ.DAT files in the input areas.
NEWTRM.DAT	contains citations copied from the NEWTRM.DAT files in the input areas.
REPORT.APP	contains a copy of the report displayed on the terminal screen at the end of the job. Any previous contents of this file are overwritten.

2.5.3. Recovery Files

A state file, **STATE.APP**, is created by **APPEND** to keep track of its progress. If **APPEND** does not complete (due to system crash, use of **CTRL-C**, or whatever), this file is used to recover and continue processing. This file is deleted when **APPEND** terminates normally.

Note that the recovery file is stored in the current area, not the master area. Unless the current area is the master area, this state file will not be detected by other programs that look for it.

2.5.4. Authority Files

Two files are used to provide data to **APPEND**, and both must be present in the master area:

CODES.LIS gives the legitimate first level category numbers.
INDEX.LIS gives area numbers, initials, and user type (indexer or supervisor) for all areas to be searched by **APPEND**.

2.6. Dialog

1. **APPEND** may be run in any area, but will normally be run in the master area. It is normally run using a mic file. The mic file establishes the proper environment and invokes the **APPEND** program. When ready to proceed, enter *RETURN*.

User:	.do append
APPEND:	.silence
	Type RETURN to begin
	*
User:	@

2. There are several possible responses at this point, depending on whether the program is in recovery mode or not:
 - a. If **APPEND** is not in recovery mode, it proceeds to process the various areas and files, as shown in step 3.
 - b. If **APPEND** has discovered that it did not complete the previous run, and goes into recovery mode, the following is displayed. The program cannot be restarted until the previous run has been completed successfully. After this message, the program proceeds as in step 3.

APPEND:	= R E C O V E R Y =
---------	---------------------

- c. If a state file for program **SORT** exists, **APPEND** cannot be run. You must run **SORT** to completion before restarting **APPEND**. The following messages are displayed, and the program terminates.

APPEND:	A SORT state file exists, so APPEND cannot be run.
	You must run SORT to completion before restarting APPEND.

- d. Likewise, if a state file for **ROUTE** exists, **APPEND** cannot be run.

APPEND:	A ROUTE state file exists, so APPEND cannot be run.
	You must run ROUTE to completion before restarting APPEND.

3. **APPEND** now begins (or continues) to process areas and files. As each area is started, the area number is displayed. As each file is copied, the number of records copied is displayed. These displays are primarily to reassure you that the program is operating correctly.

APPEND:	*** Scan of area 301 started ***
	3 record(s) copied from save.dat
	4 record(s) copied from catunk.dat
	6 record(s) copied from cat46b.dat

4. When all records from all files in all areas have been copied, **APPEND** pauses to allow you to take any special action you wish before the summary reports are displayed. For example, if you are using a hard copy terminal, you may wish to manually adjust the paper.

APPEND:	Please press RETURN to see summary tables
User:	@

5. There are three summary reports. The first shows the total number of records copied from each area:

APPEND:	Number records copied, by original area			
	Area	# records	Area	# records
	301	13	304	0
	302	15	305	7
	303	17		

6. The second summary report shows the number of citation records copied to each category file, followed by the the total number of rejected records copied, and the total number of records copied that have suggested new thesaurus terms.

APPEND:	Number records copied, by category			
	Category	# records	Category	# records
	01	27	45	17
	44	3	unk	4
	46 rejected records were copied			
	16 records with new terms were copied			

7. The third summary report shows the number of records copied to each save file, by indexer's/supervisor's initials:

APPEND:	Number records copied to each save file			
	Indexer	# records	Indexer	# records
	ABC	3	GHI	0
	DEF	15		
	End of job			

2.7. Error Messages

File xxxxxx.xxx cannot be opened.

APPEND cannot continue.

Please ask the system manager to correct the error, and rerun APPEND.

The indicated file could not be opened. **APPEND** must have access to this file to continue, so it terminates. If you cannot provide the file, please ask the system manager for help. He will need to know the file name.

Record yy of file xxxxxx.xxx cannot be read.

APPEND cannot continue.

Please ask the system manager to correct the error, and rerun APPEND.

The indicated record could not be read; please correct the error, if you can, or ask the system manager for help. He will need to know the file name and record number.

Record yy of file xxxxxx.xxx cannot be written.

APPEND cannot continue.

Please ask the system manager to correct the error, and rerun APPEND.

The indicated record could not be written; please correct the error, if you can, or ask the system manager for help. He will need to know the file name and record number.

Record yy of file xxxxxx.xxx does not match record zz of file wwwwww.www.

Preceding records match.

Please inform the system manager that these files are damaged.

APPEND is in recovery mode, and attempting to realign the indicated files so processing can continue. The files do not match. This is a serious error, and indicates that one or both files have been damaged in some fashion. There is probably little that you can do about this without the assistance of the system manager. He will need to know the file names and record numbers.

Record yy of file xxxxxx.xxx lacks a serial number.

Please inform the system manager.

APPEND continues.

APPEND is in recovery mode, and attempting to read the indicated file for serial numbers. The specified record lacks a serial number. This is an informational message; **APPEND** continues the recovery process. However, you should inform the system manager of the difficulty; he will need to know the file name and record number.

2.8. Summary of Execution

APPEND first looks to see if its state file, STATE.APP, exists. If so, recovery is assumed to be necessary, so all files are positioned to correspond to the situation just before the previous

abnormal termination.

If no state file exists, **APPEND** looks to see if the **SORT** or **ROUTE** state files exist. If either of these files exists, then **APPEND** cannot be run, so it terminates. All three of these programs write to the same category files, so if any of them has crashed, neither of the other two can be run until any file damage caused by the crash has been repaired. Furthermore, the three programs cannot be run simultaneously, or the category files may become hopelessly confused; this check will help prevent this conflict as well.

If all of these tests are passed, the two authority files are copied to memory. If either is absent from the master area, the program halts. This is not considered a crash (no state file has been created), since no category files have been altered as yet.

File **INDEX.LIS** contains a list of all areas to be processed. For each area listed in this file, the following is carried out:

A message is displayed announcing that the area has been started. If the area contains a save file, **SAVE.DAT**, it is copied to the master area file **SAVE.nnn**, where 'nnn' is the indexer's or supervisor's initials obtained from file **CODES.LIS**. If the area contains a file of rejected records, **CATREJ.DAT**, it is copied to the master area file **CATREJ.DAT**. If the area contains any category files **CATyyB.DAT**, where 'yy' is a category number, they are copied to the corresponding category files in the master area, **CATyyA.DAT**. Files containing unknown category records, and files containing potentially new thesaurus terms, are copied similarly.

After each file is copied, a message is displayed containing the record count and file name. Then, the file is deleted from the indexer or supervisor area.

After all files are copied, the summary report is displayed and saved in file **REPORT.APP**.

3. BLDINV - Build Inverted Index to the EDB Thesaurus

3.1. Purpose

This sequence of programs will build an inverted index of significant words that occur in the EDB thesaurus. The procedure involves four programs:

CPIWCP	will read the fine directory to the thesaurus, extract words from main terms, delete any insignificant words, and store the remaining words (with an indication of the main terms they were derived from) in temporary file CPIWCP.TST.
SORT	will sort the temporary file created by CPIWCP lexicographically, and store in file CPIWCP.SRT.
CPIBLD	will read the sorted file and build the inverted index.
DELETE	will delete the two temporary files.

A word is considered insignificant if (1) it has fewer than three characters, (2) it contains no letters, or (3) it occurs on a stop list. Words are defined in two different ways, and words found by either definition are included in the inverted index. First, any string of letters, numbers, commas, and dashes that is preceded and followed by some other symbol is a word. Second, if the word contains either commas or dashes, all portions that include only letters and numbers are also considered to be words. For example, consider the main term 'Di-2-Ethylhexylphosphoric Acid'. By the first definition, two words are found: 'Di-2-Ethylhexylphosphoric' and 'Acid'. By the second definition, three additional words are found: 'Di', '2', and 'Ethylhexylphosphoric'. 'Di' is rejected as too short, and '2' is rejected because it contains no letters, leaving three words to go into the inverted index.

3.2. Audience

BLDINV will normally be run by the system manager. It should be run every time the thesaurus is updated; otherwise, the inverted index will not match the thesaurus files, and programs that use the inverted index (such as THSGRF) will not operate correctly.

3.3. Areas

BLDINV will operate correctly in any area, but should normally be run in the master area.

3.4. Terminals

BLDINV will operate correctly on any terminal type, including both display screen terminals and hard copy (line-by-line) terminals.

3.5. Files

3.5.1. Input Files

The only input file is the thesaurus fine directory, THESAU.FDR.

3.5.2. Output Files

The inverted index is placed in three files, all in the master area. The files are: THESAU.PMX, THESAU.PDR, and THESAU.PBA.

3.5.3. Working Files

There are two working files, CPIWCP.TST and CPIWCP.SRT. Both files are deleted at the end of the procedure.

3.5.4. Recovery Files

There are no recovery files, since there is no recovery procedure. If the process does not run to completion, it must be restarted.

3.5.5. Authority Files

There is one authority file, STOP.LIS, containing the list of stop words. The file contains one stop word per record, and must be in alphabetic order. It must be in the current area.

3.6. Dialog

1. **BLDINV** is a mic file. It may be run in any area, but will usually be run in the master area. The mic file establishes the proper environment, and invokes program **CPIWCP**. When ready to proceed, enter *RETURN*.

User:	.do bldinv
BLDINV:	.run cpiwcp
CPIWCP:	*
User:	@

2. Program **CPIWCP** now reads through the fine directory to the thesaurus, extracting significant words from the main terms listed in that directory, and storing them in working file **CPIWCP.TST**. As each of the more than 1100 segments of the fine directory are started, a progress report is displayed, to reassure you that the program is operating correctly. When the program is finished, a summary report is displayed, giving the total number of main terms read and the total number of distinct word occurrences written in **CPIWCP.TST**.

CPIWCP:	Beginning segment	10
	Beginning segment	20
	Beginning segment	30
		
	Beginning segment	1110
	Main terms read:	25154
	Total words written:	49928

3. The system sort program is now run, to sort the working file into lexicographic order.

BLDINV:	.r sort *cpiwcp.srt=cpiwcp.tst/fixed/record:70/key:1:70
SORT:	[SRTXPN Expanding to 161P] Sorted 49928 records 733466 KEY comparisons, 14.69 per record 4071 record leaves in memory 5 runs, bias 2.45 0:01:04 CPU time, 1.30 MS per record 0:07:17 elapsed time
BLDINV:	*/exit

4. Program **CPIBLD** is now run. It will read the sorted work file, deleting duplicate words, and build the inverted index. As each 10 blocks of the index are written, a progress report is issued to reassure you that the program is operating correctly. When the program has finished, this fact is noted on the terminal, and a summary report is displayed, giving the number of word occurrences that were read from the input working file, the number of words written to the inverted index file (ie, the number of distinct words found in the input file), and the number of blocks in the inverted index directory.

BLDINV:	.run cpibld
CPIBLD:	*
	Block 10 written
	Block 20 written
	Block 30 written
	...
	Block 530 written
	Finished
	Words read: 49928
	Words written: 15123
	Blocks written: 538

5. A final table is printed by **CPIBLD** giving an indication of frequencies of words in thesaurus terms. In the example shown here, 10,799 of the 15,123 words (ie, 71%) occurred in only one main term. 1730 words occurred in two main terms, for a cumulative 82%. Finally, 18 words occurred in twenty main terms, for a cumulative 97%. From here on, individual occurrences are listed: 11 words occurred in 21 main terms, etc. At the end, we find one word that occurred in 1866 main terms.

Table of number of main terms per word

1	10799	71%	6	183	93%	11	38	96%	16	24	97%
2	1730	82%	7	98	94%	12	36	96%	17	26	97%
3	692	87%	8	109	94%	13	34	96%	18	13	97%
4	452	90%	9	83	95%	14	25	96%	19	17	97%
5	281	92%	10	73	95%	15	23	96%	20	18	97%
21	21	21	21	21	21	21	21	21	21	21	22
22	22	22	22	22	22	22	22	22	22	22	22
...	...	...	...	...	...	...	...	...	...	...	...
176	193	194	196	209	293	360	482	1866	...	...	...

6. The system delete program is run to eliminate the two working files, and the mic procedure is finished.

BLDINV:	.delete cpiwcp.tst,cpiwcp.srt
DELETE:	Files deleted:
	DOE:CPIWCP.TST
	DOE:CPIWCP.SRT
	11234 Blocks freed

3.7. Error Messages

3.7.1. CPIWCP Error Messages

Stop list too long — increase NBR_ENTS

The file of stop words contains more than 300 entries. The program terminates.

Stop word too large — change NBR_LETS to xxx

A stop word contains more than 15 characters. The program terminates.

Word too long. nnn characters

<<the actual word is printed here>>

A word in a main term has been found with more than 60 characters. All characters following the 60-th are ignored, and the program continues. The word is printed in the message.

3.7.2. CPIBLD Error Messages

The following word was used in nnn main terms — only the first 2000 were stored

<<the actual word is printed here>>

A word has been found that occurs in more than 2000 main terms. Only the first 2000 pointers to main terms are stored in the inverted index.

4. BLDLOC - Build an Empty Location File

4.1. Purpose

This program will create a location file with empty records. It should be run before any attempt is made to enter records in that file; ie, before any of the programs that move citation records are executed: **SORT**, **APPEND**, **ROUTE**, or **TICEDT**. If the location file does not exist, it will be created.

4.2. Audience

BLDLOC will normally be run by the system manager.

4.3. Areas

BLDLOC will operate correctly in any area. It will always build an empty location file in the master area.

4.4. Terminals

BLDLOC will operate correctly on any terminal type, including both display screen terminals and hard copy (line-by-line) terminals.

4.5. Files

4.5.1. Input Files

There are no input files.

4.5.2. Output Files

The location file, **LOCTN.DAT**, is the only output file.

4.5.3. Recovery Files

There are no recovery files. If the program doesn't complete normally, simply rerun it.

4.5.4. Authority Files

There are no authority files.

4.6. Dialog

1. **BLDLOC** will normally be run using a mic file, which establishes the proper environment and invokes the program. When ready to proceed, enter **RETURN**.

User:	.do bldloc
BLDLOC:	.silence
	Press RETURN to start
	*
User:	@

2. When finished, **BLDLOC** announces this fact, and quits.

BLDLOC:	The location file has been initialized
---------	--

4.7. Error Messages

There are no error messages peculiar to **BLDLOC**.

5. CDUMP - Dump an ASCII File on the Terminal

5.1. Purpose

This program will dump a file, assumed to be in ASCII format, giving both ASCII representation and octal representation. Each segment of text occupies two lines; segments are separated by a blank line. Characters are printed in ASCII on the first line; unprintable characters are given a two or three letter code (such as 'RS', 'CR', or 'NUL'). Each character is given in octal immediately under the ASCII version. A character count is given on each line.

5.2. Audience

CDUMP is intended to be used by the LBL System Development staff.

5.3. Areas

CDUMP may be run from any area.

5.4. Terminals

CDUMP will operate correctly on any terminal type, including both display screen terminals and hard copy (line-by-line) terminals. Generally, it will be run on hard copy terminals.

5.5. Files

5.5.1. Input Files

There is one input file — namely, the file to be dumped.

5.5.2. Output Files

There are no output files — all output is directed to the terminal.

5.5.3. Recovery Files

There are no recovery files, since there is no recovery procedure. If the program does not run to completion, it must be restarted.

5.5.4. Authority Files

There are no authority files.

5.6. Dialog

1. CDUMP is normally run directly, not from a mic file, so the proper area must be filled in on the execute line. When ready to proceed, enter *RETURN*.

User:	.run cdump[proj,prog]
CDUMP:	Press RETURN to begin
	*
User:	@

2. You will now be asked for the name of the file that is to be dumped. The file must be located in the current area. In the example, the file is PATH.MIC.

CDUMP:	Enter file name
User:	path.mic

3. You will now be asked for a screen width. Most terminals have 79 or 80 usable columns; the TI Omni 820 has 132.

CDUMP:	Enter screen width
User:	80

4. The file is now dumped. This continues until the end of file is reached, or the user terminates the process with a *CTRL-C*. The dump gives the character number of the first character shown on a line at the left of that line. For example:

1	056	160	141	164	150	040	163	167	164	072	057	163	145	141
15	162	143	150	075	133	063	063	060	062	054	063	060	061	054
29	163	167	164	135	015	012	056	160	141	164	150	040	164	151
43	143	072	075	144	157	145	072	133	063	063	060	062	054	063
57	060	061	135	015	012	000	000	000	000	000	000	000	000	000
End of file														

5.7. Error Messages

error in open

The input file cannot be opened. The program terminates.

6. CLNUP — Cleanup a Completed Citation File

6.1. Purpose

This program will copy a citation file, deleting any 780 or 781 fields, and deleting, from the location file, any reference to the records. It is to be run when all processing of the file has been completed, and the file is ready to be sent on to RECON.

Optionally, the program may be used to delete, from the location file, any references to records in the input citation file, without creating a new file.

6.2. Audience

CLNUP will normally be run by the system manager.

6.3. Areas

CLNUP will operate correctly from any area.

6.4. Terminals

CLNUP will operate correctly on any terminal type, including both display screen terminals and hard copy (line-by-line) terminals.

6.5. Files

6.5.1. Input Files

The input file is a named citation file. It is *not* deleted.

6.5.2. Output Files

The output file is also a named citation file, identical to the input file except that all 780 and 781 fields have been deleted.

6.5.3. Recovery Files

A state file, CLNUP.STA, is used for recovery.

6.5.4. Authority Files

There are no authority files.

6.6. Dialog

1. CLNUP is normally run using a mic file, which establishes the proper environment and invokes the program. When ready to proceed, enter *RETURN*.

User:	.do clnup
CLNUP:	.silence
	Press RETURN to begin
	*
User:	@

2. There are two possible responses at this point, depending on whether the program is in recovery mode or not. If CLNUP is in recovery mode, the following is displayed. After recovery is complete to the point at which the program crashed, it proceeds as in step 7.

CLNUP:	— Recovering from abort or system crash
--------	---

3. If **CLNUP** is not in recovery mode, it requests the name of the input file. The file must be in the current area.

CLNUP:	Input file name:
User:	savabc.dat

4. If the specified input file does not exist, you will be told, and asked for the file name again.

CLNUP:	— File 'xxxxxx.xxx' doesn't exist
	— Please try again

5. Now, the output file name is requested. You may respond in either of the following ways.

- a. To copy the input file to an output file, just enter the file name.

CLNUP:	Output file name:
User:	alldon.dat

- b. To delete, from the location file, all references to the citation records in the input file, just type *RETURN*.

CLNUP:	Output file name:
User:	@

6. If the requested output file from step 5a already exists, you will be asked whether the input file should be appended to the output file, or overwrite it. Enter *a* for append, *w* for overwrite, and *CTRL-C* for neither.

CLNUP:	File already exists: type 'a' to append, 'w' to overwrite
User:	a

7. Each record of the input file is now processed in the following way.

- a. If an output file exists, fields 780 and 781 are deleted and the record is written to the output file.
- b. In either case, any reference to the record is deleted from the location file.

8. When all finished, the following message is displayed.

CLNUP:	all done
--------	----------

6.7. Error Messages

- A record from tape xxxxxx had no serial number
This is a serious omission, so please inform the system manager. The program continues.
- Can't access location file
The location file could not be opened. If this message is repeated, you should terminate the program with a *CTRL-C*, and consult the system manager.
- Can't append to state file
- Please consult the system manager
The program terminates abnormally in this case.
- Can't create 'xxxxxx.xxx'
The requested output file does not exist, and could not be created. The program attempts to continue, but probably can't. You should consult the system manager.
- Error reading one of the records.
A citation record from the input file could not be read. You should tell the system manager about this. The program continues.
- Error writing one of the records.
A citation record could not be written in the output file. The program continues, but the output file will lack this record. You should consult the system manager.
- It turns out that you don't have write access to xxxxxx.xxx
You are attempting to write to a file that is write-protected. You will be re-prompted for an output file name.
- Record with tape number xxxxx and serial number ssssssssss was not in the location file
The indicated record has no reference in the location file. This should be considered an information message; the program continues. If this message is printed for each record in the input file, then the input file has probably been previously processed by CLNUP, or was never in the TIC editing system.
- The recovery input file doesn't exist
- Please consult the system manager
Somehow, this file has been deleted. The program aborts.
- The recovery output file doesn't exist
- Please consult the system manager
Somehow, this file has been deleted. The program aborts.
- The state file exists, but is empty
- Please consult the system manager
The program aborts.
- Unexpected EOF
This probably indicates noise on the communication line. CLNUP will re-prompt for the file name.

7. COMPCT – Compact Scattered Files

7.1. Purpose

This program will compact a set of RATFOR files into a few moderately large files. The program will read a command file ('cmd') to obtain the names of files to be gathered together. Every entry in the command file that has extension '.for' is examined for copying. If that file name with extension '.rat' can be found in the current area, that is used. Otherwise, the entry is skipped.

7.2. Audience

COMPCT is intended to be used by the LBL System Development Staff.

7.3. Areas

COMPCT will operate correctly in any area.

7.4. Terminals

COMPCT will operate correctly on any terminal type, including both display screen terminals and hard copy (line-by-line) terminals.

7.5. Files

7.5.1. Input Files

There are two types of input files. The primary input is a command file, containing the names of other files that are to be examined. Secondary inputs are the files indicated in the command file.

7.5.2. Output Files

One or more output files are created, depending on how much input is available. You will specify a nominal length for these files, and the first five characters of the file name. Actual file names are determined by adding the letters 'a', 'b', . . . , 'z' to the name you specified. Thus, a potential of 26 compressed files is allowed. For example, if 'vwxyz.rat' is specified as file name, actual file names might be 'vwxyza', 'vwxyzb.rat', and 'vwxyzc.rat'. All extensions are '.rat'.

A table-of-contents file is created, giving the contents of each output file. The file name is the five characters you specified as output file name, with extension '.toc'. For the example above, the table of contents file name would be 'vwxyz.toc'.

7.5.3. Recovery Files

There are no recovery files, since there is no recovery procedure. If the program does not run to completion, it must be restarted.

7.5.4. Authority Files

There are no authority files.

7.6. Dialog

1. COMPCT is normally run directly, not from a mic file, so the proper area must be filled in on the execute line. When ready to proceed, enter *RETURN*.

User:	.run compct[proj,prog]
COMPCT:	Press RETURN to start
	*
User:	@

2. You will now be asked for the first five characters of the output file name. The name must have exactly five characters — no more and no fewer.

COMPCT:	Enter first 5 characters of output file names (no blanks) —
User:	thsgf

3. You will now be asked for the name of the command file and the approximate output file size. Each output file will contain about this many lines — input files are not split between output files, so the number cannot be exact. A value of 1000-3000 is recommended.

COMPCT:	Enter full name of command file
User:	thsgf.cmd
COMPCT:	Enter approx. lines per output file desired —
User:	2000

4. **COMPCT** now copies the files named in the command file to the newly created output files. As each file is copied, a message is displayed giving the size of the input file, and cumulative size of the output file. For example:

File thsgra.rat opened

File thsm.rat	(191 lines) was copied to thsgra.rat (191 lines)
File auxout.rat	(156 lines) was copied to thsgra.rat (347 lines)
File auxpos.rat	(143 lines) was copied to thsgra.rat (490 lines)
File auxwrt.rat	(122 lines) was copied to thsgra.rat (612 lines)
File clrdi.rat	(13 lines) was copied to thsgra.rat (625 lines)
File collct.rat	(195 lines) was copied to thsgra.rat (820 lines)
File compos.rat	(150 lines) was copied to thsgra.rat (970 lines)
File flblnk.rat	(36 lines) was copied to thsgra.rat (1006 lines)
File getcmd.rat	(100 lines) was copied to thsgra.rat (1106 lines)
File help.rat	(24 lines) was copied to thsgra.rat (1130 lines)
File initds.rat	(155 lines) was copied to thsgra.rat (1285 lines)
File inittr.rat	(74 lines) was copied to thsgra.rat (1359 lines)
File mterm.rat	(135 lines) was copied to thsgra.rat (1494 lines)
File mtsrch.rat	(37 lines) was copied to thsgra.rat (1531 lines)
File next.rat	(181 lines) was copied to thsgra.rat (1712 lines)
File nxtmt.rat	(42 lines) was copied to thsgra.rat (1754 lines)
File postns.rat	(167 lines) was copied to thsgra.rat (1921 lines)
File prxsch.rat	(34 lines) was copied to thsgra.rat (1955 lines)
File rdtrm.rat	(77 lines) was copied to thsgra.rat (2032 lines)

File thsgrb.rat opened

File recall.rat	(31 lines) was copied to thsgrb.rat (31 lines)
File roll.rat	(146 lines) was copied to thsgrb.rat (177 lines)
File split.rat	(61 lines) was copied to thsgrb.rat (238 lines)
File stems.rat	(100 lines) was copied to thsgrb.rat (338 lines)
File tput.rat	(18 lines) was copied to thsgrb.rat (356 lines)
File trmout.rat	(101 lines) was copied to thsgrb.rat (457 lines)
File trmwrt.rat	(31 lines) was copied to thsgrb.rat (488 lines)
File uprcas.rat	(25 lines) was copied to thsgrb.rat (513 lines)
File ulcase.rat	(46 lines) was copied to thsgrb.rat (559 lines)
File wrtdis.rat	(71 lines) was copied to thsgrb.rat (630 lines)
File wrtrol.rat	(91 lines) was copied to thsgrb.rat (721 lines)
File string.rat	(272 lines) was copied to thsgrb.rat (993 lines)
File thes.rat	(851 lines) was copied to thsgrb.rat (1844 lines)
File subs.rat	(140 lines) was copied to thsgrb.rat (1984 lines)

5. The table of contents file for the example just shown looks as follows:

Output file thsgra.rat contains:

thsm.rat
auxout.rat
auxpos.rat
auxwrt.rat
clrdis.rat
collect.rat
compos.rat
flblk.rat
getcmd.rat
help.rat
initds.rat
inittr.rat
mterm.rat
mtsrch.rat
next.rat
nxtmt.rat
postns.rat
prxsch.rat
rdtrm.rat

Output file thsgrb.rat contains:

recall.rat
roll.rat
split.rat
stems.rat
tput.rat
trmout.rat
trmwrt.rat
uprcas.rat
ulcase.rat
wrtdis.rat
wrtrol.rat
string.rat
thes.rat
subs.rat

7.7. Error Messages

There are no error messages.

8. CPYCIT - Copy Citation Records

8.1. Purpose

This program will copy citation records from one or more input files, to build a new citation file.

8.2. Audience

CPYCIT may be used by anyone; however, we expect it will be used primarily by the LBL System Development Staff, and by the system manager.

8.3. Areas

CPYCIT will operate correctly from any area.

8.4. Terminals

CPYCIT will operate correctly on any terminal type, including both display screen terminals and hard copy (line-by-line) terminals.

8.5. Files

8.5.1. Input Files

The input files are one or more named citation files.

8.5.2. Output Files

There is a single named output file, in citation file format.

8.5.3. Recovery Files

There are no recovery files, since there is no recovery procedure. If the program does not run to completion, it must be restarted.

8.5.4. Authority Files

There are no authority files.

8.6. Dialog

1. CPYCIT is normally run using a mic file. The mic file establishes the proper environment and invokes the CPYCIT program. When ready to proceed, enter *RETURN*.

User:	.do cpycit
CPYCIT:	.silence
	Type RETURN to begin
	*
User:	@

2. CPYCIT requests the name of the output file. You should provide the file name and extension. The file must be located in the current area.

CPYCIT:	Output file name:
User:	newfil.dat

3. If the specified file does not exist, it is created. If it does exist, you will be informed of that fact, and asked whether you wish to append citation records to that file, or to overwrite the existing records. You should enter *a* to append, *o* to overwrite, or *CTRL-C* to halt the program (in case you wish to do neither). Do not!! hit *RETURN* after the *a* or *q*, since it will be interpreted as a response to the question in step 4, and the program will terminate.

CPYCIT:	This file already exists. Do you want to append to it or overwrite it? Type 'a' to append, 'o' to overwrite.
User:	a [Do not hit RETURN]

4. You are now asked for the name of an input file. You may respond in either of the following ways:
- a. If you wish to copy records from an input file, enter the name of the file.

CPYCIT:	Input File Name:
User:	cat40a.dat

- b. If you wish to terminate the program, just type RETURN.

CPYCIT:	Input File Name:
User:	@

5. If the specified input file does not exist, you will be told, and asked for the file name again.

CPYCIT:	That file does not exist — please try again
---------	---

6. There is now a pause while **CPYCIT** locates all of the records in the current input file. It then tells you how many records are in the file.

CPYCIT:	There are nn records
---------	----------------------

7. **CPYCIT** now asks which records are to be copied. You may respond in any of the following ways.

- a. If you wish to copy just one record, enter the record number. **CPYCIT** will then copy the record, and ask for another set of records. In the example, record number 5 is copied.

CPYCIT:	Records to copy:
User:	5

- b. If you wish to copy a range of records, enter the record numbers of the first and last record in the range, separated by a hyphen (minus sign, or dash). **CPYCIT** will then copy the records, and ask for another set of records. In the example, records 10 through 15, inclusive, are copied.

CPYCIT:	Records to copy:
User:	10-15

- c. If you wish to copy from some point to the end of the input file, enter the record number of the first record to be copied, followed by a hyphen and an asterisk. **CPYCIT** will then copy the records, and ask for another set of records.

CPYCIT:	Records to copy:
User:	23-*

- d. You may include several ranges as described in steps 7a-7c on one input line by separating them by commas.

CPYCIT:	Records to copy:
User:	5,10-15,23-*,8-11

- e. When finished with the current input file, just type RETURN. Control then returns to step 4 and requests a new file name.

CPYCIT:	Records to copy:
User:	@

8.7. Error Messages

Can't write to this file
Please choose another

The output file you specified cannot be opened for output. Probably it is write-protected. You will be prompted again for an output file name.

-- error reading record nn

The specified record cannot be read from the input file. **CPYCIT** continues.

-- error rereading record nn -- record not written

The specified record could not be read from the input file, so could not be written to the output file. **CPYCIT** continues.

-- error writing record nn

The specified record could not be written to the output file. **CPYCIT** continues.

?? Impossible error in cpycit — please tell the system manager

This should never occur. If it does, something serious is wrong. **CPYCIT** is terminated.

Invalid character ignored

A record number specified in step 7 contains a non-numeric character. It is just ignored, and **CPYCIT** continues.

?? No dynamic memory available — please tell the system manager

This should never occur. If it does, something serious is wrong. **CPYCIT** is terminated.

-- record nn has an error in it and can't be written to the output file

The specified record could not be written. **CPYCIT** continues.

?? There are too many records for cpycit to process

This should never occur. If it does, something serious is wrong. **CPYCIT** is terminated.

Unexpected EOF — Job Aborted

The specification for the output file name is incorrect; there may have been noise on the line. **CPYCIT** terminates — you may rerun it.

9. DECODE — Dump a Non-ASCII File on the Terminal

9.1. Purpose

This program will dump a file that is not assumed to be in ASCII format. Each word is given in octal, with a blank between the left and right halves. An attempt is made to print the word in ASCII immediately to the right of the octal form; bytes that do not represent printable ASCII characters cause percent signs ('%') to be printed. A word count is given for each line.

9.2. Audience

DECODE is intended to be run by the LBL System Development staff.

9.3. Areas

DECODE will operate successfully in any area.

9.4. Terminals

DECODE will operate correctly on any terminal type, including both display screen terminals and hard copy (line-by-line) terminals. However, it will normally be used with hard copy terminals.

9.5. Files

9.5.1. Input Files

The single input file is the file you wish to dump on the terminal.

9.5.2. Output Files

There are no output files — all output is directed to the terminal.

9.5.3. Recovery Files

There are no recovery files, since there is no recovery procedure. If the program does not run to completion, it must be restarted.

9.5.4. Authority Files

There are no authority files.

9.6. Dialog

1. DECODE is normally run directly, not from a mic file, so the proper area must be filled in on the execute line. When ready to proceed, enter *RETURN*.

User:	.run decode[proj,prog]
DECODE:	Press RETURN to begin
	*
User:	@

2. You will now be asked for the name of the input file, and a screen width. Most terminals have 80 usable columns; the TI Omni 820 has 132.

DECODE:	Enter file name
User:	path.mic
DECODE:	Enter screen width
User:	60

3. The file is now dumped, as follows:

1	273414	172320	.path	203476	772164	swt:
3	277474	560744	/sear	617207	555546	ch=[3
5	315406	226146	302,3	301425	471756	01,sw
7	722721	505134	t]%%.	703036	464100	path
9	723234	335172	tic: =	623374	535266	doe:[
11	315466	031130	3302,	315406	156432	301]%
13	050000	000000	%%%%%%%%	000000	000000	%%%%%%%%
End of file						

9.7. Error Messages

error in open

The input file cannot be opened. The program terminates.

10. LISTMT - List Main Terms from the EDB Thesaurus

10.1. Purpose

This program will start at a specified point in the thesaurus fine directory and list main terms until the end-of-file is reached or the program is stopped with a *CTRL-C*. The terms are listed in one or two columns, depending on paper width.

10.2. Audience

LISTMT is intended to be used by everyone.

10.3. Areas

LISTMT will operate correctly from any area.

10.4. Terminals

LISTMT will operate correctly on any terminal type, including both display screen terminals and hard copy (line-by-line) terminals.

10.5. Files

10.5.1. Input Files

The only input files are the thesaurus fine and coarse directories, located in the master area.

10.5.2. Output Files

There are no output files — all output goes directly to the terminal.

10.5.3. Recovery Files

There are no recovery files, since there is no recovery procedure. If the program does not run to completion, it must be restarted.

10.5.4. Authority Files

There are no authority files.

10.6. Dialog

1. LISTMT is normally run directly, not from a mic file, so the proper area must be filled in on the execute line. When ready to proceed, enter *RETURN*.

User:	.run listmt[proj,prog]
LISTMT:	Press RETURN to begin
	*
User:	@

2. You will now be asked for a screen width. Most terminals have 79 or 80 usable columns; the TI Omni 820 has 132.

LISTMT:	Enter screen size (eg, 80, 120) -
User:	70

3. You will now be asked for a starting place. Enter a string of characters that will identify the first main term to be displayed. The string need not be a main term itself; the first term that lexicographically follows the specified string is used as the actual starting place. If you enter the null string (ie, just enter *RETURN*), the first term to be printed will be the first term in the thesaurus.

LISTMT:	Enter starting place
User:	aci

4. Main terms are now listed. Each term is numbered, starting from 1.

LISTMT:	1	Acid Anhydrases
	2	Acid Carbonates
	3	Acid Chrome Dyes
	4	Acid Electrolyte Fuel Cells
	5	Acid Halides
	6	Acid Hydrolysis
	7	Acid Mine Drainage
	8	Acid Phosphatase
	9	Acid Phosphates
	10	Acid Proteinases
	11	Acid Rain
	12	Acid Silicates
	13	

10.7. Error Messages

Screen must have at least 69 columns, and no more than 132 columns

The screen size specified is too small; the program terminates.

The following term has nnn characters

<<the term is given here>>

A main term is found with too many characters. As many characters as possible are printed, and the program continues.

11. LISTPI - List Words from the Inverted Index

11.1. Purpose

This program will list the words in the inverted index from some point on, with the main terms that contain the words in the index. The program continues to list words and terms until it reaches the end of the inverted index, or it is stopped with a *CTRL-C*.

11.2. Audience

LISTPI is intended to be run by anyone.

11.2.1. Areas

LISTPI will operate correctly in any area.

11.3. Terminals

LISTPI will operate correctly on any terminal type, including both display screen terminals and hard copy (line-by-line) terminals.

11.4. Files

11.4.1. Input Files

The thesaurus files and inverted index files are used.

11.4.2. Output Files

There are no output files — all output is directed to the terminal.

11.4.3. Recovery Files

There are no recovery files, since there is no recovery procedure. If the program does not run to completion, it must be restarted.

11.4.4. Authority Files

There are no authority files.

11.5. Dialog

1. LISTPI is normally run directly, not from a mic file, so the proper area must be filled in on the execute line. When ready to proceed, enter *RETURN*.

```
-----
User:  .run listpi[proj,prog]
LISTPI: Press RETURN to begin
      *
User:  @
-----
```

2. You will now be asked to enter the starting place in the inverted index. Enter a character string that identifies the first word to be displayed. You need not enter an actual word; the first word that lexicographically follows the specified string is used as the starting place. If you enter the null string (ie, just enter *RETURN*), the first word in the inverted index is listed.

LISTPI:	Enter starting place
User:	yam

3. The program now lists words from the inverted index, followed by all main terms that include that word.

LISTPI:	YAMAGUCHI
	Yamaguchi Nonlocal Potential
	Yamaguchi Potential
	YAMS
	Yams
	YANG
	Lee-Yang Theory
	Yang Theorem
	Yang-Feldman Formalism
	Yang-Mills Theory
	YANG-LEE
	Yang-Lee Distribution
	YANG-MILLS
	Yang-Mills Theory
	YANGTZE
	Yangtze River
	YANKEE
	Atomic Power Company Main Yankee
	Connecticut Yankee Reactor
	Maine Yankee Reactor
	Rowe Yankee Reactor
	Vermont Yankee Reactor
	Yankee Connecticut Reactor
	

11.6. Error Messages

There are no error messages

12. LOCATE - Find Locations of Citation Records

12.1. Purpose

This program will examine the location file, and report the current file locations of specified citation records. You may specify the serial number, tape number, or both.

- If only a serial number (field 010 of the citation record) is entered, the location of all citation records that have matching serial numbers are displayed. (Normally, there will be only one such record.)
- If only a tape number is entered, the locations of all citation records that have matching tape numbers are displayed, sequenced by serial number.
- If both a serial number and a tape number are entered, the location of the (one and only) matching citation record is displayed.

12.2. Audience

LOCATE is intended to be run by everyone.

12.3. Areas

LOCATE will operate correctly in any area.

12.4. Terminals

LOCATE will operate correctly on any terminal type, including both display screen terminals and hard copy (line-by-line) terminals.

12.5. Files

12.5.1. Input Files

The only input file is the location file, contained in the master area.

12.5.2. Output Files

There are no output files — all output goes directly to the terminal.

12.5.3. Recovery Files

There are no recovery files — since LOCATE does not alter any files, no recovery procedures are necessary.

12.5.4. Authority Files

No authority files are required.

12.6. Dialog

1. LOCATE is normally run using a mic file, which establishes the proper environment and invokes the program. When ready to proceed, enter *RETURN*.

User:	.do locate
LOCATE:	.silence
	Press RETURN to start
	*
User:	@

2. **LOCATE** will now request a serial number and a tape number. You may enter either or these, both, or neither.

- a. If both a serial number and a tape number are entered, **LOCATE** will display location file information for the single matching record. A serial number is a ten digit number given in field 010 of the citation record. A tape number has the form aammddyy, where 'aa' is a two letter code identifying the originating agency, and 'mmddyy' is a date.

LOCATE:	Enter a serial number, or just RETURN						
User:	8108122265						
LOCATE:	Enter a tape number, or just RETURN						
User:	ax082781						
LOCATE:	Tape	Serial	Date	Hour	File name	Proj,prog	
	AX082781	8108122265	09-15-82	11	cat55a.dat	7,301	

- b. If just a serial number is entered, **LOCATE** will display location file information for all matching records. Normally, there will be only one such record, but we allowed for the possibility of two or more citation records accidentally being assigned the same serial number.

LOCATE:	Enter a serial number, or just RETURN						
User:	8108122265						
LOCATE:	Enter a tape number, or just RETURN						
User:	@						
LOCATE:	Tape	Serial	Date	Hour	File name	Proj,prog	
	AX082781	8108122265	09-15-82	11	cat55a.dat	7,301	

- c. If just a tape number is entered, **LOCATE** will display location file information for all matching records. There will be 100 such records much of the time.

LOCATE:	Enter a serial number, or just RETURN						
User:	@						
LOCATE:	Enter a tape number, or just RETURN						
User:	ax082781						
LOCATE:	Tape	Serial	Date	Hour	File name	Proj,prog	
	AX082781	8108122260	09-15-82	11	cat40a.dat	7,306	
	AX082781	8108122261	09-17-82	16	cat45a.dat	7,302	
	AX082781	8108122262	09-15-82	11	catunk.dat	7,306	
	etc.						

- d. If neither serial number nor tape number are entered, the program quits; this is the proper way to terminate (although just using *CTRL-C* works as well, and doesn't hurt anything).

LOCATE:	Enter a serial number, or just RETURN
User:	@
LOCATE:	Enter a tape number, or just RETURN
User:	@

3. The cycle through step 2 continues over and over as long as you wish.

12.7. Error Messages

There are no error messages peculiar to **LOCATE**.

13. PFILE — Print Files on TI OMNI 820 Terminal

13.1. Purpose

This program will print one or more files on the Texas Instruments Omni 820 Terminal. Each page has a page header consisting of page number, date, and file name. Pages are properly advanced, so the printing does not cross the perforations.

If a file name other than a .cmd file is specified, that file is printed. If a .cmd file is specified, all files named in the .cmd file that occur in the current area are printed. In the latter case, the RATFOR versions of FORTRAN files are printed, not the FORTRAN versions. Ie, if 'ticedt.for' occurs in a .cmd file, 'ticedt.rat' is printed.

Files that cannot be found in the current area are omitted. Thus, library routines in .cmd files are not usually printed (since they usually don't have extensions, or occur in other areas).

13.2. Audience

PFILE will normally be run by the LBL System Development Staff. However, anyone with access to the proper terminal may use the program.

13.3. Areas

PFILE may be run in any area, so long as the specified file is also in the area.

13.4. Terminals

PFILE will operate correctly only from a TI Omni 820 terminal.

13.5. Files

13.5.1. Input Files

The input file may be a single file, or a command file and all files specified therein.

13.5.2. Output Files

There are no output files — all output is directed to the terminal.

13.5.3. Recovery Files

There are no recovery files, since there is no recovery procedure. If the program does not run to completion, it can simply be restarted.

13.5.4. Authority Files

There are no authority files.

13.6. Dialog

13.6.1. Single File Mode

1. PFILE is normally run using a mic file, which establishes the proper environment and invokes the program. When ready to proceed, enter *RETURN*.

User:	.do pfile
PFILE:	.silence
	Press RETURN to start
	*
User:	@

2. You will now be asked for a file name. In single-file mode, you may specify any text file with extension other than '.cmd'. For example, 'ticedt.rat', 'gdandt.mac', 'sort.mic'.

PFILE:	Please enter file name
User:	name.ext

3. At this point, a page eject is sent to the terminal, and the file is printed, 55 lines per page, with a heading containing the page number, date, and file name.

13.6.2. Multiple File Mode

1. **PFILE** is normally run using a mic file, which establishes the proper environment and invokes the program. When ready to proceed, enter *RETURN*.

User:	.do pfile
PFILE:	.silence
	Press RETURN to start
	*
User:	@

2. You will now be asked for a file name. In multiple-file mode, you should specify a command file, with extension '.cmd'.

PFILE:	Please enter file name
User:	name.cmd

3. Each line of the command file is examined. If the line does not contain an extension, the program ignores it. If the extension is '.for', it is changed to '.rat'. Next, a page eject is sent to the terminal, and the file is printed, 55 lines per page, with a heading containing the page number, date, and file name.

13.7. Error Messages

Illegal file name

The file specified by the user cannot be opened; the program terminates.

14. PRTCDR — Print Thesaurus Coarse Directory

14.1. Purpose

This program prints a formatted listing of the thesaurus coarse directory on the terminal. A screen size may be specified, so that the listing will be properly formatted for an ordinary terminal or the 132 column TI Omni 820.

14.2. Audience

PRTCDR was written for LBL System Staff use, but may be run by anyone.

14.3. Areas

PRTCDR will operate correctly in any area.

14.4. Terminals

PRTCDR will operate correctly on any terminal, including both display screen terminals and hard copy (line-by-line) terminals. It is more useful on hard copy terminals, however. The program assumes a page size of 66 lines (11" paper).

14.5. Files

14.5.1. Input Files

The only input file is the thesaurus coarse directory, THESAU.CDR.

14.5.2. Output Files

There are no output files — all output goes directly to the terminal.

14.5.3. Recovery Files

There are no recovery files.

14.5.4. Authority Files

There are no authority files

14.6. Dialog

1. PRTCDR is normally run directly, not from a mic file, so the proper area must be filled in on the execute line. When ready to proceed, enter *RETURN*.

User:	.run prtcdr[proj,prog]
PRTCDR:	Press RETURN to begin
	*
User:	@

2. You will now be asked for a screen width. Most terminals have 79 or 80 usable columns; the TI Omni 820 has 132. The program then pauses for the paper to be positioned. This can be ignored for display terminals; type *G* to continue. On the Omni 820 (or other hard copy terminal), type *G*, page restore, and *RETURN*.

PRTC DR:	Enter screen width (eg. 80, 120) -
User:	80
PRTC DR:	Position paper and type G
	PAUSE
	Type G to continue, X to Exit, T to Trace
User:	G

3. The coarse directory is now printed, 54 lines per page, followed by 12 blank lines (for a total of 66 lines per page). If the screen width was set to 80, there will be five columns. If it was set to 132, there will be 8 columns.

1	1,1-D	55	AQUEO	109	BOHUN
2	1,4-D	56	ARCHA	110	BOLIV
3	2-MET	57	ARGON	111	BONDI
4	8-HYD	58	ARGON	112	BORAT
5	ABAND	59	ARGON	113	BORON
..	...	..	...	..	...

14.7. Error Messages

There are no error messages

15. ROUTE — Manual Dispersal of Citation Records

15.1. Purpose

This program will read a file (usually the “unknown category” file), and distribute the records to category files under the direct control of the user.

15.2. Audience

ROUTE will normally be run by a supervisor or the system manager.

15.3. Areas

ROUTE will operate correctly in any area that contains the authority files. It will normally be run in the master area.

15.4. Terminals

ROUTE will operate correctly on any of the following terminal types: ADM 3A, ADM 31, Hazeltine 1552, and DEC VT 100. Highlighting does not work on the ADM 3A, since that terminal does not have the facility.

15.5. Files

15.5.1. Input Files

There is one input file, in Citation File Format. Normally, this will be CATUNK.DAT, a file of “unknown category” records. The file is deleted upon normal termination.

15.5.2. Output Files

Citation records are placed in one of forty category files. You **must** give a category number for each record in the input file. Category files are named CATxxA.DAT, where ‘xx’ is the category number. (For example, CAT43A.DAT, CAT65A.DAT.) If these files already exist, ROUTE will append citation records to the files. If any do not exist, they will be created.

15.5.3. Recovery Files

A state file, STATE.RTE, is created by ROUTE to keep track of its progress. If ROUTE does not complete (due to a system crash, use of CTRL-C, user request, or whatever), this file is used to recover and continue processing. This file is deleted when ROUTE terminates normally.

15.5.4. Authority Files

One file is used to provide data to ROUTE, and it must be in the current area:

CODES.LIS contains a list of valid first-level category codes.

The inverted index files are used from the master area — no special provision needs to be made for these.

15.6. Dialog

1. ROUTE will normally be run in the master area using a mic file. The mic file establishes the proper environment and invokes the ROUTE program. When ready to proceed, enter *RETURN*.

User:	.do route
ROUTE:	.silence
	.
	Type RETURN to begin
	*
User:	@

2. There are several possible responses at this point, depending on whether the program is in recovery mode or not.

- a. If **ROUTE** is not in recovery mode, it now requests the file name. Enter the name of the citation file you wish to have distributed to category files. Normally, this will be CATUNK.DAT, as in the example, but **ROUTE** will work correctly on other files as well. If the file cannot be found, **ROUTE** will request another file name (this is not shown).

ROUTE:	Input file name:
User:	catunk.dat

- b. If **ROUTE** has discovered that it did not complete the previous run, and goes into recovery mode, the following is displayed. The program cannot be used on a new file until the previous file has been completely processed. After this message, the program proceeds as in step 3.

ROUTE:	[Recovering]
--------	--------------

- c. If a **SORT** state file exists, **ROUTE** cannot be run. You must run **SORT** to completion before restarting **ROUTE**. The following messages are displayed, and the program terminates.

ROUTE:	A SORT state file exists, so ROUTE cannot be run. You must run SORT to completion before restarting ROUTE.
--------	---

- d. Likewise, if a state file for **APPEND** exists, **ROUTE** cannot be run.

ROUTE:	An APPEND state file exists, so ROUTE cannot be run. You must run APPEND to completion before restarting ROUTE.
--------	--

3. **ROUTE** now begins (or continues) to process the input file. The title and abstract of each record is displayed on the screen, with all words found in the inverted index to the thesaurus highlighted. (There is no highlighting on the ADM 3A terminal, since it does not have this facility.) You are asked which category code should be assigned to this record; enter the two digit code (50 in the example). If you wish to delete the record, enter *d*. If you wish to temporarily quit, enter *q*.

ROUTE:	Title:	ARPS submission to Commission of Enquiry into Occupational Safety and Health in NSW
	Abstract:	This ARPS submission relates to matters concerning the use of radiation in New South Wales . The N.S.W. Radioactive Substances Act 1957 and The Radioactive Substances Regulations 1979 , as amended, are now significantly out of date. Recommendations for amendments to the legislation are given. Other areas included are the lack of a comprehensive policy in N.S.W. regarding the training requirements for persons who are to work with radiation , the employment of radiation protection officers, administering control over the uses of radiation and the importance of a national radioactive waste management policy .
User:		'd' to delete, 'nn' to route to category nn, 'q' to quit. 50

4. When all records have been distributed, the program quits.

ROUTE:	Done
--------	------

15.7. Error Messages

Can't write to that category file. Please try again.

For some reason, the selected category file could not be written. The program will request a different category number.

Category number out of range. Please try again.

The category number must be a number between 0 and 99.

Invalid number. Please try again.

Please enter a valid category number.

Invalid response. Please try again.

You must enter 'd', 'q', or a legitimate category number.

That category doesn't exist. Please try again.

The number you specified is not that of a legitimate category.

There has been a serious error.

Can't read the codes.lis file from the master area.

Please inform the appropriate people.

This file is required for the program to run. Please inform the system manager that it has apparently been deleted by mistake. **ROUTE** cannot proceed until that file is available.

There has been a serious error.

Can't write state file.

Please inform the appropriate people.

Please tell the system manager.

There has been a serious error.
Fatal error writing to category file.
Please inform the appropriate people.

Please tell the system manager that the category file has probably been damaged.

There has been a serious error.
Fatal error writing output file.
Please inform the appropriate people.

Recovery is impossible. Please inform the system manager.

There has been a serious error. The input file has been deleted. Please inform the appropriate people.

The input file that was last in use when **ROUTE** crashed has disappeared. Please inform the system manager.

There has been a serious error.
The output file is no longer writable.
Please inform the appropriate people.

The attempted recovery is aborted. Please tell the system manager.

There has been a serious error.
The record being recovered is no longer valid.
Please inform the appropriate people.

The attempted recovery cannot continue. Please tell the system manager.

There has been a serious error.
The state file exists but is empty.
Please inform the appropriate people.

The attempted recovery cannot continue. Please tell the system manager.

This record is unreadable. It will be deleted.
[type a space to continue.]

If the record is needed, please kill the program with *CTRL-C* and tell the system manager.

[Warning — couldn't delete input file.]

The file is still there. If you cannot delete it yourself, please tell the system manager. It is very important to get rid of that file.

[Warning — couldn't delete **ROUTE** state file.]

The file is still there. If you cannot delete it yourself, please tell the system manager.
ROUTE cannot be run again until the state file is eliminated.

16. SEQCIT — Sequence Citation Records

16.1. Purpose

This program will sequence the citation records in a citation file, building a new citation file in the indicated order. Any of the fields in the record may be chosen as the sequence field. Only one field may be chosen, and sequencing is in increasing lexicographic order.

The input file is deleted upon completion of the program.

16.2. Audience

SEQCIT may be used by anyone; however, we expect it will be used primarily by the system manager.

16.3. Areas

SEQCIT will operate correctly from any area.

16.4. Terminals

SEQCIT will operate correctly on any terminal type, including both display screen terminals and hard copy (line-by-line) terminals.

16.5. Files

16.5.1. Input Files

The input file is a named citation file. This file is normally deleted when the program completes.

16.5.2. Output Files

There is a single named output file, in citation file format. It is identical to the input file, except for record order.

16.5.3. Recovery Files

There are no recovery files, since there is no recovery procedure. If the program does not run to completion, it must be restarted.

16.5.4. Authority Files

There are no authority files.

16.6. Dialog

1. SEQCIT is normally run using a mic file. The mic file establishes the proper environment and invokes the SEQCIT program. When ready to proceed, enter *RETURN*.

User:	.do seqcit
SEQCIT:	.silence
	Type RETURN to begin.
	*
User:	@

2. SEQCIT requests the name of the input file. You should provide the file name and extension. The file must be located in the current area. If the file does not exist, you will be informed of

this fact and reprompted for the input file name. (This is not illustrated here.)

SEQCIT:	Input file name:
User:	oldfil.dat

3. You are now asked for the name of an output file. Please give the file name. If the file cannot be used for some reason (there may already be a file with that name in the current area, for example), you will be informed of that fact and reprompted for the output file name. (This is not illustrated here.)

SEQCIT:	Input file name:
User:	newfil.dat

4. You are now asked for the number of the field you wish to use as a sequencing field (ie, sort field). Please enter the proper number. If you wish to use field 550 (source of bibliographic information), just type *RETURN*, since this is the default.

SEQCIT:	Field to sort on (RETURN for 550):
User:	610

5. There is now a pause while **SEQCIT** locates all of the records in the current input file, and sorts them on the specified field. It then tells you how many records are in the file.

SEQCIT:	There are nn records.
---------	-----------------------

6. **SEQCIT** now copies the records from the input file to the output file in the new order. As each set of five records is copied, a progress report is displayed. This is primarily to reassure you that the program is operating correctly.

SEQCIT:	Writing record 5. Writing record 10. Writing record 15.
---------	---

7. When the copy process is finished, a final message is displayed.

SEQCIT:	Done
---------	------

16.7. Error Messages

Error in number: please try again

The number you have given for a field value has a nonnumeric character in it — there may have been noise on the communication line. You will be reprompted for the field number.

-- error reading record nn

A citation record could not be read from the input file. The program attempts to continue, but the indicated record will not be present in the output file.

-- error rereading record nn — record not written

The indicated record could not be read during the copy process. The program continues, but this record will not be present on the output file.

-- error writing record nn

The indicated record could not be written to the output file. The program continues, but this record will not be present on the output file.

Field number must be between 000 and 999.

You have specified a field number that is outside this range. The program will reprompt you for the field number.

?? impossible error in seqcit - please tell the system manager.

You should never see this error message; if you do, the program cannot continue. Please tell the system manager, and have the situation fixed before trying to rerun the program.

? no dynamic memory available — please tell the system manager

The program cannot continue. This should be a temporary condition, so you may wish to rerun the program at a later time. If the message appears again, the system manager will have to fix the situation.

-- record nn has an error in it and can't be written to the output file

The program attempts to continue, but the specified record will not be present on the output file. You may wish to tell the system manager, and have the file fixed.

That file already exists: please choose another.

SEQCIT will not overwrite an existing file. If you really want to reuse a file name, please delete the file before rerunning **SEQCIT**. Otherwise, choose a different output file name.

That file doesn't exist: please choose another

The file you have specified for the input file does not exist. You may have misspelled it, or there may have been noise on the communication line. You will be reprompted for the input file name.

That file is not writable: please choose another.

The file you have specified as the output file could not be created. Choose another file name, or stop **SEQCIT** (with a *CTRL-C*), fix the problem, and then rerun it.

There are too many records for seqcit to process

This error should never be seen. If it does occur, you may wish to wait awhile and retry **SEQCIT**. Otherwise, ask the system manager to fix whatever is wrong, and then rerun the program.

Unexpected EOF — job aborted.

The input file name could not be read. This probably indicates noise on the communication line, so you may wish to try the program again. If the problem doesn't disappear, please ask the system manager to fix the situation.

[Warning: was unable to delete input file.]

The input file protection mode was set to prohibit deletion. This is only a warning message; if you really want the file deleted, you should change the protection code and use **DELETE**.

17. SORT — Distribute Citations to Category Files

17.1. Purpose

This program will read a file of document citation records, and attempt to place each record in a category file. Any records that cannot be properly categorized are added to the "unknown category" file.

To find the EDB category, the program first looks for a Subject Categories field (540). If this field exists in the citation record, and contains a legitimate EDB category, the first two digits are used for routing. If no category can be found this way, the program attempts to use the INIS category (field 610). If this field exists in the record, and contains a legitimate INIS category, the corresponding EDB category is looked up in a table and used. If neither field yields a category number, the record is assigned the "unknown" category.

When all records have been distributed, a report is written both to the terminal screen and to a report file, showing how many records were added to each file.

17.2. Audience

SORT will normally be run by the system manager.

17.3. Areas

SORT will operate correctly in any area, but will normally be run in the master area.

17.4. Terminals

SORT will operate correctly on any terminal type, including both display screen terminals and hard copy (line-by-line) terminals.

17.5. Files

17.5.1. Input Files

There is one named input file, in Citation File Format. Normally, this will be a file of citation records created from magnetic tape received from INIS, API, etc. The file must be in the current area. The file is deleted upon normal termination.

17.5.2. Output Files

Citation records are placed in one of forty category files, or into the "unknown category" file. Category files are named CATxxA.DAT, where 'xx' is the category number. (For example: CAT43A.DAT, CAT65A.DAT.) The unknown category file is named CATUNK.DAT. If these files already exist, **SORT** will append citation records to the files. If any do not exist, they will be created. The files will be placed in the current area.

A copy of the report displayed on the terminal screen at the end of the job is stored in file REPORT.SRT, in the current area. Any previous contents of this file are overwritten.

17.5.3. Recovery Files

A state file, STATE.SRT, is created by **SORT** to keep track of its progress. If **SORT** does not complete (due to system crash, use of *CTRL-C*, or whatever), this file is used to recover and continue processing. This file is deleted when **SORT** terminates normally.

Note that the recovery file is stored in the current area, not the master area.

17.5.4. Authority Files

Two files are used to provide data to **SORT**, and both must be in the master area:

CAT.MAP gives the category mappings from INIS categories to TIC categories.

CAT.NUM gives the legitimate first level category numbers.

17.6. Dialog

1. **SORT** will normally be run in the master area using a mic file. The mic file establishes the proper environment and invokes the **SORT** program. When ready to proceed, enter *RETURN*.

User:	.do sort
SORT:	.silence
	Type RETURN to begin
	*
User:	@

2. There are several possible responses at this point, depending on whether the program is in recovery mode or not.
 - a. If **SORT** is not in recovery mode, it now requests the name of the citation file you wish to have distributed to category files. If the file cannot be found, **SORT** will request another file name (this is not shown here).

SORT:	Enter input file name:
User:	fffff.fff

- b. If **SORT** has discovered that it did not complete the previous run, and goes into recovery mode, the following is displayed. The program cannot be used on a new file until the previous file has been completely processed.

SORT:	RECOVERY
	Processing will continue on file
	name.ext

- c. If an **APPEND** state file exists, **SORT** cannot be run. You must run **APPEND** to completion before restarting **SORT**. The following messages are displayed.

SORT:	An APPEND state file exists, so SORT cannot be run.
	You must run APPEND to completion before restarting SORT.

- d. Likewise, if a state file for **ROUTE** exists, **SORT** cannot be run.

SORT:	A ROUTE state file exists, so SORT cannot be run.
	You must run ROUTE to completion before restarting SORT.

3. **SORT** now begins (or continues) to process the file. As each set of five records is copied from the citation file to a category file, a progress report is displayed. This is primarily to reassure

you that the program is operating correctly.

SORT:	Processing beginning. 5 records copied to category files. 10 records copied to category files. 15 records copied to category files. 20 records copied to category files. etc End of file. xx records copied. Please press RETURN to see records dispersed to categories.
-------	--

The program has completed, and pauses to allow you to take any special action you want before the report is displayed.

4. The report shows how many records were distributed to the various category files. In this example, 3 records went to CAT10A.DAT, etc.

SORT:	Category # records	Category # records
	10 3	55 8
	50 7	unk 2
	End of job	

17.7. Error Messages

File xxxxxx.xxx cannot be opened.

SORT cannot continue.

Please ask the system manager to correct the error.

The indicated file could not be opened. **SORT** must have access to this file to continue, so it terminates. If you cannot provide the file, please ask the system manager for help. He will need to know the file name.

— File xxxxxx.xxx doesn't exist.

— Please try again.

The requested input file does not exist in the current area. Perhaps you have misspelled the name; if so, please re-enter the name. Otherwise, please stop **SORT** with a **CTRL-C**, obtain the file, and start again.

Record yy of file xxxxxx.xxx cannot be read.

SORT cannot continue.

Please ask the system manager to correct the error, and rerun **SORT**.

The indicated record could not be read; please correct the error, if you can, or ask the system manager for help. He will need to know the file name and record number.

Record yy of file xxxxxx.xxx cannot be written.

SORT cannot continue

Please ask the system manager to correct the error, and rerun **SORT**.

The indicated record could not be written; please correct the error, if you can, or ask the system manager for help. He will need to know the file name and record number.

- Record yy of file xxxxxx.xxx does not match record zz of file wwwwww.www.
- Preceding records match.
- Please inform the system manager that these files are damaged.
- SORT cannot continue.

SORT is in recovery mode, and attempting to realign the indicated files so processing can continue. The files do not match. This is a serious error, and indicates that one or both files have been damaged in some fashion. There is probably little that you can do about this without the assistance of the system manager. He will need to know the file names and record numbers.

- Record yy of file xxxxxx.xxx lacks a serial number.
- Please inform the system manager.
- SORT continues.

SORT is in recovery mode, and attempting to read the indicated file for serial numbers. The specified record lacks a serial number. This is an informational message: **SORT** continues the recovery process. However, you should inform the system manager of the difficulty; he will need to know the file name and record number.

- Unexpected EOF.
- Please try again.

This probably indicates noise on the communication line. **SORT** will re-prompt for the file name.

17.8. Summary of Execution

SORT first looks to see if its state file, **STATE.SRT**, exists. If so, it assumes recovery is necessary, and positions all files to correspond to the situation just before the previous abnormal termination.

If no state file exists, **SORT** looks to see if the **APPEND** or **ROUTE** state files exist. If either of these files exists, then **SORT** cannot be run, so it terminates. All three of these programs write to the same category files, so if any of them has crashed, neither of the other two can be run until any file damage caused by the crash has been repaired. Furthermore, the three programs cannot be run simultaneously, or the category files may become hopelessly confused; this check will help prevent this conflict as well.

Next, the two authority files are copied to memory. If either is absent from the current area, the program halts. This is not considered a crash (no state file has been created), since no category files have been altered as yet.

Now, the input file is read, citation record by record. For each record in the input file, the following activity takes place:

- The citation record is copied from the input file to the computer memory.
- The EDB subject category field (540) and the INIS category number field (610) are used to deduce the proper EDB category number.
- The record history field (field 781) is updated to reflect the record movement. Normally, this field will not exist — **SORT** will create it. If it should happen to exist, the new movement information is appended to the field.
- The record is written to the proper category file.
- The location file is updated to reflect the new location of the record.
- The state file is updated, as required, to reflect program status.

When the entire input file has been distributed, the summary report is displayed and saved in file **REPORT.SRT**. Finally, the state file and the input file are deleted.

18. THSGRF — Thesaurus Display

18.1. Purpose

The EDB Thesaurus contains over 24,000 main terms, linked together to form a network. Each main term may have any of the following relationships and attributes:

- SN Scope Note. Delimits the scope of the term.
- DEF Defines the meaning of the term, as used in the thesaurus.
- DA Dates. Gives the date upon which the term was first entered into the thesaurus, and the date the entry was last modified.
- SEE Provides a link from a non-preferred term to a preferred term.
- SF Seen from. Reciprocal of SEE.
- USE Provides a link from a non-preferred term to a term that must be used instead. Terms with USE attributes cannot be used to index citations.
- UF Used for. Reciprocal of USE.
- BT Broader term. Identifies main terms of broader scope than the current main term.
- NT Narrower term. Identifies main terms of narrower scope than the current main term.
- RT Related term. Identifies main terms related to the current main term in ways other than broader and narrower.

Program **THSGRF** provides interactive searching of the thesaurus. By entering a main term through the keyboard, you have available on the display screen a picture of the main term with most of its relationships and attributes. This includes two levels of broader and narrower terms. Another display will show the remaining attributes. You may also list all main terms that begin with a specified string of characters, and all main terms that contain a specified word.

18.2. Audience

THSGRF is intended to be run by everyone.

18.3. Areas

THSGRF will operate correctly in any area.

18.4. Terminals

THSGRF will operate correctly from a variety of terminals. Four display terminals may be used: ADM 3A, ADM 31, Hazeltine 1552, and VT 100. Hard copy (line-by-line) terminals may also be used.

18.5. Files

18.5.1. Input Files

THSGRF uses the EDB thesaurus, and the inverted index to the thesaurus. No special provision need be made for these files — **THSGRF** knows where they are located.

18.5.2. Output Files

There are no output files — all output goes directly to the terminal.

18.5.3. Recovery Files

There are no recovery files — since **THSGRF** does not alter any files, no recovery procedures are necessary.

18.5.4. Authority Files

The thesaurus help file, THHELP.DAT, is used from the master area.

18.6. Dialog

The dialog required to operate **THSGRF** correctly is rather complex. We discuss it here according to the various commands.

18.6.1. Beginning the Program

1. **THSGRF** will normally be run using a mic file. The mic file establishes the proper environment and invokes the **THSGRF** program. When ready to proceed, enter *RETURN*.

User:	.do ticedt
TICEDT:	.silence
	Press RETURN to start
	*
User:	@

2. **THSGRF** must know what kind of terminal is being used, so it asks. Enter the digit that corresponds to the terminal you are using. If your terminal is not one of the stated varieties, results of the rest of the run are unpredictable. In the example shown here, a Hazeltine 1552 terminal is specified.

THSGRF:	Please specify terminal type: 1-line by line, 2-ADM 3A, 3-ADM 31 4-Hazeltine 1552, 5-DEC VT 103
User:	4

3. The program now enters a cycle consisting of (1) **THSGRF** prompting you for an command, (2) you providing the command, and (3) **THSGRF** carrying out the command. It then asks for the next command. This cycle continues until the command to quit is given.

THSGRF:	Enter next command - (main term, +nt, -nt, +rt, -rt, +bt, -bt, *stem, #word, @, &, %, ?, \$)
User:	electrons

Many of the symbols used here are shift keys on the terminal. You have the option of pressing the key unshifted — **THSGRF** will properly interpret this. That is, a '6' is interpreted as '^' on the Hazeltine 1552 or VT 103, and as a '&' on the two ADM terminals. The various options have the following intentions.

main term	Display portion of thesaurus surrounding the main term
+nt	Scroll narrower terms upwards
-nt	Scroll narrower terms downwards
+rt	Scroll related terms upwards
-rt	Scroll related terms downwards
+bt	Scroll broader terms upwards
-bt	Scroll broader terms downwards
*stem	List terms beginning with 'stem'
#word	List terms containing 'word'
@	Display descriptive information about a main term
&	Redisplay the last main term used
%	List last few main terms requested
?	List help messages
\$	Terminate the program
-	Adjust the sort size

18.6.2. Main Term Display

To produce a display of the thesaurus surrounding a main term, just enter the term when a command is requested. For example, if you enter *Nucleons*, the following is displayed. On terminals that have two levels of brightness, the main term, NT1 terms, and BT1 terms are highlighted.

BROADER TERMS		RELATED TERMS	NARROWER TERMS
		Antinucleons	Neutrons
		Brueckner Method	Cold Neutrons *
		Charge Independence	Cosmic Neutrons
		Effective Range Theory	Epithermal Neutrons
		Hard-Core Potential	Fast Neutrons
		Levinger-Bethe Theory	Fission Neutrons *
			Intermediate Neutrons
Fermions		-----	Photoneutrons
Baryons		Nucleons	Pile Neutrons
*Hadrons		DA: 120174;120174	Polyneutrons *
Baryons		-----	Resonance Neutrons
			Slow Neutrons
		Massey-Mohr Equation	Solar Neutrons
		Nucleon-Deuteron Interactions	Thermal Neutrons
		Interactions	Photonucleons
		Nucleon-Nucleon Potential	Photoneutrons
		Ope Potential	Protons
		Pseudovector Coupling	Cosmic Protons
		Rosenfeld Force	Delayed Protons

You can see that the display is basically in three columns. The main term is displayed centered on the screen, with some of its attributes. The dates are always shown here, with a few USE, UF, SEE, and SF terms (if the main term has any). Two levels of broader terms are shown to the left — BT2 terms are at the far left. Narrower terms are shown similarly to the right, with NT2 terms at the far right. Related terms are displayed above and below the center portion of the display.

The asterisk (‘*’) to the left of ‘Hadrons’ means that it has a broader term that is not displayed here; to see it, you should request the display for *Hadrons*. The asterisks to the right of ‘Cold Neutrons’, ‘Fission Neutrons’, and ‘Polyneutrons’ similarly mean that these terms have narrower terms that are not displayed here.

Here is another example, this time for *Electric Batteries*. The ‘UF’ and ‘SN’ mean that there are additional UF terms not shown here, and that there is a scope note. To see these, use the @ command. If other attributes existed for the term that are not shown here, their existence would also be noted on the lower line of dashes.

BROADER TERMS	RELATED TERMS	NARROWER TERMS
Electrochemical Cells	Battery Paste	Lead-Acid Batteries
	Battery Separators	Metal-Gas Batteries
	Cardiac Pacemakers	Aluminium-Air Batteries
	Charge State	Cadmium-Air Batteries
	-----	Iron-Air Batteries
	Electric Batteries	Lithium-Chlorine Batteries
	DA: 120174;120174	Lithium-Water-Air Batteries
	UF: Batteries (Electric)	Nickel-Hydrogen Batteries
	UF: Secondary Batteries	Silver-Hydrogen Batteries
	---- UF ----- SN --	Batteries
	Electric-Powered Vehicles	Zinc-Air Batteries
	Electrical Equipment	Zinc-Chlorine Batteries
	Electrolytic Cells	Metal-Metal Batteries
	Electromotive Force	Metal-Metal Oxide Batteries
	Energy Storage	

18.6.3. Scrolling

Scrolling allows you to examine broader, related, and narrower terms that will not fit on the display screen. For example, if you request +nt for the Electric Battery display, you will get the following.

BROADER TERMS	RELATED TERMS	NARROWER TERMS
Electrochemical Cells	Battery Paste	Metal-Metal Oxide
	Battery Separators	Batteries
	Cardiac Pacemakers	Iron-Nickel Batteries
	Charge State	Nickel-Cadmium
	-----	Batteries
	Electric Batteries	Nickel-Zinc Batteries
	DA: 120174;120174	Silver-Cadmium Batteries
	UF: Batteries	Batteries
	(Electric)	Silver-Zinc Batteries
	UF: Secondary	Zinc-Manganese
	Batteries	Batteries
	----- UF ----- SN ---	Metal-Nonmetal Batteries
	Electric-Powered Vehicles	Lithium-Copper
	Electrical Equipment	Chloride Batteries
	Electrolytic Cells	Lithium-Sulphur
	Electromotive Force	Batteries
	Energy Storage	Sodium-Sulphur
		Batteries
		Zinc-Bromine Batteries

Now, if you request +rt, this display changes to the following.

BROADER TERMS	RELATED TERMS	NARROWER TERMS
Electrochemical Cells	Energy Storage Systems	Metal-Metal Oxide
	Hybrid Electric-Powered	Batteries
	Vehicles	Iron-Nickel Batteries
	Matrix Materials	Nickel-Cadmium
	-----	Batteries
	Electric Batteries	Nickel-Zinc Batteries
	-----	Silver-Cadmium
	Off-Peak Energy Storage	Batteries
	Primary Batteries	Silver-Zinc Batteries
	Solid Electrolytes	Zinc-Manganese
		Batteries
		Metal-Nonmetal Batteries
		Lithium-Copper
		Chloride Batteries
		Lithium-Sulfur
		Batteries
		Sodium-Sulfur
		Batteries
		Zinc-Bromine Batteries

18.6.4. Secondary Display of Attributes

The various attributes frequently will not fit on the main term display (SEE, SF, USE, UF, UF+, DEF, and SN), so a separate display if available to show these. All of the attributes are

shown on this display (DA, plus those just listed), as in the following example. The attributes are requested by a @ command.

Electric Batteries	DA: 120174;120174
SN: Devices for production and/or storage of electrical energy from chemical reactions; excludes FUEL CELLS and RADIOISOTOPE BATTERIES.	
UF: Batteries (electric) Secondary Batteries Storage Batteries Voltaic Cells	

18.6.5. Terms Beginning with Specified Stem

The * command will produce a list of all main terms in the thesaurus that begin with the specified stem. If there are a great many such terms, they will be shown one screen load at a time, and you will be asked if you wish to see the next portion. Answer y or n. For example, if you request *electri, the following is displayed:

Main Terms Beginning with Specified Stem

Electric Appliances	Electric Discharge Pumping
Electric Arcs	Electric Discharges
Electric Batteries	Electric Fields
Electric Born Model	Electric Filters
Electric Bridges	Electric Furnaces
Electric Cables	Electric Fuses
Electric Charges	Electric Generators
Electric Coils	Electric Grounds
Electric Condensers	Electric Heating
Electric Conductivity	Electric Hexadecapole Transitions
Electric Conductors	Electric Impedance
Electric Contractors	Electric Logging
Electric Contacts	Electric Measuring Instruments
Electric Controllers	Electric Moments
Electric Currents	Electric Monopole Transitions
Electric Dipole Moments	Electric Monopoles
Electric Dipole Transitions	Electric Motors
Electric Dipoles	Electric Octupole Transitions

Do you wish to see the next screen load of matching terms? (Y or N)

18.6.6. Main Terms Containing Specified Word

You may list all main terms that contain a given word with the # command. The terms are listed one screen load at a time, and you will be asked if you wish to see the next portion. Answer y or n. On terminals that have two levels of brightness, the specified word is highlighted. For example, if you requested #electric, the following results.

Main Terms Containing Specified Word

Batteries (Electric)	Electric Cables
Bridges (Electric)	Electric Charges
Cables (Electric)	Electric Coils
Chugoku Electric Power Company	Electric Condensers
Reactor	Electric Conductivity
Coils (Electric)	Electric Conductors
Condensers (Electric)	Electric Contractors
Conductivity (Electric)	Electric Contacts
Conductors (Electric)	Electric Controllers
Contacts (Electric)	Electric Currents
Converters (Electric)	Electric Dipole Moments
Currents (Electric)	Electric Dipole Transitions
Discharges (Electric)	Electric Dipoles
Electric Appliances	Electric Discharge Pumping
Electric Arcs	Electric Discharges
Electric Batteries	Electric Fields
Electric Born Model	Electric Filters
Electric Bridges	Electric Furnaces

Do you wish to see the next screen load of main terms? (Y or N)

18.6.7. Recall Previous Main Terms Displayed

Occasionally, you may find it useful to examine the last few main terms that were displayed. This can be done by the % command; the last 15 (or fewer) main terms displayed are listed, as in the following. (If you have requested fewer than 15 terms, only those requested are displayed, of course.)

Last Few Main Terms Requested

Nucleons
Electric Batteries
Electrochemical Cells
Electrons
Hadrons

18.6.8. Redisplay Last Main Term

You may redisplay the last main term by the & command. This is especially useful if you are looking at the attributes display, and wish to return to the main display. You can alternate between the main and attribute displays by alternating @ and &.

18.6.9. Display Help Messages

A brief help message is available, if you enter the ? command. The message fits on the screen, and merely reminds you what the various commands mean.

18.6.10. Adjust the Sort Size

Occasionally, some of the BT, NT, or RT's associated with a main term are not in lexicographic order in the thesaurus. In order to restore proper order, THSGRF sorts small sets of terms

when they are fetched from the thesaurus. In a few cases, where there are a great many such terms, a full sort can be quite time consuming. For example, most main term displays require only a few seconds to complete. The term *Nuclei* can require up to 15 seconds (elapsed time in an otherwise unused DEC 10) for such sorting before the main display is presented on the screen. This delay can be controlled by telling **THSGRF** not to sort large sets of terms through the `^` command. The program will ask for the maximum size of sets to be sorted; any set of BT, NT, or RT's that exceed this number are left unsorted. A good value to use is *10*; then all small sets are sorted, but large sets are not. If you use this option, you may occasionally see some unsorted sets of terms.

THSGRF: Enter maximum size of BT, RT, and NT sets which may be sorted User: 10

18.6.11. Terminate THSGRF

To terminate the program, simply enter `$`. (You may also terminate with a *CTRL-C* if you wish to.)

18.7. Error Messages

All BT's have been displayed

You have entered a *+BT* command, and there are no more BT's to display.

All NT's have been displayed

You have entered a *+NT* command, and there are no more NT's to display.

All RT's have been displayed

You have entered a *+RT* command, and there are no more RT's to display.

Illegal command — correct or enter ? for help

You entered a command that cannot be recognized. **THSGRF** displays this message, waits a few seconds for the message to be read, and reprompts for input. There may have been a transmission error, the command may have been misentered, or the command may not be allowed in the current context. For example, you cannot scroll columns unless the main term display is visible on the screen.

Not a main term

You entered a term to be displayed, and it is not in the EDB thesaurus as a main term. **THSGRF** waits a few seconds for the message to be read, and prompts for new input.

Please enter YES or NO

You have been asked a question that requires a *YES* or *NO* answer. You may respond with *YES*, *Y*, *y*, *NO*, *N*, or *n*.

Storage exhausted — remaining BT's cannot be displayed

You are scrolling in the BT column, and more terms exist in the thesaurus than are allowed for in the program.

Storage exhausted — remaining NT's cannot be displayed

You are scrolling in the NT column, and more terms exist in the thesaurus than are allowed for in the program.

Storage exhausted — remaining RT's cannot be displayed

You are scrolling in the RT column, and more terms exist in the thesaurus than are allowed for in the program.

The help file could not be opened — please advise the systems programmer

The help file cannot be opened. Probably, it has been accidentally deleted from the master area — you will need the assistance of the systems programmer or the system manager to have it restored.

There are no main terms in the thesaurus that contain this word

You requested a list of all main terms that contain a specified word (the # option), and there are no such terms.

There is no term in the thesaurus that has this stem as a prefix

You requested all terms that match a specified stem (the * option), and there are no such terms.

You have reached the beginning of the BT's

You have entered a *-BT* command, and there are no more BT's to display.

You have reached the beginning of the NT's

You have entered a *-NT* command, and there are no more NT's to display.

You have reached the beginning of the RT's

You have entered a *-RT* command, and there are no more RT's to display.

19. TICEDT — Edit Citation Records

19.1. Purpose

This program is used by indexers and supervisors to edit and index citation records. It is the heart of the entire system, and is the most complex program to operate. The program is menu driven, using a hierarchy of menus. Frequently a choice in one menu will lead to a new menu (with, usually, some action in between). The exact contents of the menus is somewhat context-dependent, so some lines appear only if certain conditions are met.

19.2. Audience

TICEDT is intended to be run by indexers, and their supervisors.

19.3. Areas

TICEDT will operate in any area that is known to the system, except the master area. It cannot be run in the master area; the program will actively prevent this, since the integrity of files could not be guaranteed. An area is "known" if it is listed in file INDEX.LIS in the master area.

19.4. Terminals

TICEDT will operate correctly only from a Hazeltine 1552 or DEC VT 100 terminal.

19.5. Files

19.5.1. Input Files

The only input file is a citation file, and it may reside in the current area or in the master area. If the file is in the current area, a state file will exist that is used to reestablish your position within the citation file.

19.5.2. Output Files

Quite a number of output files may be created by TICEDT in the current area.

SAVE.DAT	will contain all citation records that you wish to send on to RECON.
CATyyB.DAT	will contain citation records that you wish to have rerouted to category 'yy'.
CATUNK.DAT	will contain citation records that you wish to have rerouted, but whose proper category is unknown.
CATREJ.DAT	will contain all citation records that you wish to have rejected from the system. Records may be rejected if they duplicate previously processed records, or if they do not apply to subject areas of interest to TIC, or for other reasons.
NEWTRM.DAT	will contain all citation records with potentially new thesaurus terms.

19.5.3. Recovery Files

There will be one recovery file in the current area for each category being worked on. The recovery file for category 'xx' is named STAx.x.DAT.

19.5.4. Authority Files

Several authority files are used by TICEDT. All are located in the master area.

CAT.LIS	provides a list of valid EDB six-digit category codes.
CODES.LIS	gives the legitimate first level category numbers.
FIELD.MEN	contains the prompts for the Primary Record Editing menu.

HELP.nnn (where 'nnn' is 'men', 'gen', 'tic', or 'sed') contain the various help messages.

INDEX.LIS gives area numbers, initials, and user type (indexer or supervisor) for all areas recognized by the system.

LOCTN.DAT is the location file.

THESAU.nnn (where 'nnn' is 'cdr', 'fdr', 'bas', 'pba', 'pdr', or 'pmx') are the pieces of the EDB thesaurus and inverted index to the thesaurus.

19.6. Dialog

The dialog required to operate **TICEDT** correctly is rather complex. We discuss it here according to the various menus used.

19.6.1. Beginning the Program

1. **TICEDT** will normally be run using a mic file. The mic file establishes the proper environment and invokes the **TICEDT** program. When ready to proceed, enter *RETURN*. If you want to see the help messages, enter an *h* before *RETURN*.

User:	do ticedt
TICEDT:	.silence
	Type a carriage return to get started. Precede is with an 'h' to get help.
User:	@

2. You will now be asked which terminal is being used. Enter 3 if you are on a Hazeltine 1552, or 4 if you are on a DEC VT 100 terminal. No other varieties are allowed. Note that no *RETURN* is required.

TICEDT:	Please enter terminal number: 3=Hazeltine 1552, 4=VT 103:
User:	4

3. There are several possible responses at this point.
 - a. If your project/programmer number cannot be found in file INDEX.LIS, the following message is displayed, and the program terminates.

TICEDT:	You are an unknown user You must be added to the file index.lis before you can use this program
---------	--

- b. If you are an indexer, you will be asked for the category number to be worked on. Enter the two digit category number. The program then proceeds to step 4.

TICEDT:	Please enter two digit category code:
User:	43

- c. If you are a supervisor, you will be asked for the initials of an indexer. Enter the three initials for the indexer whose work you wish to examine. The program then proceeds to step 5.

TICEDT:	Please enter the initials of an indexer:
User:	abc

4. You are now known to be an indexer. The program searches for a file of citations for the category designated. The file may be in the current area, if you had been working on it previously, and had not finished with it. Otherwise, the master area is searched for this file.

- a. If the file is still in the current area, and you had terminated using one of the menu options for termination, then recovery is "normal", and the following is displayed.

TICEDT:	[normal recovery]
---------	-------------------

The previous termination is considered a crash if the DEC 10 quit operating during an edit session, or if you entered a *CTRL-C*. An appropriate message is displayed; if termination occurred during an actual edit, the second line will also be seen.

TICEDT:	[crash recovery] Type 'n' to resume editing the record where you left off
---------	--

In either case, the program proceeds to the Citation File Table of Contents menu.

- b. If there is no partially completed citation file in the current area, **TICEDT** looks to see if any state file exists in the master area for **APPEND**, **ROUTE**, or **SORT**. If any such state file exists, the following message is displayed and **TICEDT** aborts. The indicated program must be run to completion before restarting **TICEDT**.

TICEDT:	State file for 'nnnnnn' program exists
---------	--

- c. If there is no partially completed citation file in the current area and no such state file in the master area, **TICEDT** searches the master area for an appropriate file. The following message is displayed, and up to twenty records are copied from the master area to the indexer area. The program proceeds to the Citation File Table of Contents menu.

TICEDT:	=== Initialization in Progress ===
---------	------------------------------------

- d. Finally, if no file for the selected category can be found in the current area or in the master area, this fact is noted on the terminal screen, and you are asked for a different category code. The program then returns to step 4, above.

TICEDT:	There are no records to edit for this category - please try another category. Please enter two digit category code:
User:	65

5. You are known to be a supervisor. The procedure is like that just described under '4' above (with the obvious substitution of 'indexer' for 'category').

19.6.2. Citation File Table of Contents Menu

After the citation file has been made available in the current area, a Table of Contents is displayed. This display gives the title of each citation record in the input file, and an indication of its disposition if it has been previously edited. The first column of the menu shows the citation number (just the relative record number on the input file). The second column shows the current disposition, and the last gives the citation title. The disposition may be changed; only the latest disposition is displayed. Possible dispositions are as follows:

@nn	Record is to be rerouted to category 'nn'. Record is to be rerouted to the 'unknown category' file.
%	Record is to be rejected.
&	Record has potentially new thesaurus terms.
!	Record is to be saved for inclusion in EDB.

Here is an example of a Title of Contents menu. The first record has been saved, the second rerouted to category 40, and the third rejected.

[1]	!	Arrangements for scattering electrons
[2]	@40	Angular divergence for electron beams
[3]	%	Models for superconducting cyclotrons
[4]		Kyoto University cyclotron field stability
[5]		Chromaticity improvement for KEK main ring
[6]		Magnetic field trimming effect
[Type a space to continue]		

After a space has been entered (by depressing the space bar on the terminal), the Record Selection menu is displayed. The Table of Contents may be stopped by typing CTRL-O, followed by the space bar.

19.6.3. Record Selection Menu

This menu permits you to select a record to be edited, to exit, or to complete editing of the input file. The menu is as follows:

d - display directory of records
 n - edit next record
 e - edit record
 r - recover record
 x - exit temporarily
 w - write records into save file and exit

Option:

((Note: the 'n' and 'w' options
 do not always appear))

Enter one of the indicated letters: **d**, **n**, **e**, **r**, **x**, or **w**. They have the following meanings:

- d** The Citation Table of Contents menu is displayed, as discussed in the preceding subsection.
- n** The next unedited record is made available for editing, as discussed below, and the Primary Record Editing menu is displayed. This option is only displayed if there are indeed unedited records in the file.
- e** **TICEDT** asks which record to edit — you may choose any of them, whether they have been edited already or not. If the selected record has been previously edited, the latest version is made available for re-editing. The Primary Record Editing menu is displayed.

TICEDT:	What is the number of the record you want to edit?
User:	5

- r** This option permits you to return to a previous version of a record. **TICEDT** asks which record is to be edited. You are then told how many versions are available, and asked which one is to be used. This version becomes the current version, and is made available for editing.

TICEDT:	What is the number of the record that you want to edit?
User:	5
TICEDT:	There are 8 versions
	Which version would you like to edit?
User:	6

- x** **TICEDT** exits. It will restart at the Citation File Table of Contents menu when invoked again for this category. This is a preferred way to terminate **TICEDT** when the citation file hasn't been completed.
- w** This entry appears in the menu only if all records in the category file have been directed to one file or another. It causes the directed dispositions to actually take place, and **TICEDT** to terminate. If there are more records from this category in the master area, this fact is noted before termination.

TICEDT:	[There are still records in this category] ((if the user is an indexer))
TICEDT:	[There are still records for this indexer] ((if the user is a supervisor))

19.6.4. Primary Record Editing Menu

After a record has been selected for editing, the Primary Record Editing menu is displayed. This gives the citation title at the top of the screen, with field options below. Certain fields have been identified as being important; the important fields that actually occur in the current citation record are identified in this menu. The following is an example.

```

Title: Arrangements for scattering electrons

Enter a field/operation

m ----- (primary title (M))
e ----- (patent assignee)
f ----- (file selected for)
c ----- (categories)
h ----- (record history)
k ----- (keywords)
a ----- (abstract)

> ----- (operations - subordinate menu)

here:

```

Most of the fields are simply available for display and editing; a few require further comment.

- c Categories. Each citation record is required to have one or more categories assigned. This field must be selected before **TICEDT** will allow the record to be written to the save file. The program will check all entries for validity; by comparing the six digit category numbers you entered against an authority list of valid numbers.
- n Suggested new term. If you believe new subject descriptors must be added to the thesaurus to properly describe the record, they are entered here. If this occurs, the record must be routed to the new term file, NEWTRM.DAT.
- h Record history. This field can be examined, but not changed. **TICEDT** will enforce this restriction. The field contains a complete history of the record's movements from file to file, giving date and time, program name, area, and file name.
- k Keywords (subject descriptors). **TICEDT** exists to help you index citations, so this option is of primary importance. Great care is taken to give you every aid possible. The process is described below, under its own heading, as it is rather complex.

The following table shows what fields will be identified in the Primary Record Editing menu, if they occur in the record.

Letter	Tag	Description
t	090	Primary title (A)
s	100	Subtitle (A)
m	110	Primary title (M)
b	120	Subtitle (M)
p	130	Primary title (S)
u	170	Affiliation (A)
l	190	Affiliation (M)
o	200	Original title (M)
e	300	Assignee
f	530	File selected for
c	540	Categories
g	620	Title augmentation
n	780	Suggested new terms
h	781	Record history
k	801	Keywords
a	950	Abstract

Once a field is selected for editing, the display screen is cleared, and the selected field is displayed. The editing commands that you may use are described below, under a separate heading.

19.6.5. Secondary Record Editing Menu

If the ">" option is selected in the Primary Record Editing menu, the secondary menu is displayed, as shown below. You will note that many of the options given in this menu are shift keys on the terminal. On the Hazeltine 1552 and VT 100 terminals, you have the option of pressing the key unshifted — TICEDT will properly interpret this. That is, '2' is interpreted as '@', '4' as '\$', etc.

Enter a field/operation	
*	----- (display entire record)
+	----- (add a new field)
—	----- (delete an existing field)
#	----- (edit a field other than the above)
@	----- (reroute the record)
.	----- (reroute to the unknown category file)
%	----- (reject the record)
&	----- (record has potentially new thesaurus terms)
!	----- (save the edited record)
?	----- (help)
\$	----- (terminate the run)
<	----- (thesaurus display)
here:	

The various options are as follows:

- * Display entire record. The entire citation record is displayed on the display screen. The screen editor is used for this display, so that scrolling is available; however, no updating is allowed. When the editor is exited, control returns to the Primary Record Editing menu.
- + Add a new field. This option allows a new field to be added to the citation record. The program will request the three digit field tag, and then use the screen editor to allow you to enter the field. The tag is checked for validity, and rejected if invalid. The screen editor is used to enter the new field; when the editor is exited, control returns to the Primary Record Editing menu.

User:	+
TICEDT:	Enter tag of field to be added:
User:	230

- Delete a field. Many of the fields may be deleted from the citation record by using this operation. **TICEDT** will request the three digit field tag, and carry out the deletion if permitted. The record history field (781), for example, cannot be deleted. Control returns to the Primary Record Editing menu.

User:	—
TICEDT:	Enter tag of field to be deleted:
User:	240

- # Edit any field. Most of the fields in the citation record may be edited by this option. It is intended to permit occasional editing of fields not mentioned in the Primary Record Editing menu, and thus will rarely be used. As with adding and deleting fields, the tag value is checked for validity. The screen editor is used to update the selected field; when this is exited, control returns to the Primary Record Editing menu. The 781 field (record history) cannot be edited with this option.

User:	#
TICEDT:	Enter tag of field to be updated
User:	250

- @ Reroute the record. If you examine the record, and decide it does not belong in this category, you have the option of rerouting it to another category file by this operation. The program will request the two digit category number of the proper category (say, 'yy'), and mark the record to be saved in file CATyyB.DAT in the current area. The actual record movement, or course, will not take place until all of the records in the file have been edited and the 'w' option is selected in the Record Selection menu. The category number is checked for validity. Control returns to the Table of Contents menu.

User:	@
TICEDT:	Please enter two digit category code:
User:	65

Reroute to Unknown Category. Reroute the citation record to the file containing records whose category number is unknown, CATUNK.DAT. Control goes to the Table of Contents

menu.

- % Reject the record. A record may be rejected either because it duplicates a record previously indexed, or because it does not pertain to a subject area of interest to TIC. All rejected records are written to the rejected records file, CATREJ.DAT in the current area. Control returns to the Table of Contents menu.
- & Potential New Thesaurus Terms. Occasionally, the 24,000+ terms in the thesaurus are not sufficient to satisfactorily index a citation — particularly in rapidly changing areas of science. If you believe a new thesaurus term is required, place the term in field 780, and direct the record to a special file, NEWTRM.DAT, by using this operation. Control returns to the Table of Contents menu.
- ! Save the Record. When all desired changes have been made to the citation record, and you believe it is ready to go on to RECON, request this operation. The record is sent to file SAVE.DAT in the current area, if editing has been completed. **TICEDT** makes some effort to be sure the record is complete, by rejecting this operation unless the category, keyword, and file-selected-for fields have been examined. (This restriction operates only on indexers; no such check is made for supervisors.) The program returns to the Table of Contents menu.
- ? Help. Help files are available to assist in general use of the indexing system, and in specific use of the screen editor. After the help messages have been displayed, control returns to the Primary Record Editing menu.
- \$ Terminate the Program. This is a preferred method of temporarily stopping **TICEDT**. When restarted, the program will begin again at the Table of Contents menu.
- < Thesaurus Display. Enter the thesaurus display program, **THSGRF**, to display portions of the thesaurus. This program is explained in its own chapter; entry from this point is to paragraph 3 of the dialog. When you exit from the thesaurus display (with a '\$' command), control returns to the Primary Record Editing menu.

19.6.6. Screen Editor Commands

A full screen editor is used by **TICEDT** to edit the various fields. The editor we are currently using works only on a Hazeltine 1552 or VT 100 terminal — hence the restriction mentioned earlier on the terminals upon which **TICEDT** can operate. It makes use of the numeric keypad on this terminal for many single-key operations; other operations involve use of control keys. Functions available are described in the remainder of this section.

Positioning the Cursor

None of the facilities described here actually change any text in the field being edited. Their sole function is to move the cursor from one position to another, in preparation for altering the text.

- Move one space in any direction. The four arrows on the numeric keypad may be used to move the cursor one space right, left, up, or down.
- Home cursor. The *home* (Hazeltine 1552) or *PF4* (VT 100) key may be used to position the cursor to the upper left corner of the screen.
- Next line. The *return* key may be used to move the cursor to the beginning of the next line.
- Move to next word. Use the *tab* key to move to the beginning of the next word. If you wish to move forward several words, just use the *tab* key several times.
- Move to preceding word. Use the *shift tab* key (ie, press *tab* while holding down the *shift* key) to move back to the beginning of the last preceding word. You may repeat this if you wish, to move several words backwards. (This option works only on the Hazeltine 1552 terminal.)
- Scrolling forward (downwards). Use the *RED* (Hazeltine 1552) or *PF2* (VT 100) key on the numeric keypad to scroll forwards through the text. This is necessary if there is more text in the field than will fit on the screen; that generally only happens with the abstract, or if you are

looking at the entire record. Scrolling is in units of eight lines.

- Scrolling backward (upwards). Use the *GREY* (Hazeltine 1552) or *PF3* (VT 100) key on the numeric keypad to scroll backwards through the text.
- Search forward for text. It is possible to search forwards for a string of characters that matches a specified string. You first press the *ENTER* key, then type in the text you wish to search for. If you now press *CTRL-R*, the cursor will be moved to the first instance of the string that follows the current cursor position. Press *CTRL-R* again to find the next instance of the string. For example, to search for the next occurrence of 'electr': *ENTER e l e c t r CTRL-R*.
- Search backward for text. You may similarly search backward for text that matches a string by using *CTRL-E* instead of *CTRL-R*. To search for the last three occurrences of 'electr': *ENTER e l e c t r CTRL-E CTRL-E CTRL-E*.
- Search forward for word. It is possible to search forward for a word that matches a specified word that is already on the screen. Position the cursor to the beginning of the word to be searched for. Then, press *ENTER*, move to the beginning of the next word by tabbing, and press *CTRL-R*. For example, to search for a word that lies under the cursor: *ENTER TAB CTRL-R*.
- Search backward for word. You may similarly search backward for text that matches a word by using *CTRL-E* instead of *CTRL-R*. In this case, the sequence of commands would be: *ENTER TAB CTRL-E*.

Changing the Text

- Replacement. To replace text, simply type over it. In this context, 'text' includes any blank space on the terminal screen. You may 'add' text in any such blank space at will; in particular, you may feel free to add to the end of a field by just typing in the information.
- Insertion. You may enter insert mode by typing the period ('.') on the numeric keypad. Now, enter the text. You exit insert mode (and return to edit mode) by again typing the period on the numeric keypad. Unlike the replacement function described just above, insertion is used to put text into the field inbetween characters that are already there. Thus, if a field ends with "and the bomb went boom." and you want to put 'atomic' between 'the' and 'bomb', use insert mode. However, if you want to add 'The city went up in smoke.' after 'boom' you may use replacement mode and just type in the text. (Of course, you can use insert mode here if you wish.)
- Insert blank line. You may insert a blank line by pressing *CTRL-D*. This is particularly useful in editing the keyword field.
- You may delete the character that currently underlies the cursor by typing 4 on the numeric keypad. To delete several characters, you may type 4 several times.
- Delete preceding character. To delete the character that precedes the cursor, type 5 on the numeric keypad.
- Delete rest of line. To delete the remainder of the current line, type 2 on the numeric keypad. The character under the cursor, as well as all following characters on that line, are deleted. The screen display is redrawn, to close up the text. To delete the entire line, position the cursor at the beginning of the line, and type 2.
- Delete a block of text. Position the cursor to the beginning of the text block, and type *ENTER*. Then, position the cursor to the end of the text that is to be deleted, and type 4 on the numeric keypad.
- Change character to upper case. Simply type 8 on the numeric keypad to change the character under the cursor to upper case.
- Change character to lower case. Type 7 on the numeric keypad to change the character under the cursor to lower case.

- Change text to upper case. Position the cursor to the beginning of the text block, and type *ENTER*. Then, position the cursor to the end of the text block that is to be changed to all upper case, and type 8 on the numeric keypad.
- Change text to lower case. Position the cursor to the beginning of the text block, and type *ENTER*. Then, position the cursor to the end of the text block that is to be changed to all lower case, and type 7 on the numeric keypad.

Moving Text Around

- Copy text. To copy a block of text from one location to another (leaving the original intact), position the cursor to the beginning of the text block, and type *ENTER*. Then, position the cursor to the end of the text block, and type *CTRL-V*. Finally, move the cursor to the location to which the text is to be inserted, and type *CTRL-G*. The text will be inserted immediately before the cursor position.
- Move text. To move a block of text from one location to another (deleting the original block), position the cursor to the beginning of the text block, and type *ENTER*. Then, position the cursor to the end of the text block, and type *CTRL-V*. Note that the cursor is returned to the beginning of the text block. Type *ENTER* again, move the cursor again to the end of the text block, and type 4 on the numeric keypad (to delete the block). Finally, move the cursor to the location to which the text is to be inserted, and type *CTRL-G*.

Exiting from the Editor

There are three methods of exiting the editor.

- Exit keeping changes. Type *CTRL-Z* to exit from the editor, keeping all changes you have made intact.
- Exit with no changes. Type *CTRL-C* to exit from the editor, leaving the original field unchanged.
- Re-edit the field. Type *CTBL-B* to discard all changes you have made to the field, and restart the editor on the field. This is useful if you have messed up the edit, and wish to start over.

19.6.7. Editing Subject Descriptors (Keywords)

The method of editing descriptors will be explained by a series of examples. Let us suppose you have requested keyword editing for a citation that discusses hydroelectric, solar and geothermal power. TICEDT responds with the screen display shown next:

**** Accepted Keywords ****

Chemical Composition
Columbia River
Emissivity
Environmental Effects
Fabrication
Geothermal Resources
Hot Springs
Hydroelectric Power Plants
Intake Structures
Measuring Methods
Nevada
Optical Properties
Performance
Pipes
Reflectivity
Spectrally Selective Surfaces
Thermal Waters
Titanium Nitrides
Zirconium Nitrides

You must now decide which keywords should be marked as M's and which as Q's. These are indicated by using the screen editor to flag the keywords on the screen with ':mx' and ':qx' (where 'x' is a digit), as shown next.

**** Accepted Keywords ****

Chemical Composition
Columbia River
Emissivity
Environmental Effects
Fabrication
Geothermal Resources
Hot Springs
Hydroelectric Power Plants
Intake Structures
Measuring Methods
Nevada
Optical Properties
Performance
Pipes
Reflectivity
Spectrally Selective Surfaces
Thermal Waters
Titanium Nitrides
Zirconium Nitrides

If you now enter *CTRL-Z*, the program will save the keywords in the subject descriptor field. If keywords are again requested, the M's and Q's will be sorted out and displayed as shown next. Note that in the previous two illustrations, the keywords were displayed in alphabetical order; now,

they are in numerical order. All unflagged keywords are shown first, any unnumbered M's next, and all numbered M's and Q's last.

**** Accepted Keywords ****

Columbia River
Environmental Effects
Fabrication
Measuring Methods
Pipes

Hydroelectric Power Plants:t1
 Intake Structures:q1
Intake Structures:m2
 Performance:q2
Titanium Nitrides:m3
 Emissivity:q3
 Reflectivity:q3
Zirconium Nitrides:t4
 Emissivity:q4
 Reflectivity:q4
Spectrally Selective Surfaces:t5
 Optical Properties:q5
Nevada:a6
 Hot Springs:q6
Hot Springs:t7
 Geothermal Resources:q7
Thermal Waters:t8
 Chemical Composition:q8

At this point, suppose you decide to use three splits — one for hydroelectric power, one for solar power, and one for geothermal power. The split numbers are entered as prefixes to each line, in the form 'n=keyword', where 'n' is the split number. If *CTRL-Z* and then 'k' are entered, the keywords look like this:

**** Accepted Keywords ****

```

1 = Columbia River
1 = Environmental Effects
1 = Pipes

1 = Hydroelectric Power Plants:t1
  1 = Intake Structures:q1
1 = Intake Structures:m2
  1 = Performance:q2

2 = Measuring Methods

2 = Titanium Nitrides:m3
  2 = Emissivity:q3
  2 = Reflectivity:q3
2 = Zirconium Nitrides:t4
  2 = Emissivity:q4
  2 = Reflectivity:q4
2 = Spectrally Selective Surfaces:t5
  Optical Properties:q5

3 = Nevada:a6
  3 = Hot Springs:q6
3 = Hot Springs:t7
  Geothermal Resources:q7
3 = Thermal Waters:t8
  Chemical Composition:q8

```

19.7. Error Messages

There are a great many "internal" errors that can cause **TICEDT** to print an error message and abort. These are not described here. If one of them should occur, please write down the entire message, and ask the system programmer or system manager for assistance.

There are numerous error checks made on keyword editing; most of these are self-explanatory, and are therefore not described further here.

A field of tag specified already exists

You cannot add a field that is already present in the citation record.

Are you sure?

You have requested a service that is of such a nature that the program wishes to be sure you mean it. Please type 'y' if the service is to be performed; 'n' otherwise.

Can't add history field

You are not allowed to make any changes to the record history field.

Can't delete history field

You are not allowed to make any changes to the record history field, including deletion.

?? can't rename temporary master file to master file.

From the master area do:

```
.rename xxxxxx.xxx=yyyyyy.yyy
```

Ticedt can be continued as after a normal exit.

Be sure that the renaming is done!!

Can't save record:

((reason))

You must examine fields 530 (file selected for), 540 (subject categories), and 801 (subject descriptors) before the citation record can be written to the save file.

Help file not available

Please ask the system manager for help. The program continues.

help message file not available

Please ask the system manager for help. The program continues.

*** illegal category xx deleted from field ***

All entries to the subject category field are checked against a list of valid categories; the one indicated here was not in that list, and so was deleted from the field in the citation record.

illegal menu code

The menu item you have specified does not correspond to one of the available options.

Illegal record number.

The number you have given does not correspond to one of the records in the file.

Illegal tag number

The number you have given is not a legitimate tag number, so the field could not be added or updated. Tags must be between 000 and 999.

Illegal version number

The version number you have requested does not correspond to one of the available versions.

invalid category code

[record not rerouted]

The number you have entered is not a legitimate first level category code, so the record could not be rerouted.

No field with tag nnn exists.

You cannot delete a field that isn't in the citation record.

No field with tag specified exists.

There is no such field in the citation record, so it cannot be edited.

Please try another category

The number you have typed does not correspond to a legitimate category number; please retype it, or specify a valid category number.

State file for 'append' program exists

The program aborts. Program **APPEND** must be run to completion before **TICEDT** can continue. Please ask the system manager for help.

State file for 'route' program exists

The program aborts. Program **ROUTE** must be run to completion before **TICEDT** can continue. Please ask the system manager for help.

State file for 'sort' program exists

The program aborts. Program **SORT** must be run to completion before **TICEDT** can continue. Please ask the system manager for help.

The scratch file nnnnnn.nnn exists, yet there is no state file
Delete it?

If you think you are in the middle of editing a file, answer 'n' and ask the system manager for help; somehow, the state file has been deleted. Otherwise, answer 'y'; the program will continue.

There are no records in this category

There are no records in the master area for the category you wish to edit.

There are no records to edit from that indexer.

There are no records in the master area from the indexer you wish to review.

[There are still records for this indexer]

There are more records in the master area for the indexer whose work is being examined.

[There are still records in this category]

There are more records in the master area for the category being edited.

There is no indexer with those initials

If there actually is such an indexer, this may be caused by a typing error, or noise on the communication line.

This program must not be run in the master area

Please use a different area

You are not allowed to run **TICEDT** in the master area, since file integrity could not be guaranteed. The program aborts; you must use a different area.

Unexpected End of File

This probably means the citation record has been damaged. Please tell the system manager.

Unexpected EOF while expecting tag

There was probably noise on the communication line; please retype the request.

Unexpected EOF — try again

There was probably noise on the communication line; please retype the request.

Unexpected EOF while expecting tag

The tag could not be read; perhaps there was noise on the communication line. Please try again to add the field.

Unknown option — try again

The option you have requested is not one of the available choices. Please choose from the menu.

You must use the 'h' command to view the history field

You are not allowed to update the history field.

20. TYPCIT — Print Citation Records on a Terminal

20.1. Purpose

This program will print citation records from any citation file on the terminal. You may skip records at the beginning of the file, limit the number of records to be listed, and select the fields to be listed.

20.2. Audience

TYPCIT may be used by anyone; however, we expect it will be used primarily by the LBL System Development Staff, and by the system manager.

20.3. Areas

TYPCIT will operate correctly from any area.

20.4. Terminals

TYPCIT will operate correctly on any terminal type, including both display screen terminals and hard copy (line-by-line) terminals.

20.5. Files

20.5.1. Input Files

The single input file is a named citation file.

20.5.2. Output Files

There are no output files — all output is directed to the terminal.

20.5.3. Recovery Files

There are no recovery files, since there is no recovery procedure. If the program does not run to completion, it must be restarted.

20.5.4. Authority Files

There are no authority files.

20.6. Dialog

1. TYPCIT is normally run using a mic file, which establishes the proper environment and invokes the program. When ready to proceed, enter *RETURN*.

User:	.do typcit
CPYCIT:	.silence
	Type RETURN to begin
	*
User:	@

2. TYPCIT clears the terminal screen and requests the name of a citation file to be printed on the terminal. Enter the file name. The file must be in the current area.

TYPCIT: User:	Citation File: cat40a.dat
------------------	------------------------------

3. If the selected file does not exist, you are informed of that fact and re-prompted for the file name.

TYPCIT:	-- Citation file xxxxxx.xxx doesn't exist
---------	---

4. You are now asked how many initial records are to be skipped. If you just type *RETURN*, no records are skipped. If you enter a number, that many records are skipped.

TYPCIT: User:	Number of Records to Skip: 4
------------------	---------------------------------

5. You are now asked how many records are to be printed. If you just type *RETURN*, all remaining records are printed. If you enter a number, no more than that many records are printed.

TYPCIT: User:	Number of Records to Print: 13
------------------	-----------------------------------

6. You are now asked what fields are to be printed. You may respond in any of the following ways:

- a. To print all fields, just type *RETURN*.

TYPCIT: User:	Tags to Print: @
------------------	---------------------

- b. To print a single field, enter its tag number.

TYPCIT: User:	Tags to Print: 10
------------------	----------------------

- c. To print a range of fields, enter the tag numbers of the first and last fields in the range, separated by a hyphen (minus sign, or dash).

TYPCIT: User:	Tags to Print: 340-400
------------------	---------------------------

- d. You may include several ranges as described in steps 6a-6c by separating them with commas.

TYPCIT:	Tags to Print:
User:	340-400,10,200-260,60

7. **TYPCIT** now lists the selected fields on the terminal screen. The details vary according to the option selected in step 6 and the terminal type.

- a. *Display Screen Terminal.* The screen is cleared and the specified fields are listed. Fields are listed in the order specified in step 6b-6d, with a blank line separating citation records. If all fields are included (step 6a), the screen is cleared before each record, and the record header is included in the display. If only one field was requested, the blank line between records is omitted. See step 8 for control commands.

TYPCIT:	HDRC AX082781 AX082781
	010 8108122260
	020 J
	060 Taniguchi, Tatsuyoshi (Kawasaki Medical School)
	230 ISSN 0386-5924
	260 Kawaski Igakkai-Shi
	340 5
	

- b. *Hard Copy Terminal.* This option is set up for the TI Omni 820 terminal. Each page has a heading and footing giving file name, page number, and date. Otherwise, the listing is just as in step 7a. See step 9 for control commands.
8. *Display Screen Control.*
- *CTRL-M, CTRL-J, RETURN, LF, Keypad DOWN.* These keys all cause the display to scroll up one line, and add a new line at the bottom of the screen. The program does not distinguish among these alternatives.
 - *q, Q, CTRL-Z.* The program terminates immediately.
 - *CTRL-L, Keypad RIGHT.* Re-display the current screen load.
 - *Keypad UP* (Hazeltine 1552 and VT 100 terminals only). Scroll down one line, adding a new line at the top of the screen.
9. *Hard Copy Terminal Control.*
- *CTRL-S.* Interrupt the listing temporarily.
 - *CTRL-Q.* Restart the interrupted listing.
 - *CTRL-L.* Reprint the current page.
 - *q, Q, CTRL-Z.* The program terminates immediately.

20.7. Error Messages

-- Illegal Character in Tag: Please Try Again

Your response to the "Tags to Print" request contains an illegal character.

-- Invalid Range for Tag: Please Try Again

You have specified an invalid tag range in response to the "Tags to Print" request.

-- Tag out of Range: Please Try Again

You have specified a tag value that is greater than 999.

There was a fatal error reading the next record

A record on the input file could not be read.

-- Too Many Tags — Please Try Again

You have requested for more than 1000 tags to be printed. This probably indicates a transmission error.

This report was done with support from the Department of Energy. Any conclusions or opinions expressed in this report represent solely those of the author(s) and not necessarily those of The Regents of the University of California, the Lawrence Berkeley Laboratory or the Department of Energy.

Reference to a company or product name does not imply approval or recommendation of the product by the University of California or the U.S. Department of Energy to the exclusion of others that may be suitable.

TECHNICAL INFORMATION DEPARTMENT
LAWRENCE BERKELEY LABORATORY
UNIVERSITY OF CALIFORNIA
BERKELEY, CALIFORNIA 94720