

DSM51



The New Retail Operation of LSI Products

Logical Systems Co.
9406 N. 107th St.
Milwaukee, WI 53224

TABLE OF CONTENTS

Getting Started with DSM51	2
Filename	3
Describe	3
Output File	4
Sub-Option #1 - Select	5
Entering Select Strings	6
Use of Connectives	7
When to Sort	8
Sub-Option #2 - Edit Select	9
Sub-Option #3 - Mark Delete	10
Sub-Option #4 - Return	10
Edit Input	10
Sort	11
Accessing Index Files	13
Using DSM51 Within a JCL	13
Warranty	

DSM51/CMD

DSM51 is a high speed, disk virtual sort utility which will index a random access disk file. DSM51 will create an index file which can be used by an application program to access the records in sorted order, based on user specified sort and select criteria. DSM51 will support logical record lengths of up to 1024 bytes, and will allow field lengths of up to 253 characters. Up to 12 different fields within each record may be used as "selection criteria" for the sort. The sort may be performed in either ascending or descending order, and multiple selection fields may be "joined" together in the sort (the length of the data sorted cannot exceed 253 characters). Optionally, a single character within each record may be used as a "deleted record" indicator. Records matching the delete criterion will not be included in the sort, and their record numbers may be written to a "deleted record" index file.

DSM51 is designed exclusively for the TRS-80 Model 1/3, running under the LDOS 5.1 Operating System. The syntax for using DSM51 is:

```
=====
| DSM51 mapfile JCL
|
| Optional Parameters:
|
| mapfile -   Represents the name of a file which contains
|              previously defined DSM51 sort and select
|              information. mapfile is a file created by
|              DSM51. If not specified, all sort and select
|              information will be prompted for.
|
| JCL      -   Indicates that DSM51 is being executed within
|              a JCL file. If used, mapfile must be
|              specified. This must be used only when DSM51
|              is being executed within a JCL file, and
|              must be specified if a JCL is used.
|
| abbr: None
|
|=====
```

Getting Started with DSM51

The DSM51 utility consists of two machine language programs - DSM51/CMD and DSORT51/CMD. Both programs are required in order to perform a sort. In order to initiate a sort, use the DSM51/CMD program.

Once DSM51 has been entered, all DSM51 primary options will appear on the bottom line of the screen.

DSM51 uses a "Line Editor" whenever information needs to be entered. A line of periods <.> will indicate that the Line Editor is active. The number of periods will signify the maximum length of the input line. As characters are typed, the periods will be replaced by the entered characters. To finish an entry while in the Line Editor, the <ENTER> key must be pressed. The <LEFT ARROW> key may be used to erase the last character typed.

Upon initial entry to DSM51, the screen will be divided into four areas. The areas will be separated from each other by horizontal lines. From top to bottom on the screen, these areas will contain information pertaining to:

- 1) Input File - The file to be sorted.
- 2) Input Fields - The starting position and length of each field in a record.
- 3) Output Fields - The fields which are to be included in the select and sort.
- 4) The options - This line will also be used to input information for the other 3 areas of the screen.

Information can be entered into each area by choosing the appropriate menu option. The options available are:

- | | |
|----------|---|
| Filename | - Enter the name of the file to be sorted. The Lrl (Logical Record Length) must also be entered. |
| Describe | - Describe the structure of each record in the file. This includes the starting position of each field and the length/type of each field. |
| Output | - Enter the sort/select criteria, according to the field specifications entered through Describe. |
| Edit | - Can be used to change incorrect input field definitions. |
| Sort | - Perform a sort using the current specifications. |

After an option is selected, the menu will be replaced by either a command line or a second option list, which will prompt you for the necessary information. During the entry of information the <BREAK> key will be active. If it is pressed, you will be returned to the "most current" option list.

If you wish to exit DSM51 and return to the DOS Level, press <BREAK> at the main option list. You will be asked to verify your decision. Press <BREAK> at this time to exit DSM51. If you do not wish to exit the program, press <ENTER>.

Filename

The Filename selection allows you to specify the file to be sorted. It must be used before any of the other options can be chosen. You will be prompted for the name of the file to be sorted. If the file is not found, an informative message will appear.

After a valid input file has been specified, you will be prompted to enter the Logical Record Length (Lrl) of each record in the file. DSM51 supports Lrl's up to and including 1024 bytes, and can sort up to 65535 logical records. In order to obtain meaningful results, the Lrl entered should correspond to the actual Logical Record Length used to create and add data to the file.

Once the filename Lrl have been entered, the top portion of the display will show the name of the file to be sorted, the Lrl and the number of records in the file. You may go back and change the sort file name or the Lrl at any time. Simply choose Filename at the main menu to do so.

Note: It is recommended that the Lrl be entered carefully prior to Describing input fields. This is because shortening the Lrl may cause "already defined" input fields to become invalid.

Describe

After the filename has been entered, the Describe option should be used to establish the fields in each record. At least one input field must be described before any of the other menu options may be selected.

In order to describe an input field, you will need to supply answers to three prompts. For each input field defined, the following prompts will appear in the order listed:

- 1) Numeric type, ENTER if ASCII IS,IU,BS,BD ..
- 2) Enter field starting byte in record
- 3) Enter field length ...

The first prompt determines the field type. If the given field contains ASCII information, just press <ENTER> in response to this prompt.

If the field which you are defining contains "compressed" numeric information, you will need to enter the two characters which describe the type of numeric information. Here are the allowable compressed numeric types:

- IS - 2 byte signed integer
- IU - 2 byte unsigned integer
- BS - 4 byte single precision value, stored in BASIC format
- BD - 8 byte double precision value, stored in BASIC format

After the field type has been entered, a second prompt will appear for the starting location of the field within the record. The appropriate value should be entered in response to this prompt, keeping in mind that relative byte positions are offset from 1 (e.g. if the field begins at the very first byte in the record, the starting byte should be entered as 1).

The last prompt will appear only if the field is an ASCII field. At this point, the length of the field (in bytes) should be entered. If the field type is a compressed numeric type, the field length will default to the appropriate value (as shown in the table).

After all entries have been made for the definition of a field, additional fields may be defined. Upon the completion of entering input fields, you may press <BREAK> in response to any of the prompts to return to the menu.

Field definition follows a free format, in that the fields do not have to be defined in the order of appearance in the record (e.g. field 1 can be defined as starting at position 10 in the record, while field 2 can start at position 1). It is also allowable to "overlap" field definitions.

Futhermore, you need only define those fields in the record that you will be sorting/selecting on. As an example, suppose that you wish to perform a sort based on the information contained within the first 20 bytes of each record in the file, where the Lrl of each record is 90 bytes. In this case, you would only need to define 1 field, whose starting position would be byte 1, with a length of 20 bytes.

The only restriction in defining fields is that no field can extend beyond the end of the record, based on the Lrl of the file. Entries which would cause such an occurrence will be ignored, and you will be re-prompted to enter the information.

If the Lrl of the file is changed and becomes smaller, none of the field definitions will be changed (thus it is possible to have field definitions beyond the Lrl). However, you will not be allowed to sort until the field definitions are changed in accordance to the new Lrl.

Output File

The Output File option allows you to define the specifications for sorting/selecting records, based upon the fields entered in Describe. After entering the Output mode, a sub-menu will appear, with the following options available:

- 1) Select - Identify the fields to be involved in the sort/select.
- 2) Edit Select - Modify the current sort/select criteria.
- 3) Mark Delete - Establish "deleted" record criterion.
- 4) Return - Return to the main menu.

Sub-Option #1 - Select

The Select option will allow you to specify the criteria to use in the sorting process. A series of prompts will appear for each select position:

- 1) Enter input field number ..
- 2) Add field to the sort string (Y/N)? .
- 3) Enter select string
- 4) Relation EQ,GE,LE,GT,LT,NE <ENTER> means EQ ..
- 5) Enter connective (AND, OR) or ENTER if done ...

The first prompt allows you to identify the field number to use for the selection. The field number must correspond to a previously defined field (see Describe). Simply enter the field number corresponding to that displayed in the Input Fields section of the screen.

The second prompt indicates whether or not the information in this field will be added to the sorted information. If you wish to specify selection criteria only for this field, and do not care if the information is sorted according to this field, answer this prompt by entering <N>. If you want the information in this field to participate in the sort, answer this prompt with <Y>.

Prompts 3 and 4 work together in determining the records that will be selected and included in the sort. In general, information will be entered to establish limits for evaluating a record. If the data in the record falls within the specified limits, it will be included in the sort and its record number will be written to the index file. The third prompt will establish the criteria to be used, while the fourth prompt will be used to determine whether or not the information is within the specified limits.

As an example, assume that we have a field defined which contains ASCII zip codes. We wish to create an index file containing the record numbers of all records with the zip code 53223. In order to do this with DSM51, we would specify a "Select String" of 53223 (prompt number 3) and a "Relation" of EQ (prompt number 4). The select string and relation indicate that we only want records having a "zip code equal to 53223".

In this example, the <ENTER> key could have been pressed in response to the "Relation" prompt (without typing EQ), since "Equal to" is the default. The other relations are:

- GE - Greater Than or Equal to.
- LE - Less Than or Equal to.
- GT - Greater Than.
- LT - Less Than.
- NE - Not Equal To.

The last prompt for the select option (Prompt number 5) will allow a "Connective" to be specified. The valid connectives are <AND> and <OR>, and are applied to select criteria in "left to right" order.

Let us expand on our previous example and include records whose zip codes fall in the range of 53200 to 53999, inclusive. The following table will illustrate how prompts 3-5 should be answered to achieve the desired results:

Select Criteria (Prompt #3)	Relation (Prompt #4)	Connective (Prompt #5)
53200	GE	AND
53999	LE	

As the select criteria is being entered, the information will appear in the Output Fields area of the screen. Up to 12 different selection criteria may be specified. Due to the limitations of the display, only 6 select criteria are displayed at any one time. If more than 6 are entered, that portion of the screen will be cleared after the entry of the 6th item, and the 7th selection criteria will replace the 1st one as it is entered.

When there are more than 6 select criteria, the down arrow key may be used at the menus to advance the display of the output fields to the next "page". To end the selection process, press <ENTER> in response to the connective prompt, and you will be returned to the output file sub-menu. You will notice that there is no requirement for a connective on the last select criteria entered. If you leave select (by pressing <ENTER> for the connective) and wish to enter additional selections at a later time, you will be prompted for a connective for the last selection entered. After the connective is entered, selection will continue as before.

Entering Select Strings

A few points need to be made concerning the entry of select strings. The length of the "select string" prompt will normally be the length of the field definition. However, if the field is defined to be larger than 30 characters, you will be limited to the entry of 30 characters for the select string.

The entry of numerics will be limited to 6 characters for integers, nine for single precision, and 17 characters for double precision. With respect to the entering of "fractional" values (i.e. single and double values with digits to the right of the decimal point), it is best to allow a "ranging" of values when looking for an exact match. This is due to the fact that rounding errors may occur when the input value is changed to compressed format. For example, if you wish to select all records that have a value of 16.7 in a single precision field, rather than expressing this as "Equal to 16.7" (where it is possible that a "match" might not be found), if the expression "Greater Than 16.699 AND Less Than 16.701" is used, you are assured of always finding the desired match.

When specifying select strings, the question mark <?> may be used as a wildcard character, and will cause a match on any character. Use of the question mark will be very important when specifying a relationship of "equal to". The following example shows how the question mark affects select strings and matches:

Let us assume that we have an ASCII data file which contains two fields: Field 1 is a last name field (15 characters long) and field 2 is a first name field (10 characters long). Let us further suppose that we wish to create an index file of all records having the last name "Johnson".

One way of doing this would be to set up our criteria as "Equal to Johnson". This method however, would NOT find any last names entered with a trailing comma (since "Johnson," is Greater than "Johnson"). To make sure that all Johnson's are found (regardless of the manner in which they are stored on disk), the following select string could be used with a relationship of equal to:

Johnson????????

Using this select string, matches will be based on the first 7 characters of the field being equal to "Johnson", while whatever is in the remaining 8 characters will automatically match (due to the wildcard characters).

I M P O R T A N T N O T E

In most cases where an exact match is desired, any trailing characters in the select string should be filled with a question mark to guarantee that the desired matches are found. One important reason for this is that the select string will NOT be padded with trailing right spaces if it does not fill the field length (unlike the BASIC command LSET, which will pad the right part of the

string with spaces). Thus, if the select field length is 15 characters and the select string is "Johnson" (with a relation of equal to), any string of "Johnson" on disk which is followed by 8 spaces will NOT match (it will in essence be Greater Than the specified select string, since the comparison is performed on all 15 characters).

Another important point to make concerns the use of upper and lower case in specifying a select string. In order to establish a consistent selection method, all alphabetic characters contained in a select string (i.e. the characters A-Z and a-z) will be internally converted to lower case. Furthermore, any alphabetic characters read from the data file will be converted to lower case prior to doing the selection comparisons and sorting operation (please note that the data file will be left untouched). This method will assure that selections can always be made in a meaningful and logical manner. For the most part, these conversions will not corrupt the results obtained from the sort. The only time a true ASCII sort will not be obtained is when upper case alphabetic characters are sorted along with special characters within the decimal range of 91-96. In this case, the alphabetic characters will be seen as "greater than" the special characters.

Another case in which using wildcard characters will prove beneficial is in selecting all records which have a field that begins with the same character. For example, suppose that we would like to select all records having a last name that begins with J. If the field is defined to be 15 characters in length, we could perform this type of selection by specifying a select string of J????????????? (the character J with 14 question marks) and a relation of equal to.

One final but very important note on the entering of select strings is the use of a default select string. If the select string prompt is answered by just pressing <ENTER> (i.e. no characters are entered for the select string), a default string of question marks (having a length of the defined field length) will be assigned to the select string. This represents an "everything matches" select string, and is useful in performing a sort on all records. When specifying an "everything matches" select string, the relation must be Equal To.

Use of Connectives

For the most part, using connectives in determining the selection of records will be a straight forward AND or OR type situation (e.g. greater than 53000 and less than 53999). However, when multiple fields (more than 2) are used to perform the selections, the train of logic can become quite complex, since the evaluation of criteria is done in field number order.

For example, fields 1 and 2 are each evaluated (as either True or False), with the connective applied to obtain an intermediate result (e.g. if the connective is AND, both fields 1 and 2 must be true for the intermediate result to be true). Subsequent fields are then evaluated and are used with the previous intermediate result to obtain a new intermediate result. These evaluations continue in such a manner until the last selection has been evaluated, at which time the final result is obtained.

Suppose that we have a file containing names and addresses. We are interested in finding people that have a last name of either Jones or Smith who live in the zip code range of 53220 through 53250, inclusive. Let us also assume that the last name field is defined as input field 1, with a length of 5 characters and that the zip code field is defined as input field 2, with a length of 5 characters. The table below will illustrate the proper selection entries and connective usage to obtain the desired results:

Select Field	Select Criteria	Relation	Connective
1	Jones	EQ	OR
1	Smith	EQ	AND
2	53220	GE	AND
2	53250	LE	

Following through the logic, we can see that the first two lines will evaluate true only if the name is Smith or Jones. If the name is not Smith or Jones, the first evaluation is false. Seeing that the remaining connectives are all AND, if the first evaluation is false, then the end result will also be false. If the first evaluation is true, then all of the remaining evaluations must also be true in order for the record to be selected (i.e. the zip code MUST be in the proper range, as well as the name being one of Smith or Jones).

Compare the table above (which shows the proper method of obtaining the desired results) to the one shown below, taking note of the subtle differences in the train of logic.

Select Field	Select Criteria	Relation	Connective
2	53220	GE	AND
2	53250	LE	AND
1	Jones	EQ	OR
1	Smith	EQ	

In this particular example, the results will be correct for all of the people with the last name of Jones. However, due to the manner in which the logic has been established, ALL Smith's (regardless of the zip code) will be evaluated as matching, which is not the desired result.

When to Sort

The main function of DSM51 is its ability to sort large amounts of information in a relatively short period of time. There is a differentiation to be made however, between the terms "selecting" and "sorting". To select a record implies that the record meets some specified criteria and will be included in the sort operation.

The sort operation creates an index file. This index file contains all selected record numbers, organized (in ascending or descending order) according to the information contained in one or more fields. The point to be made is that a select field need NOT necessarily be sorted, and (for the most part) vice versa.

As an example, let us revert back to our data file which contains names and addresses. We have defined input field 1 (zip code) to be 5 characters long, and input field 2 (last name) to be 10 characters long. We wish to sort by last name all records within the zip code range 53000 to 53999, inclusive. The chart below will show one method for setting up select criteria to accomplish this:

Select Field	Sort Flag	Select Criteria	Relation	Connective
1		53000	GE	AND
1		53999	LE	AND
2	S	??????????	EQ	

Notice that the sort is being performed on field 2, but the actual selection is being done on field 1 (since field 2 is defined as "match anything"). Aside from dictating the records to be sorted, field 1 will have no bearing in the results of the sort.

Let us modify this example to illustrate the use of more than one field in a sort operation. Assume that we wish to select the same records (those within the zip code range 53000 to 53999) and sort these records primarily according to last name. In addition, if two of the selected records have the same last name, we would like to sort these secondarily in zip code order. The following table will illustrate the required select/sort criteria:

Select Field	Sort Flag	Select Criteria	Relation	Connective
2	S	??????????	EQ	AND
1	S	53000	GE	AND
1		53999	LE	

There are two important points to be drawn from this example with respect to specifying sort fields. First, the physical order in which sort fields are entered in the select table differentiates the primary sort field from the secondary sort field. In this case, where the sort is to be done primarily by last name (field #2) and secondarily by zip code (field #1), field 2 MUST be the first entry in the select table. If the ordering of the entries was switched, the sort would have been performed primarily by zip code and secondarily by last name. When it comes to entering sorted fields into the select table, the first sorted entry in the table will be the main sort field, with secondary fields following, in order of appearance in the table.

The second point deals with the sort flag specification for fields that appear more than once in the select table. In the above example, field 1 is entered twice in the select table, but the sort flag is only specified for the first entry. When select criteria is entered, you will always be prompted if you wish to add the select field to the sort string, regardless of whether or not the field is already sorted. If the same field is specified as a sorted field more than once, it will appear in the sort string as many times as specified. This will not affect the results of the sort, but it will waste both time and the memory allocated for each sort string (remember, the maximum length for all sorted fields cannot exceed 253 characters).

Sub-Option #2 - Edit Select

The edit select option at the Output Fields sub-menu will allow you to edit, insert or delete entries from the select table. After choosing this option, the following prompt will appear:

Enter output field number ..

Answer this prompt by entering the number corresponding to the item in the select table that you wish to deal with. Once the select position number has been entered, this prompt will appear:

<C>hange, <R>emove, or <I>nsert field ? .

Answer this prompt by entering one of either <C>, <R> or <I>.

<C> will allow you to change the information for the specified selection entry. Information will be changed in the same manner in which it was initially entered (via select), with all of the prompts appearing one at a time.

<R> will allow you to remove the specified selection entry from the table. The given select criteria will be removed from the table and all trailing entries will be shuffled up accordingly.

<I> will allow you to insert select criteria at the position specified. Insertions into the select table will be made prior to the position specified. Thus, if you have 5 entries in the select table and you specify an insert at position 1, prompts will appear for the

item to be inserted (in the same manner as select). After all of the prompts have been answered, the newly entered item will appear at position 1 in the select table and all of the existing selections will be shuffled down one position.

Sub-Option #3 - Mark Delete

The mark delete option will allow you to specify one byte in each record which will denote deleted records. It can be used only with data files which contain an internal deleted record indicator within each record. Any records which match the delete criterion will not be included in the sort. In addition, if a deleted record mark is specified, you will have an option to write the record numbers of deleted records to a separate delete file (see Sort for more information). The following prompts will appear, one at a time:

- 1) Value in decimal that marks a delete (0-255) ...
- 2) Relation of mark EQ,GE,LE,GT,LT,NE ..
- 3) Position of mark in record

The first prompt requests the character (delete mark) which will be used to compare for a deleted record. It must be entered as a decimal value in the range of 0-255.

The second prompt is for the relation of the delete mark to the character in the file to determine whether the record is deleted. The relations listed are the same as those entered in select.

The last prompt is for the relative byte of the record (offset from 1) which contains the delete mark.

As an example, suppose we have a data file in which deleted records are denoted internally, by the first byte in the record being a 0. In order to mark deleted records with DSM51 in this case, the first prompt would be answered with 0, the second prompt with EQ and the third prompt with 1. Anytime a zero byte is found in the first byte of a record, that record will be seen as deleted, and will not be included in the sort.

Sub-Option #4 - Return

The return option will return you to the main menu from the output fields sub-menu. Pressing <BREAK> at the output fields sub-menu will also return you to the main menu.

This concludes our discussion of the Output Fields option at the main menu.

Edit Input

Selecting Edit Input at the main menu will allow you to edit or remove the field definitions shown in the Input Fields section of the screen. After choosing this option, the following prompts will appear, one at a time:

- 1) Enter input field # to change ..
- 2) Numeric type, ENTER if ASCII IS,IU,BS,BD ..
- 3) Enter new starting byte, 0 to delete field
- 4) Enter field length ...

The first prompt is for the input field number that you wish to edit or remove. Enter the number which corresponds to the given field, as shown in Input Fields section of the screen.

The second prompt is for the field type, and should be answered accordingly (see Describe for more information).

The third prompt is for the field starting byte in the record and is similar to prompt number 2 in Describe. Answer this prompt by entering the starting position of the field

within the record. If you wish to remove the field from the Input Fields table, enter 0 in response to this prompt.

The last prompt is for the length of the field and again is similar to the length prompt (prompt 3) in Describe. If the field is a compressed numeric type, you will not be prompted for the length.

After an input field is edited, the information on the screen corresponding to the input field will be updated accordingly.

If an input field is removed, all trailing input fields will be shuffled up one position and the screen display will be updated accordingly. For example, if 5 input fields have been defined and input field number 3 is removed, the result will be four input fields, numbered 1-4 (i.e. the old input field number 4 will become the new input field number 3, and the old field number 5 will become the new field number 4).

Some special considerations need to be made when deleting or editing an input field after select criteria have been specified. If an input field is deleted, all necessary adjustments will automatically be made in the select table. Any entry in the select table of the deleted field will be removed, and all field numbers will be changed accordingly (due to the shuffling of input fields).

When editing an input field, changing the starting position of the field in the record will cause no problems. However, changing the field length may cause discrepancies with previously defined select strings for that field. If the field length becomes shorter, the length of the select string will be shortened automatically to a maximum of the new field length.

If the field length becomes larger, the select string will be left as is. In this case, you may need to edit the select strings of any output fields which refer to the edited input field number. In particular, if the old select string contains trailing wildcard characters, it may be necessary for you to re-enter the select string to incorporate additional trailing question marks.

This concludes our discussion of the Edit Input option at the main menu.

Sort

The Sort option at the menu will allow you to perform a sort on the specified input file, using the parameters that have been entered in Describe and Output. In order to perform a sort, at least one of the select fields must have the sort option specified. Also at this time, the sort parameters are checked to make sure that they are all valid. If no sort field has been specified, or an invalid parameter has been detected, an informative message will be displayed and you will need to correct the problem before you will be allowed to sort.

Prior to the sort being performed, the following prompts will appear, one at a time.

- 1) Save this as a Map file (Y/N) ? .
- 2) Enter output file to create
- 3) Sort direction <A>scend or <D>escend: .
- 4) Drive for DELETE file, ENTER if none .
- 5) Enter drive for work file .

The first prompt is asking whether or not you wish to create a map file. A map file is a separate disk file which contains all currently defined sort parameters. It is useful to create a map file if you wish to control DSM51 in a JCL (see "Using DSM51 within a JCL" for more information), or if you will be performing the same sort at a later time. If you do not wish to create a map file, enter N in response to the prompt.

If you want a map file to be created, answer the prompt with <Y>. An additional prompt will appear, asking you for the map file name. At this time, enter the name that you wish to assign to the map file, and all of the current sort parameters will be written to this file.

After a map file has been created, it is a simple matter to perform subsequent sorts using the same parameters. Assume that you have created a map file name MYSORT/MAP. At a later time, if you wish to sort the same file with the same sort parameters, you could use the following command to execute DSM51 (note that the command must be entered exactly as shown):

DSM51 MYSORT/MAP

Upon coming into DSM51, all sort parameters will be established, with the screen display showing the current settings. At this time simply enter the Sort command at the menu to perform the sort. Of course, any of the parameters in a map file can be edited in the normal manner with DSM51 prior to sorting.

The second prompt is for the name of the index file which will be created. Enter the name you wish to assign to the index file. The third prompt is for the direction in which the sort is to be performed. Answer this prompt with <A> if you want the records to be sorted in Ascending (least to greatest) order. To sort in Descending (greatest to least) order, enter <D> in response to this prompt.

The fourth prompt will appear only if a deleted record mark has been specified. If you wish to write the record numbers of deleted records out to a separate disk file, answer this prompt with the drive number that you want the deleted record file to be written to. If you do not wish to write a deleted record file, press only <ENTER> in response to this prompt.

When creating a deleted record file, the filename of the deleted record file will be the same as the filename assigned to the index file, and an extension of "/DEL" will be assigned to it.

The last prompt deals with the placement of a temporary work file. Depending on the length of the sort string and the number of records to be sorted, a temporary work file may be needed to perform the sort. Answer this prompt by entering the drive on which you want the work file to be built on.

The same rules which apply to the naming of the deleted record file also apply to the work file, except that the file extension will be "/TMP".

To approximate the amount of space needed by the work file, add 4 to the length of the sort string, and multiply this result by the number of records to be sorted. If the value you get is greater than 33,000 (and no LDOS drivers/filters are in high memory), a work file may be needed and will take up approximately that much disk space. No work file will be needed if the value is less than 33,000.

After all prompts have been answered, the sorting process will begin. At this point, the program DSORT51/CMD will be loaded from disk and will perform the actual sort. The screen will be cleared and a signon message will appear.

While the sort is taking place, informative messages will appear showing the progress of the sort. The first message to appear will be "Loading" and will show the record numbers as they are read into memory (in increments of 100). As many records as allowed by available memory will be read into memory. After the last record has been read, or if all memory has been used, the records will be sorted. After the records have been sorted, information will be written to the work file (if it is needed), and the cycle of loading and sorting will continue for the next group of records.

After all records have been sorted, the index file will be created. If more than one pass was needed, the number of record numbers to write to the index file will count down

from the total number of records sorted to 0. Once finished, a message will appear showing the total number of records sorted and the operation will be complete.

Accessing Index Files

The index files which are created by DSM51 contain compressed integer representations of all record numbers which were selected. These record numbers are stored in the file according to the results of the sort. If the sort was performed in ascending order, the first record number in the index file will be the record with the smallest sorted data.

It is a simple matter to utilize the index file when accessing your records in sorted order. The following BASIC program statements will illustrate a method by which all records in the index file may be accessed in sorted order.

```
10 OPEN"R",1,"MYINDX/IND",2  'Open the index file (Lr1 of 2)
20 FIELD 1, 2 AS S$          'Field the index file
30 OPEN"R",2,"MYDATA/DAT",1r1 'Open data file with proper lrl
40 FIELD 2, xxxxxxxxx        'Establish fields in the data file
50 FOR L%=1 to LOF(1)         'For as many records as are indexed
60 GET 1,L%                   'Get the record number from the index file
70 GET 2,CVI(S$)              'Get that record from the data file
80 ' Do whatever processing is required (e.g. print the records)
90 NEXT L%                    'Continue with the next one
100 CLOSE                     'Close the files when done
110 END
```

In this example, all records that were sorted will be accessed in sorted order (either ascending or descending, depending upon the direction of the sort). Although very simplistic, this example shows the one method to use in accessing an index file.

There are various other methods which can be used to "search" through an index file to find a given piece of information in the data file. Although these methods will not be discussed here, one very reliable and quick method is to perform a "binary" search of the data file using the index file.

I M P O R T A N T N O T I C E

Due to the nature of alphabetic ASCII data (i.e. the characters A-Z and a-z), an internal conversion to lower case is performed on all alphabetic characters during the selection process, prior to performing the sort (please note that the data file is left unchanged). In some cases (to assure proper results when using a "searching" routine), it is recommended that alphabetic characters in the "search string" be converted to lower case, and the same type of conversion be performed after the data is read in from disk, prior to performing any comparisons.

As a final note, accessing a deleted record file will be done in much the same manner, since the record numbers are stored as compressed integers.

Using DSM51 Within a JCL

DSM51 may be controlled entirely by a JCL file. In order to do so, you must utilize a map file which contains all of the desired sort parameters, and include in the JCL file answers to all of the prompts for the Sort option. The JCL parameter must also be specified in the DSM51 command line (it will appear after the map file on the command line).

The following is a sample JCL file which will totally automate a DSM51 sort. The assumptions are: the map file is named SORTED/MAP, the index file to be created will be named

SORTED/IND, the sort will be in ascending order, the deleted record file will be stored on drive 4 and the work file will be placed on drive 3.

DSM51 SORTED/MAP (JCL)

READ ME
SORTED/IND

A
4
3

If the above JCL file is named SORT/JCL, the following command can be entered to automate the entire sort process:

DO SORT

The first two lines of the JCL file will always be as shown (with the exception of the map file name). The first line is the DSM51 command line. The second line initiates the sort option at the DSM51 menu.

The third line of the JCL file is the index file to be created. The fourth line dictates the sort direction.

The fifth line represents the drive on which the delete file is to be placed. If a deleted mark is specified, a drive for the delete file must be specified. If no deleted mark was specified, the fifth line in the JCL file must be omitted (i.e. the JCL file will contain five lines instead of six).

The last line of the JCL file will always be the drive on which to place the work file.

I M P O R T A N T N O T I C E

This package is sold on an "as-is" basis. Logical Systems makes no expressed or implied warranty of any kind with regard to the software or documentation. Under no circumstances will Logical Systems, Inc. assume any liability for actual, incidental or consequential damages resulting from the use of this package.

From time to time, updates to this product may become available for a nominal charge. Customer Service information on this product and any available updates may be acquired by contacting Logical Systems at the regular customer service address.

Logical Systems
PO Box 23956
Milwaukee, WI 53223

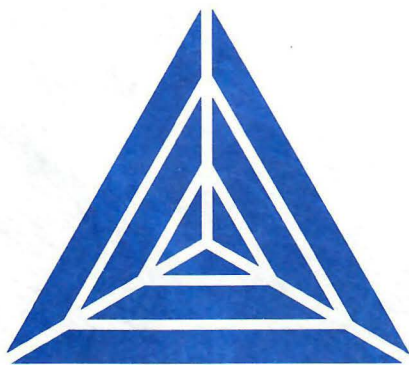
414-355-5454

The entire DSM51 package and manual is Copyrighted 1985 by Logical Systems. It is unlawful to reproduce any part of this product, by any means, for commercial purposes or any other form of distribution, without the expressed written consent of, or a licensing agreement from Logical Systems.

DSM51

©1985 by Logical Systems Co.

Serial # DSM51-20044



MISCO Systems, Inc.