SYSTEM/32

IBM System/32
Utilities Program Product
Reference Manual
Sort

IBM System/32
Sort
Programming Information

SC21-7633 2
File No. S32 33

Program Number
5725-UT1

IBM System/32
Utilities Program Product
Reference Manual
Sort

Third Edition (November 1977)

This is a major revision of, and obsoletes SC21-7633-1. Changes are indicated by a vertical line at the left of the change.

This edition applies to version 7 of Program Product 5725-UT1 and to all subsequent versions and modifications until otherwise indicated in new editions or technical newsletters.

Requests for copies of IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A Reader's Comment Form is at the back of this publication. If the form is gone, address your comments to IBM Corporation, Publications, Department 245, Rochester, Minnesota 55901.

Preface

Purpose of This Manual

The Sort Reference Manual is written to familiarize readers with the types of sort that can be run on the System/32, and to serve as a reference for the programmer who must define and run sort programs. Use this manual to:

- Identify and select the type of sort needed
- Complete sequence specifications sheets
- Supply OCL and sequence specifications when using your own procedures
- Optimize sort run time

Audience

The primary audience for the Sort Reference Manual is the account programmer. The secondary audience includes third party programmers and IBM field support personnel (SE, PSR, PSCE, and instructor).

How This Manual is Organized

This publication contains the following six levels of information.

1. *Sequence Specifications Overview* shows which columns on the sequence specifications sheet you must consider when you want to sort a file.
2. *Column Summaries* list the possible entries for each column on the sequence specification sheet.
3. *Column Descriptions* explain the possible entries for each column in greater detail.
4. *How to Run Sort* shows how to enter OCL statements to load and execute a sort job.
5. *Performance Considerations* explains the various factors that affect the time required to run a sort job.
6. *Sample Jobs* show how to fill out the sequence specifications sheets for typical jobs.

Here is how to use each level:

- Use the *Overview* in Chapter 1 to see which columns on the sequence specifications sheet you must consider when you want to sort a file.
- Use the *Column Summaries* at the beginning of Chapters 2 through 4 for a quick recall of the possible entries for each column on the sequence specifications sheet.
- Use the *Column Descriptions* in Chapters 2 through 4 when you need a detailed explanation of the entries for a particular column.
- Use *How to Run Sort* in Chapter 5 to set up your files and specify what you want done with the information in the files.
- Use the *Performance Considerations* in Chapter 6 for the most efficient method to run your sort job.
- Use the *Sample Jobs* in Chapter 7 to review coding techniques for actual jobs.

Column summaries and column descriptions are provided for each type of specification:

- Header
- Record type
- Field description

System Requirements

The Sort portion of the IBM System/32 Utilities Program Product runs on all models of the System/32 and supports all available features.

Related System/32 Publications

- *IBM System/32 Introduction*, GC21-7852
- *IBM System/32 System Control Programming Reference Manual*, GC21-7593
- *IBM System/32 Utilities Program Product Reference Manual—Data File Utility*, SC21-7600
- *IBM System/32 Displayed Messages Guide*, GC21-7704

Titles and abstracts of other related publications are listed in the *IBM System/32 Bibliography*, GC20-0032.

Contents

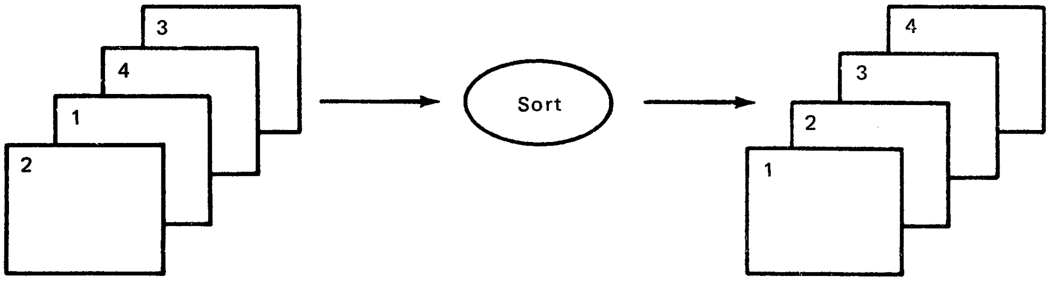
Chapter 1. Overview of Sort

The sort program is part of the System/32 Utilities Program Product. You can use the sort program to rearrange, drop, and reformat records in your file.

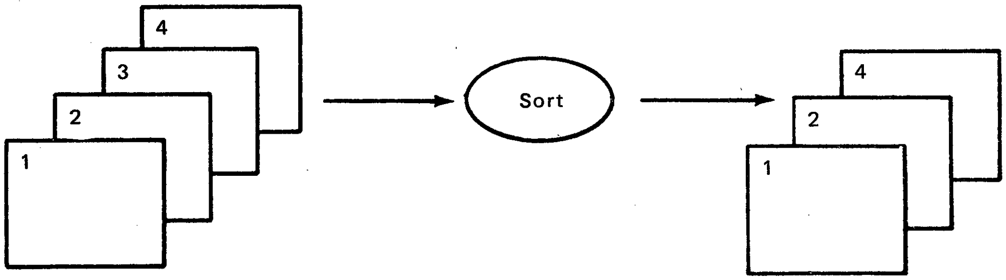
SORT FUNCTIONS

The three basic functions of sort are:

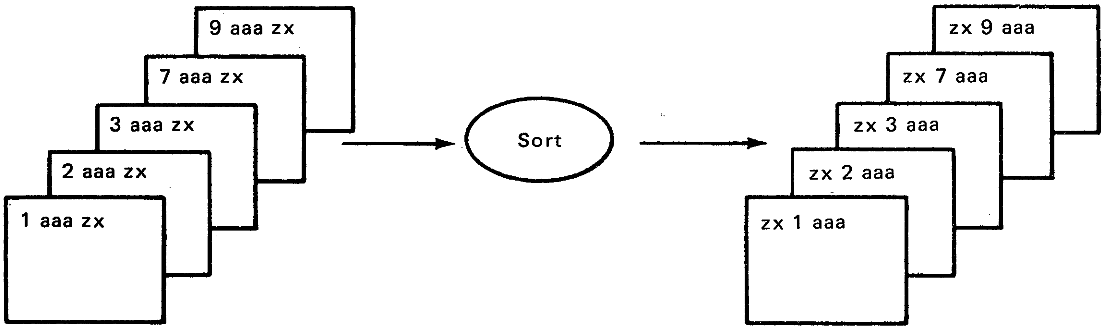
- To rearrange the records in a file.



- To drop records from a file.



- To reformat the records in a file.

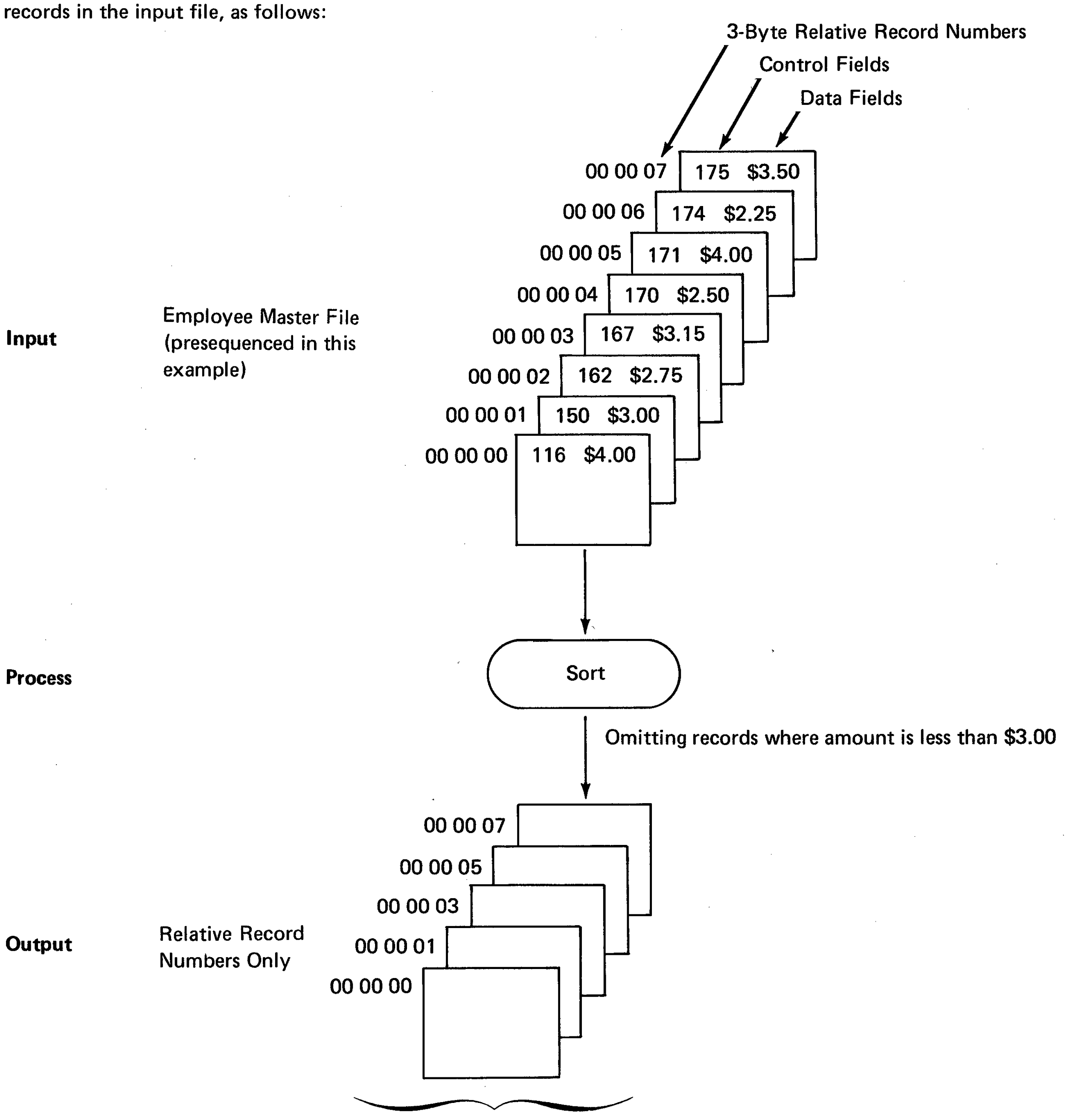


TYPES OF SORT: ADDROUT, TAGALONG, AND SUMMARY TAGALONG

The three types of sort jobs are: *addrout* (address out); *tagalong* (specified data fields can be included with control fields in the sorted records); and *summary tagalong* (a total or totals are accumulated in the sorted records).

Addrout Sort

The output from an addrout sort job consists of 3-byte hexadecimal relative record numbers of some or all of the records in the input file, as follows:



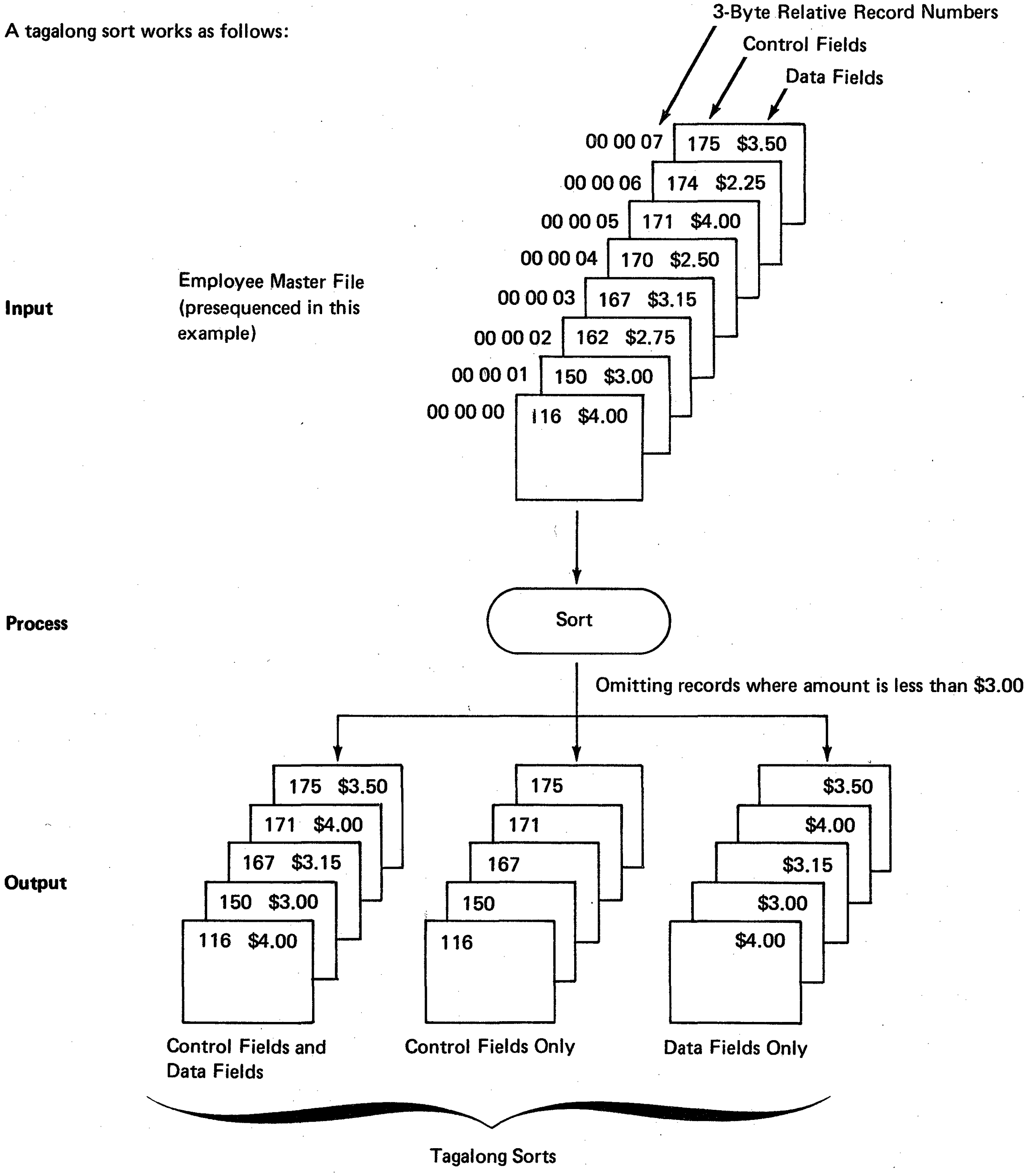
Addrout Sort

Tagalong Sort

The output from a tagalong sort is a file of sorted records which can contain:

- Control fields only
- Data fields only
- Control fields and data fields

A tagalong sort works as follows:



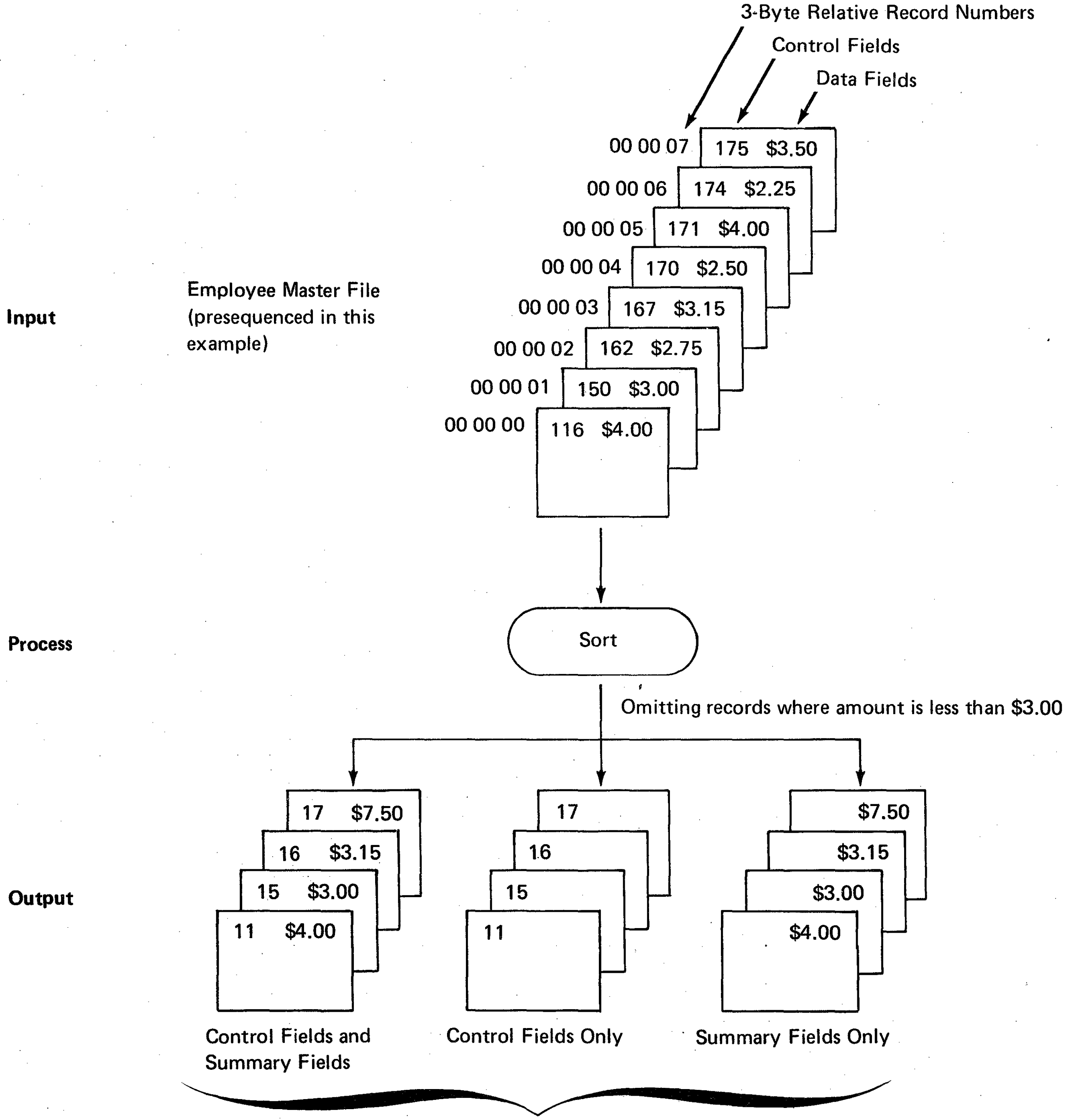
Summary Tagalong Sort

The output from a summary tagalong sort is a file of sorted summary records that usually contains:

- Control fields only
- Summary fields only
- Control fields and summary fields

Summary sort output can also include data fields. However, you cannot determine which input record's data field will appear in the summary record.

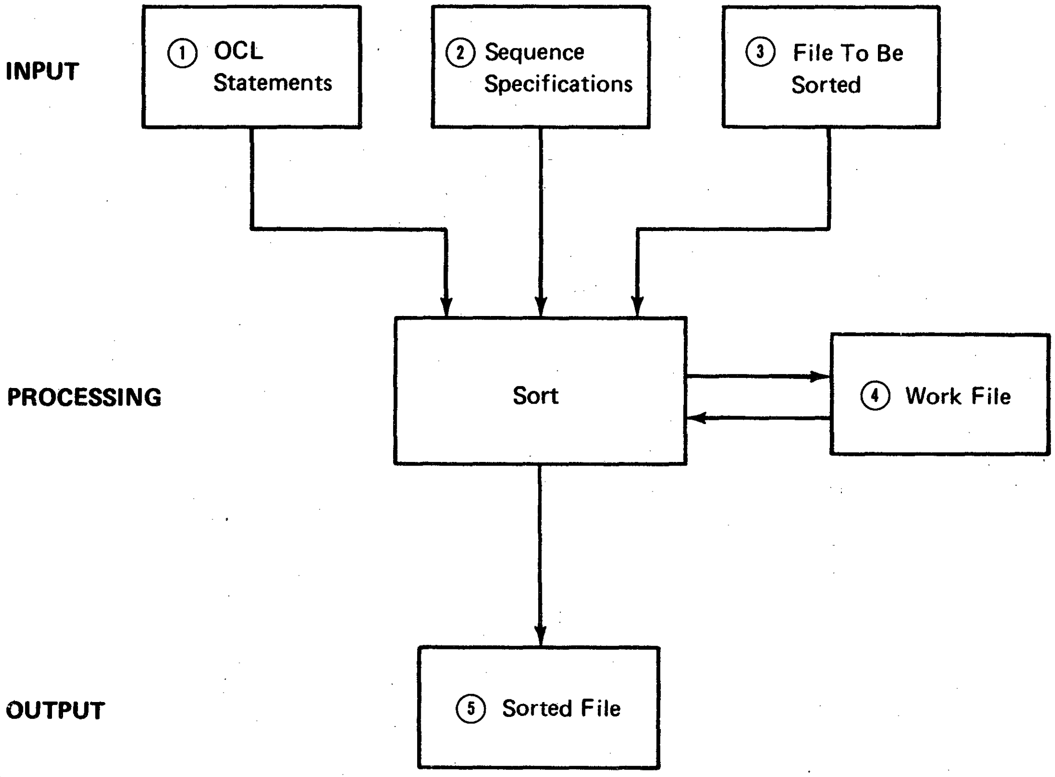
A summary tagalong sort that sorts only on the first two digits of the control field works as follows (note the summarizing that occurs in the data fields that have a 17 in the control field):



Summary Tagalong Sorts

HOW SORT WORKS

The input needed to run a sort job is: an input file to be sorted; sequence specifications; and operation control language (OCL) statements. Sort works as follows:



① *OCL Statements* are your instructions to the system.

② *Sequence Specifications* are your instructions to the sort program.

③ *File to be Sorted* can be an indexed, sequential, or direct disk file with fixed length records, but sort processes each sequentially.

④ *Work File* is a work area on disk where sort writes all of the records you want to sort. The program can then work with the records in the work area without disturbing the input file.

⑤ *Sorted File* can contain:

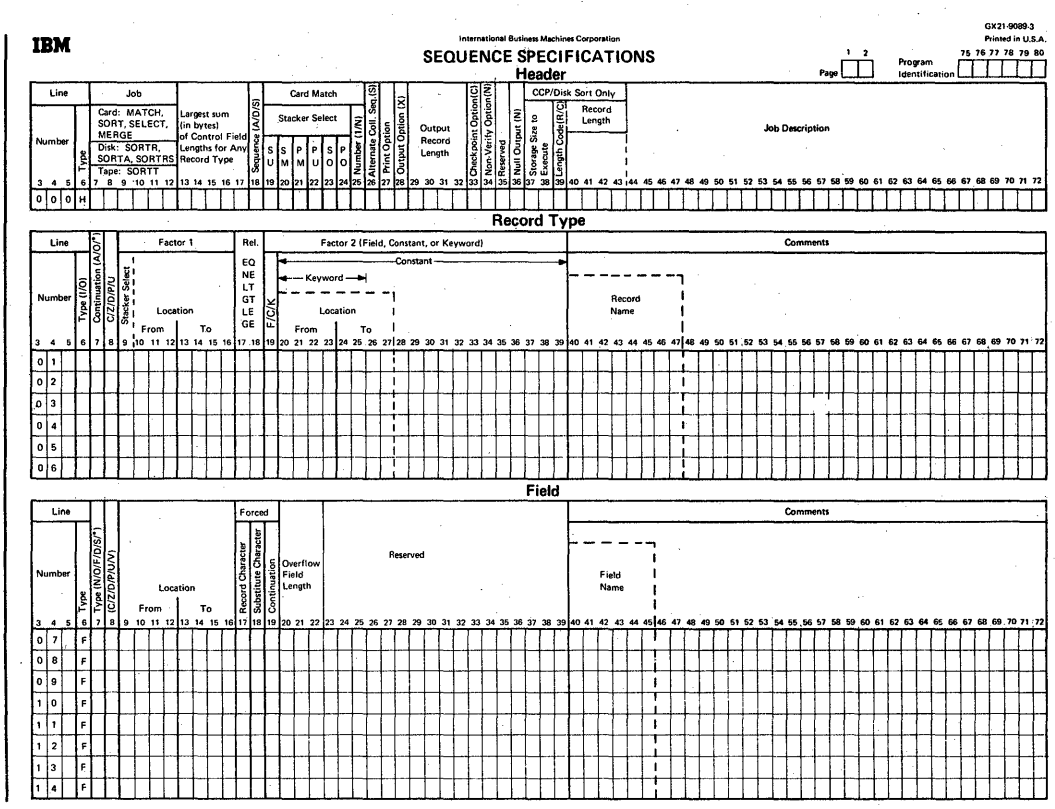
- The relative record numbers of the records in the input file.
- All or part of the records contained in the input file.
- Summarized fields for each record type in the input file.

Sort Sequence Specifications

You must always specify how you want a file sorted. You use sequence specifications to describe the type of sort you want, which records you want to sort, and how you want the records sorted. Sequence specifications can be stored as a source member or within a procedure member in the library; these specifications can also be entered through the keyboard. See Chapter 5, *How to Run Sort* for a detailed description of how to supply sequence specifications for your sort job.

Sequence Specifications Sheet

The sequence specifications sheet contains three different kinds of specifications: header specification, record type specifications, and field specifications. The sample sequence specifications sheet shows the location and purpose of all three:



IBM

International Business Machines Corporation

GX21-9089-3
Printed in U.S.A.

SEQUENCE SPECIFICATIONS

Page (1 2) □□ Program Identification (75 76 77 78 79 80) □□□□□□

Header

Line Number (3 4 5)	Type (6)	Job (7–12): Card: MATCH, SORT, SELECT, MERGE; Disk: SORTR, SORTA, SORTRS; Tape: SORTT	Largest sum (in bytes) of Control Field Lengths for Any Record Type (13–17)	Sequence (A/D/S) (18)	Card Match Stacker Select SU (19)	SM (20)	PM (21)	PU (22)	SO (23)	PO (24)	Number (1/N) (25)	Alternate Coll. Seq.(S) (26)	Print Option (27)	Output Option (X) (28)	Output Record Length (29–32)	Checkpoint Option(C) (33)	Non-Verify Option(N) (34)	Reserved (35)	Null Output (N) (36)	CCP/Disk Sort Only: Storage Size to Execute (37–38)	Length Code(R/C) (39)	Record Length (40–43)	Job Description (44–72)
0 0 0	H																						

Here you specify the type of sort job you will run.

Record Type

Line Number (3 4 5)	Type (I/O) (6)	Continuation (A/O/*) (7)	C/Z/D/P/U (8)	Factor 1: Stacker Select; Location From (9–12)	Location To (13–16)	Rel. EQ NE LT GT LE GE (17–18)	F/C/K (19)	Factor 2 (Field, Constant, or Keyword): Constant; Keyword; Location From (20–23)	To (24–27)	(28–39)	Comments: Record Name (40–47)	(48–72)
0 1												
0 2												
0 3												
0 4												
0 5												
0 6												

Here you specify which input file records you will sort.

Field

Line Number (3 4 5)	Type (6)	Type (N/O/F/D/S/*) (7)	(C/Z/D/P/U/V) (8)	Location From (9–12)	Location To (13–16)	Forced: Record Character (17)	Substitute Character (18)	Continuation (19)	Overflow Field Length (20–22)	Reserved (23–39)	Comments: Field Name (40–45)	(46–72)
0 7	F											
0 8	F											
0 9	F											
1 0	F											
1 1	F											
1 2	F											
1 3	F											
1 4	F											

Here you specify how you want the records sorted and formatted.

Sort Execution

You can create your own procedure member to execute sort. This procedure member is stored in the library. The contents of a procedure member that contains your OCL are explained in Chapter 5, *How to Run Sort.*

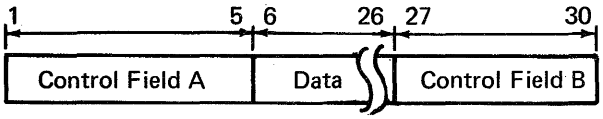
You can include sequence specifications in a procedure member (following the RUN statement), or you can access the source member that contains your sequence specifications through use of the SOURCE statement. See *Source Member Information Needed When Using User-Supplied Sort Procedures* in Chapter 5.

The DFU SORT command can also be used to execute sort without providing OCL statements. This command is described in Appendix D of the *IBM System/32 Utilities Program Product Reference Manual—Data File Utility,* SC21-7600.

Sort Operation

Sort works with the records in your file as follows:

1. Sort reads a record from the input file.



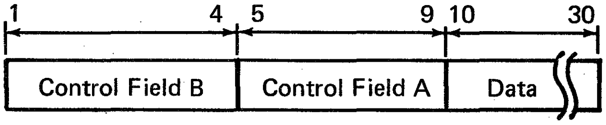
Note: Control fields are the fields you want the sort program to use to sort the records. See *Column 7* under *Column Descriptions* in Chapter 4 for a detailed discussion of control fields.

2. Sort checks your record type specifications to see whether the record is one you want to sort. (Often you may not want to sort all of the records in the file.)

3. If the record is one you want to sort, sort builds a work record, formatting it according to your field specifications. (The format of the work record is important because it controls the format of the output record.)

   Assume that your field specifications say:
   a. Put the contents of positions 27 through 30 in the input record into positions 1 through 4 of the work record.
   b. Put the contents of positions 1 through 5 in the input record into positions 5 through 9 of the work record.
   c. Put the contents of positions 6 through 26 in the input record into positions 10 through 30 of the work record.

   The work record sort builds would look like this:



   See *Column 7* under *Column Descriptions* in Chapter 4 for a more detailed discussion of the work record.

   If you do not drop control fields, they will always precede data fields in the work and output records. (See *Column 28* under *Column Descriptions* in Chapter 2 for a discussion of how to drop control fields.)

4. Sort writes all of the input records you want to sort into the work file.

5. Sort checks your header and field specifications to see how you want to arrange the records in the output file. (You arrange the records in ascending or descending order by control fields.)

6. Sort writes the records into the output file in the order you have specified.

OVERVIEW OF SEQUENCE SPECIFICATIONS

Order of Sequence Specifications

The normal order of sequence specifications is:

1. Header line

2. Record type lines

3. Field lines

This order can vary, however, depending on how many records you want to sort and the format of those records. (Format refers to the locations, lengths, and types of fields in a record.) If, for example, you want to sort all of the records in a file—and they all have the same format—you do not have to fill out record type specifications. (Not filling out record type specifications is often referred to as an implied include-all.)

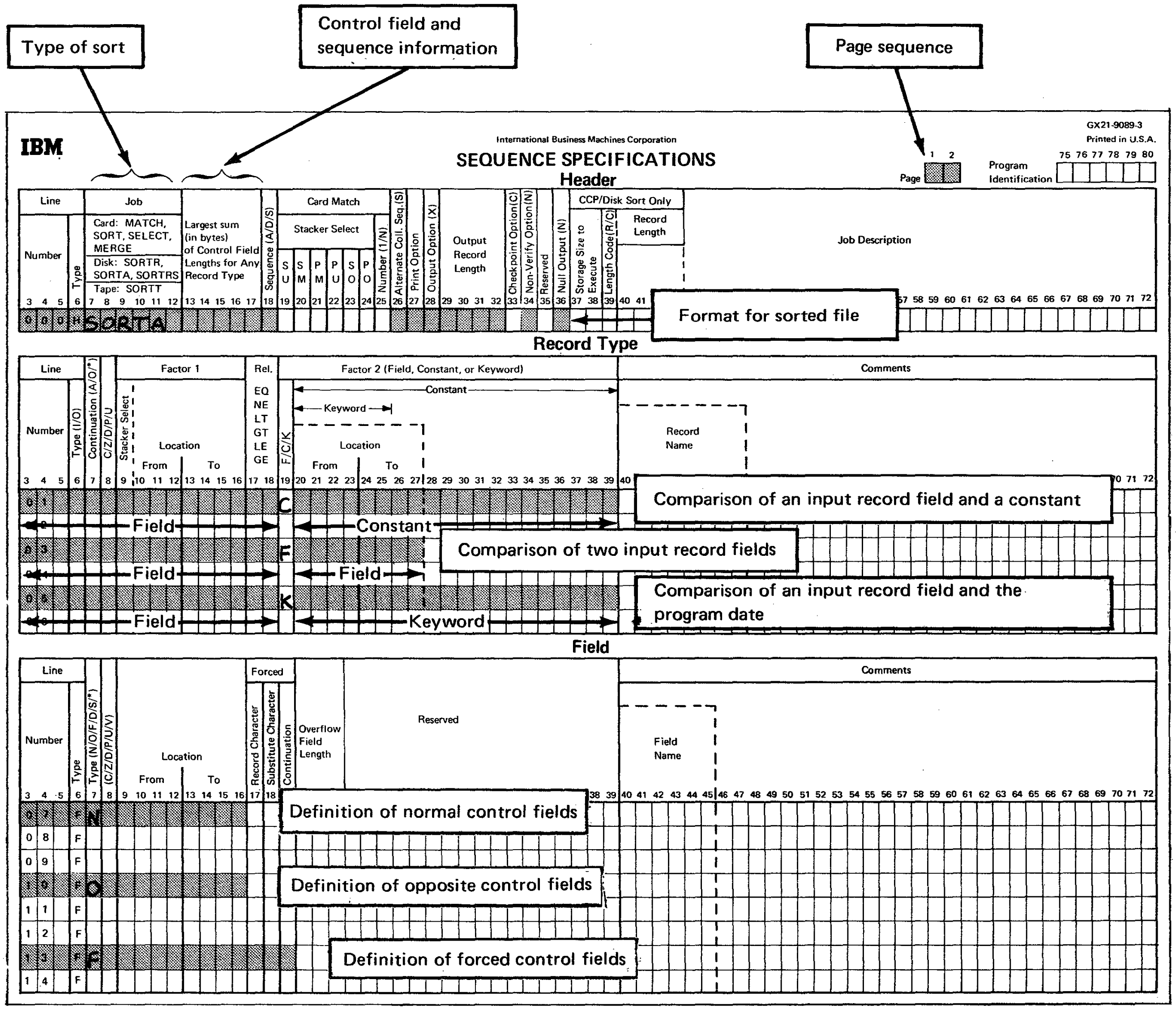
Number of Records to be Sorted	Format of Records to be Sorted	Order of Sequence Specifications
All of the records in the file (implied include-all)	All the same format	1. Header line 2. Field lines
Some of the records in the file	All the same format	1. Header line 2. Record type lines 3. Field lines
All or some of the records in the file	Several different formats	1. Header line 2. Record type lines 3. Field lines For first type of record format 4. Record type lines 5. Field lines For second type of record format 6. Record type lines 7. Field lines One set for each additional type of record format[1]

[1] This does not mean that the records in the file must be grouped by format type. The sequence specifications must be grouped (one set of record type and field lines for each format type). The records themselves can be in any order.

For detailed information on how to include or omit certain records when you are sorting a file, see *Column 6* under *Column Descriptions* in Chapter 3.

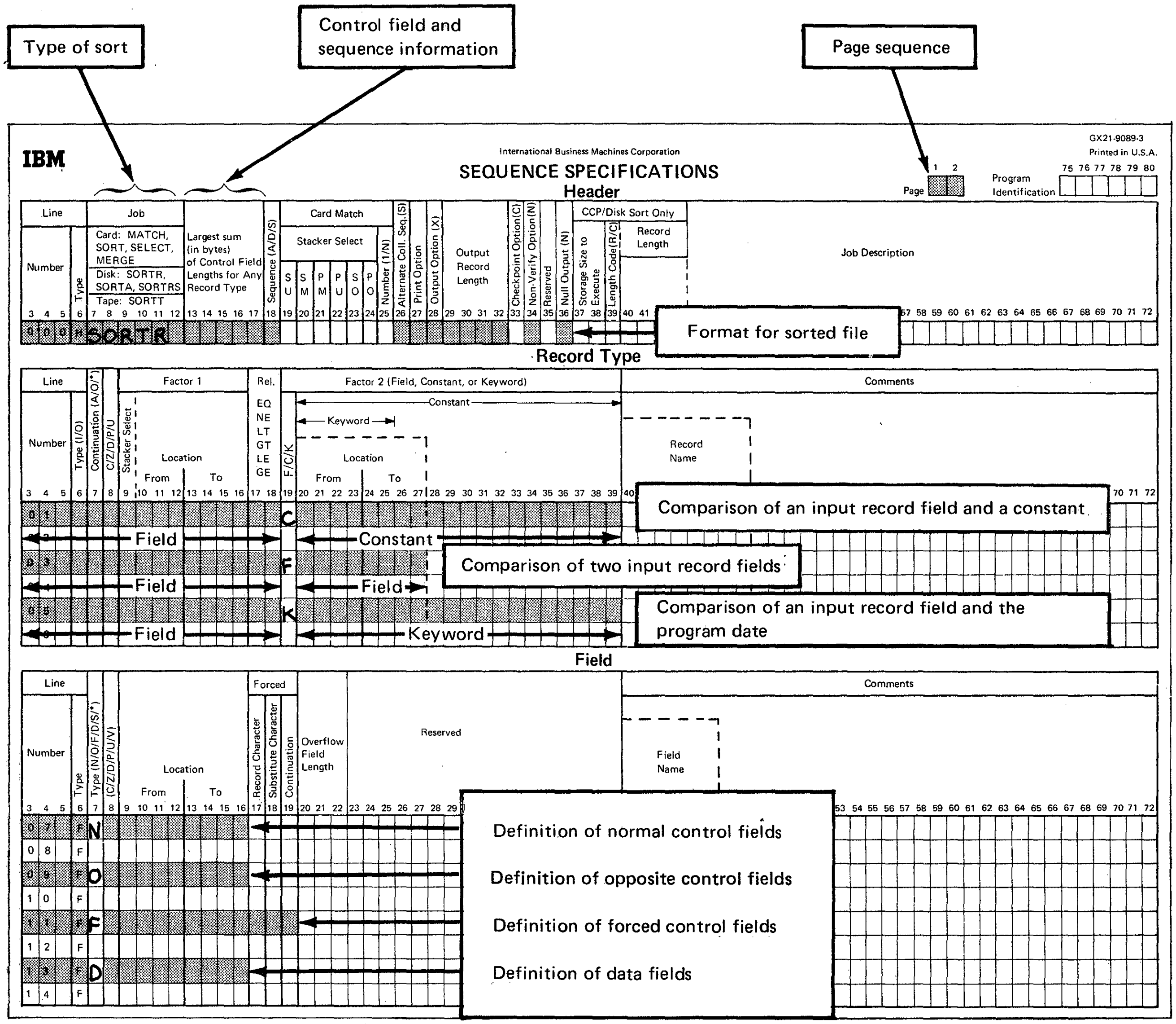
Specification Columns to Consider for Addrout Sort (SORTA)

The shaded columns are the ones you must consider for an addrout sort job.



Specification Columns to Consider for Tagalong Sort (SORTR)

The shaded columns are the ones you must consider for a tagalong sort job.



Type of sort

Control field and sequence information

Page sequence

IBM

International Business Machines Corporation

SEQUENCE SPECIFICATIONS

GX21-9089-3
Printed in U.S.A.

Page 1 2

Program Identification 75 76 77 78 79 80

Header

Line: Number (3 4 5), Type (6)

Job (7–12): Card: MATCH, SORT, SELECT, MERGE; Disk: SORTR, SORTA, SORTRS; Tape: SORTT

Largest sum (in bytes) of Control Field Lengths for Any Record Type (13–17)

Sequence (A/D/S) (18)

Card Match: Stacker Select: SU (19), SM (20), PM (21), PU (22), SO (23), PO (24)

Number (1/N) (25)

Alternate Coll. Seq.(S) (26)

Print Option (27)

Output Option (X) (28)

Output Record Length (29–32)

Checkpoint Option(C) (33)

Non-Verify Option(N) (34)

Reserved (35)

Null Output (N) (36)

CCP/Disk Sort Only: Storage Size to Execute (37 38), Length Code(R/C) (39), Record Length (40 41)

Job Description

0 0 0 H SORTR

Format for sorted file

Record Type

Line: Number (3 4 5), Type (I/O) (6), Continuation (A/O/*) (7), C/Z/D/P/U (8), Stacker Select (9)

Factor 1: Location From (10–12) To (13–16)

Rel. (17 18): EQ NE LT GT LE GE

F/C/K (19)

Factor 2 (Field, Constant, or Keyword): Constant; Keyword; Location From (20–23) To (24–27)

Comments: Record Name

0 1 C — Field — Constant

Comparison of an input record field and a constant

0 3 F — Field — Field

Comparison of two input record fields

0 5 K — Field — Keyword

Comparison of an input record field and the program date

Field

Line: Number (3 4 5), Type (6), Type (N/O/F/D/S/*) (7), (C/Z/D/P/U/V) (8)

Location From (9–12) To (13–16)

Forced: Record Character (17), Substitute Character (18)

Continuation (19)

Overflow Field Length (20–22)

Reserved (23–)

Comments: Field Name

0 7 F N — Definition of normal control fields

0 8 F

0 9 F O — Definition of opposite control fields

1 0 F

1 1 F F — Definition of forced control fields

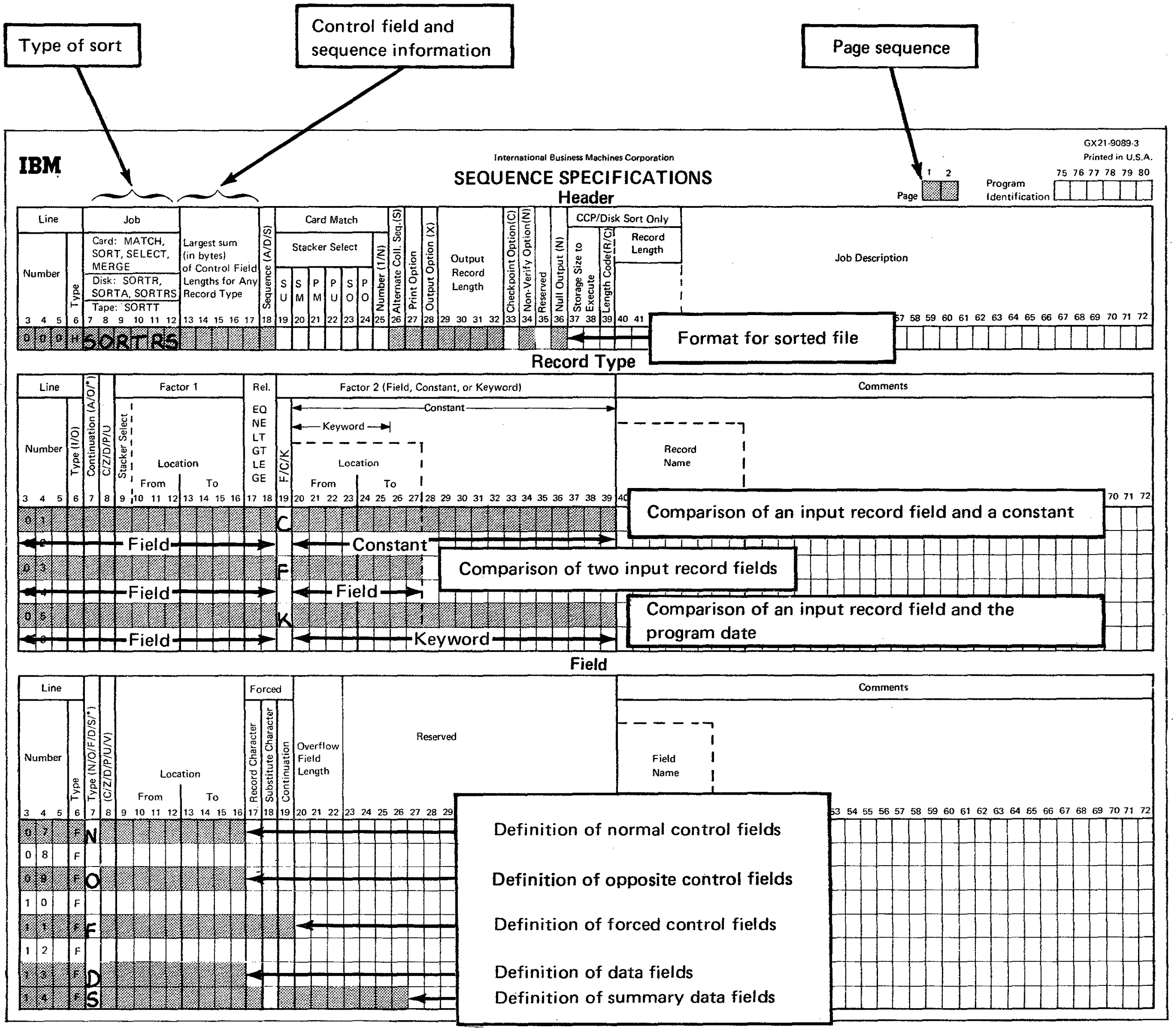
1 2 F

1 3 F D — Definition of data fields

1 4 F

Specification Columns to Consider for Summary Tagalong Sort (SORTRS)

The shaded columns are the ones you must consider for a summary Tagalong sort job.



Type of sort

Control field and sequence information

Page sequence

IBM International Business Machines Corporation

SEQUENCE SPECIFICATIONS

GX21-9089-3 Printed in U.S.A.

Page 1 2 Program Identification 75 76 77 78 79 80

Header

Line Number 3 4 5 | Type 6 | Job: Card: MATCH, SORT, SELECT, MERGE; Disk: SORTR, SORTA, SORTRS; Tape: SORTT 7 8 9 10 11 12 | Largest sum (in bytes) of Control Field Lengths for Any Record Type 13 14 15 16 17 | Sequence (A/D/S) 18 | Card Match: Stacker Select SUM 19 20 21 22 23 24 | Number (1/N) 25 | Alternate Coll. Seq.(S) 26 | Print Option 27 | Output Option (X) 28 | Output Record Length 29 30 31 32 | Checkpoint Option(C) 33 | Non-Verify Option(N) 34 | Reserved 35 | Null Output (N) 36 | CCP/Disk Sort Only: Storage Size to Execute 37 38 | Length Code(R/C) 39 | Record Length 40 41 | Job Description

0 0 0 H SORTRS

Format for sorted file

Record Type

Line Number | Type (I/O) | Continuation (A/O/*) | C/Z/D/P/U | Stacker Select | Factor 1 Location From To | Rel. EQ NE LT GT LE GE | F/C/K | Factor 2 (Field, Constant, or Keyword): Constant, Keyword, Location From To | Comments: Record Name

0 1 C — Comparison of an input record field and a constant

0 2 Field — Constant

0 3 F — Comparison of two input record fields

0 4 Field — Field

0 5 K — Comparison of an input record field and the program date

0 6 Field — Keyword

Field

Line Number | Type | Type (N/O/F/D/S/*) | (C/Z/D/P/U/V) | Location From To | Forced: Record Character, Substitute Character | Continuation | Overflow Field Length | Reserved | Comments: Field Name

0 7 F N — Definition of normal control fields

0 8 F

0 9 F O — Definition of opposite control fields

1 0 F

1 1 F F — Definition of forced control fields

1 2 F

1 3 F D — Definition of data fields

1 4 F S — Definition of summary data fields

Only one header specification line is used for each sort job. Possible column entries, summarized on this page and in Appendix C for quick reference, are explained in detail later in this chapter.

HEADER SPECIFICATION COLUMN SUMMARY

Column Summary		
Columns	**Entries**	**Explanation**
1-2	00-99	Page number.
3-5	000	Header line number.
▒ 6	H	Header line identification.
▒ 7-12	SORTA	Addrout sort job.
▒	SORTR	Tagalong sort job.
▒	SORTRS	Summary tagalong sort job.
▒ 13-17	1-256	Largest sum (in bytes) of control field lengths for any record type.
▒ 18	A	Records in sorted file to be in ascending order by control fields.
▒	D	Records in sorted file to be in descending order by control fields.
▒	S	Not used in System/32 sort jobs.
19-25	Blank	Not used in System/32 sort jobs.
26	Blank	Use standard collating sequence in compare operations.
	S	Use an alternate collating sequence in compare operations. ALTSEQ statements will define the collating sequence to be used.
27	0 or Blank	Print: Sequence specifications. Diagnostic messages. Program-status messages. Action messages. Formatted messages.
	1	Print: Program-status messages. Action messages. Formatted messages.
	2	Print: Action messages. Formatted messages.
	3	Print nothing (except formatted messages).
28	Blank	Keep control fields in output records in tagalong sort jobs.
	X	Drop control fields from output records in tagalong sort jobs.
▨ 29-32	1-4096	Length (in bytes) of output records in tagalong sort jobs.
33	Blank	Not used in System/32 sort jobs.
34	N	Data written on the work file will not be verified.
	Blank or character other than N	Data written on the work file will be verified.
35	Blank	Reserved for system use.
36	Blank	Operator message SORT-7724 will be issued.
	N	Operator message SORT-7724 will not be issued.
37-39	Blank	Not used in System/32 sort jobs.
40-72	Any characters	Comments.
	▒	Columns that must be filled in for all sort jobs.
	▨	Columns that must be filled in for a tagalong sort.

COLUMN DESCRIPTIONS

Columns 1-2 (Page Number) and 3-5 (Line Number)

Page number (columns 1 and 2) and line number (columns 3 through 5) form a 5-digit sequence number. As the sort program reads sequence specifications, it checks the sequence numbers to make sure they are not in descending order. If the numbers are in descending order (if page 02 specifications come before page 01 specifications, for example) and specifications are being issued the sort program places a warning (the letter S) next to the line. (The S stands for sequence error.) After issuing a warning the program continues reading the rest of the specification lines and then issues a message and waits for further instructions from the operator. The operator can either continue or end the job.

Because page number applies to all lines on a page, columns 1 and 2 appear only once, in the upper right corner of the page. You number the pages in ascending order.

The line number of the header line is always 000 and is preprinted on the coding sheet.

Column 6 (Line Type)

Column 6 of the header line contains a preprinted H to identify the line as a header line.

Columns 7-12 (Job)

Columns 7 through 12 tell the program what type of sort job you want to do. SORTA means addrout sort. SORTR means tagalong sort. SORTRS means summary tagalong sort.

Columns 13-17 (Largest Sum (in bytes) of Control Field Lengths for Any Record Type)

To calculate this entry:

1. Add together the lengths of the control fields (N, O, and F in column 7 of field specifications) for each type of input record.
2. Enter the largest of these totals in columns 13 through 17. (The total must not exceed 256.)

For more information, see *Control Fields, Column 7* under *Column Descriptions* in Chapter 4.

Column 18 (Ascending or Descending Sequence)

Control fields control the sequence of records in the sorted output file. Your column 18 entry indicates the sequence in which you want the records sorted.

Column 18 Entry	Sequence
A	Ascending sequence by control field
D	Descending sequence by control field
S	Not used in System/32 sort jobs

Column 26 (Collating Sequence)

Column 26 specifies the collating sequence you want sort to use in compare operations. (Compare operations determine whether one character is equal to, greater than, or less than another character.)

Standard Collating Sequence

No entry in column 26 tells the sort program to use the standard collating sequence. There are slight variations in the standard collating sequence, depending on whether you are using both the zone and digit portions of the characters in your records, the zone portions only, or the digit portions only. Appendix B shows the complete collating sequence of each situation.

Alternate Collating Sequence

An S in column 26 tells the program you want to change the standard collating sequence. To do this you must supply ALTSEQ statements immediately following the header specifications. (Appendix B explains how to code ALTSEQ statements.)

Note: Do not use a packed or unpacked factor in an include or omit record type specification (P or U in column 8) if you specify an alternate collating sequence.

Column 27 (Print Option)

Sort can issue:

- Sequence specification lines
- Diagnostic messages (S, T, and W type messages), for any errors in sequence specifications
- Program-status messages (I type messages) to identify various states of the job
- Action messages (A type messages, followed by formatted message displays), to identify circumstances requiring attention before you can continue the job

Note: The programmer messages are contained in Appendix E, *Sort Programmer Messages.* The operator messages are contained in the *IBM System/32 Displayed Messages Guide*, GC21-7704.

Column 27 indicates which of the preceding information you want the sort program to issue during a job:

Column 27 Entry	Program Issues
0 or Blank	Sequence specifications Diagnostic messages Program-status messages Action messages Formatted messages
1	Program-status messages Action messages Formatted messages
2	Action messages only Formatted messages
3	Formatted messages

Column 28 (Output Option for Tagalong Sorts)

Column 28 applies to tagalong sort (SORTR, SORTRS) jobs only. It indicates whether or not you want the program to drop control fields from output records after the records are sorted. A blank in column 28 means keep the control fields; X means drop them.

Considerations for Dropping Control Fields

Control fields are normally dropped when you use opposite control fields or an alternate collating sequence. In these two cases, the program changes the control information (during the sorting process) so that it is meaningless.

Using Fields Both as Control and Data Fields

If you are using opposite control fields or an alternate collating sequence and you want to keep the control information in a meaningful form in the output records, describe the fields twice: once as control fields and once as data fields. Data fields are not involved in the sorting process and are not changed by the program.

Columns 29-32 (Output Record Length (in bytes) for Tagalong Sorts)

Columns 29 through 32 apply to tagalong sort (SORTR, SORTRS) jobs only. The entry in these columns tells the program the length of records in the final sorted file.

If you do not drop control fields, the length includes both control and data fields. If you drop control fields, the length includes only data fields. Record length must be 1 through 4096 bytes.

Calculating Output Record Length When Dropping Control Fields

For each type of record, total the lengths of all the data fields you are including in the job. Select the largest total. Enter this number in columns 29 through 32.

Calculating Output Record Length When Not Dropping Control Fields

For each type of record, total the lengths of the data fields. Select the largest total, add this total to the number in columns 13 through 17, and put the sum in columns 29 through 32.

Column 34 (Nonverify Option)

Column 34 applies to all sort runs. If an N is placed in this column of the sort header statement, none of the data written on the work file will be verified. A blank or a character other than N will cause the data written on the work file to be verified.

Note: If you use the nonverify option (to minimize run time) when your output file is to overlay the input file, the input file might be destroyed if a terminal error occurs before end of job. Therefore, if the input file cannot be easily recreated, you should have a duplicate copy for backup. The input file can be destroyed under identical circumstances whether the nonverify option is used or not. However, the chance for input file destruction is greater when using the nonverify option; therefore, use this option with care.

Column 36 (Null Output—Bypass Halt for MSG SORT-7724)

Column 36 applies to all sort runs. A blank can cause sort to issue message SORT-7724 NO INPUT RECORDS INCLUDED. The operator can either select option 0, which creates an empty output file, or option 3, which cancels the job without creating an output file. An N in column 36 causes sort to create an empty output file without issuing message SORT-7724.

Columns 40-72 (Job Description)

Columns 40 through 72 are for your comments. You can use any characters you want in these columns. If the program prints specification lines (the column 27 entry is a zero or blank), the comments you include in these columns are printed. Comments have no effect on the program.

Chapter 3. Record Type Specifications

Record type specifications tell sort which of the records in a file are to be sorted. Remember, if all of the records in a file are to be sorted, and they all have the same format, record type specifications are not required.

Possible column entries, summarized on this page and in Appendix C for quick reference, are explained in detail in this chapter.

RECORD TYPE SPECIFICATION COLUMN SUMMARY

Column Summary

Columns	Entries	Explanation
1-2	00-99	Page number.
3-5	01x-06x	Line number. You can leave column 5 (x) blank, or enter any value to keep the specifications in ascending order.
6	I	Include line.
	O	Omit line.
7	Blank	This is the first line of a set of I *or* O record type lines.
	A	AND lines. These specifications continue the definition of the record described on the preceding line.
	O	OR line. These specifications define a different type of record than the one on the previous line.
	*	Comment line.
8	C	Use both zone and digit portions of characters.
	Z	Use only zone portion of a 1-character field.
	D	Use only digit portions of characters.
	P	Signed packed decimal data.
	U	Signed unpacked decimal data.
9-12	1-4096	The input record position in which the factor 1 field begins (blank if field is only one position long).
13-16	1-4096	The input record position in which the factor 1 field ends.
17-18	EQ	Factor 1 must equal factor 2.
	NE	Factor 1 must not equal factor 2.
	LT	Factor 1 must be less than factor 2.
	GT	Factor 1 must be greater than factor 2.
	LE	Factor 1 must be less than or equal to factor 2.
	GE	Factor 1 must be greater than or equal to factor 2.
19	F	Factor 2 is another field in the same input record.
	C	Factor 2 is a constant.
	K	Factor 2 is a keyword.
20-23	1-4096	The input record position in which the factor 2 field begins (blank if field is only one position long).
24-27	1-4096	The input record position in which the factor 2 field ends.
20-39	Any characters	The factor 2 constant.
40-72	Any characters	Comments.

▨ Columns that must be filled in.

COLUMN DESCRIPTIONS

Columns 1-2 (Page Number) and 3-5 (Line Number)

Page number (columns 1 and 2) and line number (columns 3 through 5) form a 5-digit sequence number. As the sort program reads sequence specifications, it checks the sequence numbers to make sure they are not in descending order. If the numbers are in descending order (if page 02 specifications come before page 01 specifications, for example) and specifications are being issued, the sort program places a warning (the letter S) next to the line. (The S stands for sequence error.) After issuing a warning, the program continues reading the rest of the specification lines, then issues a message and waits for further instructions from the operator. The operator can either continue or end the job.

Because page number applies to all lines on a page, columns 1 and 2 appear only once, in the upper right corner of the page. You number the pages in ascending order.

Record type line numbers are 01 through 06. The numbers in columns 3 and 4 are preprinted on the coding sheet. You can leave column 5 blank or enter any value to keep your specifications in ascending order. If you have more than six record type lines, use another coding sheet and start at line 01.

Out-of-Sequence Lines

Use column 5 when you want to insert a specification without renumbering the other specifications. For example, to insert a specification line between lines 01010 and 01020, you can number it 01015, code it, and then continue to fill out the rest of the sheet.

Be sure any lines that are out of sequence on your coding sheet are clearly marked. You can do this by writing a note in the margin of the page with an arrow pointing to where the insert belongs. (*Sample Job 3* in Chapter 7, *Sample Sort Jobs.*)

Column 6 (Line Type)

Column 6 identifies the line type. An I in this column stands for either an include or an include-all line; an O stands for an omit line.

If the type of sort job you are running requires you to code record type lines, you must use include or include-all lines to describe the records you want sorted. Records not described in include lines are not sorted.

Include Line

Include lines identify records you want the program to sort by describing particular record fields.

Include-All Line

Include-all is a special form of include line; it has no record description (columns 7 through 39 are blank). An include-all line tells the program to sort all of the records that have not been described by any preceding include or omit line for the job. Records referred to by an include-all line must have the same field specifications.

Note: Only one include-all line can be used per job. If used, it must be the last record type line for that job.

Omit Line

Omit lines identify records you do *not* want the program to sort. Omit lines are not required, but can be helpful when you have many types of records you want the program to use and just a few you want omitted. Omit lines are normally followed by an include-all line, telling the program to sort all of the records that are not described by omit lines.

Sets

There are two types of sets: include sets and omit sets. An include set identifies one or more record types you want to include in your sort job. An omit set identifies one or more record types you want to omit from your sort job. (The records in any record type always have at least one characteristic in common–such as an X in position 5.)

Here are four rules to remember when you are using include and omit sets:

1. All include sets must end with a field line.
2. Omit sets never have field lines.
3. Every omit set must be followed by an include set.
4. The last set must be an include set.

There are five types of include sets and three types of omit sets.

Include Sets[1]

Type	Column 6	Column 7	Explanation	
Include AND lines	H, F, or O		Header line, field line, or omit line.	
	I	ƀ	New record type indicated by a blank in column 7.	Record Type Lines
	I	A	Lines that describe the same record type (as the previous line) have an A in column 7.	
	F		Field line(s).	
Include OR lines	H, F, or O		Header line, field line, or omit line.	
	I	ƀ	New record type indicated by a blank in column 7.	Record Type Lines
	I	O	Lines that describe a different record type (than the previous line) have an O in column 7.	
	F		Field line(s).	
Include AND and OR lines	H, F, or O		Header line, field line, or omit line.	
	I	ƀ	New record type indicated by a blank in column 7.	Record Type Lines
	I	O	This line designates a record type which is different than, but has the same field lines as, the record type described in the previous line(s).	
	I	A	This line continues the same record type of a previous line or lines. This record type can be continued (IA), or a different record type can be started (IO), provided all record types have the same field lines. Record types with different field lines would have to start a new include set.	
	F		Field line(s) for either A or O record types.	
Include only one record type (implied include-all)	H		Header line.	
			No record type lines.	
	F		Field line(s).	
Include-all	H, F, or O		Header line, field line, or omit line.	
	I		Tells the sort program to sort all of the records that have not been described by any preceding include and omit lines. Records referred to in this manner must have identical field specifications.	Record Type Line
	F		Field line(s).	

[1] Every include set must end with field lines. An include set can be followed by another include set, an omit set, or // END.

Note: Records not described in include sets will not be sorted.

Omit Sets[1]

Type	Column 6	Column 7	Explanation	
Omit AND lines (one record type)	H or F		Header line or field line (last line of include set).	
	O	ƀ	New record type indicated by a blank in column 7.	Record Type Lines
	O	A	Lines that describe the same record type (as the previous line) have an A in column 7.	
Omit OR lines (different record types)	H or F		Header line or field line (last line of include set).	
	O	ƀ	New record type indicated by a blank in column 7.)	Record Type Lines
	O	O	Lines that describe a different record type (than the previous line) have an O in column 7.	
Omit AND and OR lines (different record types)	H or F		Header line or field line (last line of include set).	
	O	ƀ	New record type indicated by a blank in column 7.	Record Type Lines
	O	A	Lines that describe the same record type (as the previous line) have an A in column 7.	
	O	O	Lines that describe a different record type (than the previous line) have an O in column 7.	

[1] There are no field lines in omit sets. Each omit set must be followed by an include or an include-all set.

Guide to Using Include and Omit Sets

When to Use Include Sets: If you want to sort only a few records in a file, use an include set for each type of record you want to sort.

When to Use Omit Sets: If you want to sort all but a few records in a file, use omit sets followed by either an include set for each type of record you want to sort or an include-all set.

Mixing Include and Omit Sets: You can mix include and omit sets. But because sort processes the sets in the order they are coded, you must be particularly careful when you do this. For example, if you wanted to omit all records with a 2 in position 10 but sort those with a 2 in positions 10 and 15, you would have to specify the include set before the omit set. If you specified the omit set first, all the records you wanted to sort would be omitted from the job.

Column 7 (Continuation or Comments)

Column 7 indicates the line's relationship to the preceding line.

Column 7 Entry	Explanation
Blank	This line is the first of a set of include or omit lines. (The type of set is indicated by the column 6 entry; I for include or O for omit.)
A	This line is a continuation of the preceding line. The A stands for AND.
O	This line applies to a different record type than the preceding line, but the control field specifications for both are the same. The O stands for OR.
*	This line is a comment line. Comment lines do not affect program operation in any way. Their only purpose is to help you remember what you were doing in a certain section of coding. Comments are printed only if column 27 of the header line contains a zero or a blank.

Column 8 (C/Z/D/P/U)

Your column 8 entry tells sort how to interpret data in the factor 1 and factor 2 fields during compare operations. When the fields contain alphameric data, a C, Z, or D entry tells sort what portions of the characters to use. When the fields contain signed numeric data, a P or U entry tells sort whether the data is packed or unpacked.

	Column 8 Entry	Compare Operations	Maximum Field Length[1]
Alphameric Data	C	Use both zone and digit portions of the characters	256 characters
	Z	Use only the zone portion of the character	1 character
	D	Use only the digit portions of the characters	16 characters
Signed Numeric Data	P[2]	The numeric data is packed	8 bytes or 15 digits and a sign
	U[2]	The numeric data is unpacked	16 digits

[1] For both factor 1 and factor 2 fields, which must be the same length.

[2] Do not use a packed or unpacked factor in an include or omit record type specification (P or U in column 8) if you specify an alternate collating sequence (S in column 26 of the header line).

Significance of the Column 8 Entry

You use record type specifications to tell sort which records you want to sort. You do this by instructing sort to test each record by comparing the data in a specific field with the data in another field in the same record, with a constant, or with all or part of the program date. The data you are comparing is the factor 1 field; the data you are comparing it with (the contents of another field in the same record, a constant, or all or part of the program date) is the factor 2 field. The result of the comparison determines whether or not that record will be sorted. Because sort sees your data as nothing more than a series of electronic bits, you must tell the program how to interpret the data:

- If the data is alphameric, what part of the characters should be compared.
- If the data is signed numeric, is it in packed or unpacked format.

Interpreting Data

EBCDIC characters are represented by a byte composed of 8 bits.

Each character has two parts: a zone portion and a digit portion. Some characters have identical zone portions; some have identical digit portions. No two characters have identical zone and digit portions.

	How It Looks Inside the Computer	
Character	Zone Portion	Digit Portion
*	0101	1100
1	1111	0001
2	1111	0010
3	1111	0011
K	1101	0010
?	0110	1111
P	1101	0111
blank	0100	0000 [1]
0	1111	0000

[1] Notice that the digit portions of a zero and a blank are exactly the same.

If you instruct sort to use only the digit portions of characters (by putting a D in column 8), characters with identical digit portions will look alike and compare as equal. Likewise, if you instruct sort to use only the zone portion of characters (by putting a Z in column 8), characters with identical zone portions will look alike and compare as equal. So your column 8 entry is critical in ensuring that your compare operations produce the results you intend.

Suppose, for example, that you want to sort only records with a 2 in column 15 and a 2 in column 50. To get those records included in your sort, you would have to put a C in column 8. The C tells the sort program to use both the zone and digit portions of characters in its compare operations, and no other character has the same zone and digit portions as a 2.

If you put a D in column 8, you would also get the records with a 2 in column 15 and a 2 in column 50. However, you would also get a lot of records you did not want, because several characters have the same digit portion as a 2.

Alphameric Data

When the factor 1 and factor 2 fields contain alphameric data, the column 8 entry must specify one of the following:

1. The zone and digit portions of the characters (C entry).

2. Only the zone portion of the character (Z entry).

3. Only the digit portions of the characters (D entry).

Numeric Data

When the factor 1 and factor 2 fields contain numeric data, the column 8 entry must answer two questions:

1. Are the numbers signed or unsigned?

2. Are the numbers packed or unpacked?

Format of Numeric Data		Column 8 Entry	Definition	Consideration[1]
Unsigned		D	Number does not have a sign.	Absolute values of the numbers are used. If a number has a sign, it is ignored. For example, -3 would be considered equal to +3.
Signed	Packed	P	Number always carries a sign. When the number is placed in main storage it has a digit portion only.	The sign controls the comparison. For example, -3 is less than 0 and +5 is more than -6.
	Unpacked	U	Number always carries a sign. When the number is placed in main storage, it has both a zone and a digit portion	The sign controls the comparison. For example, -3 is less than 0 and +5 is more than -6.

[1] Before comparing numeric data, sort automatically converts any leading blanks to zeros.

Signed Numbers: Signed numbers can be either positive or negative. The sign of a number is indicated by a 4-bit binary code.

Sign	Binary Code
+	1111[1]
+	1010
+	1100
+	1110
–	1101
–	1011

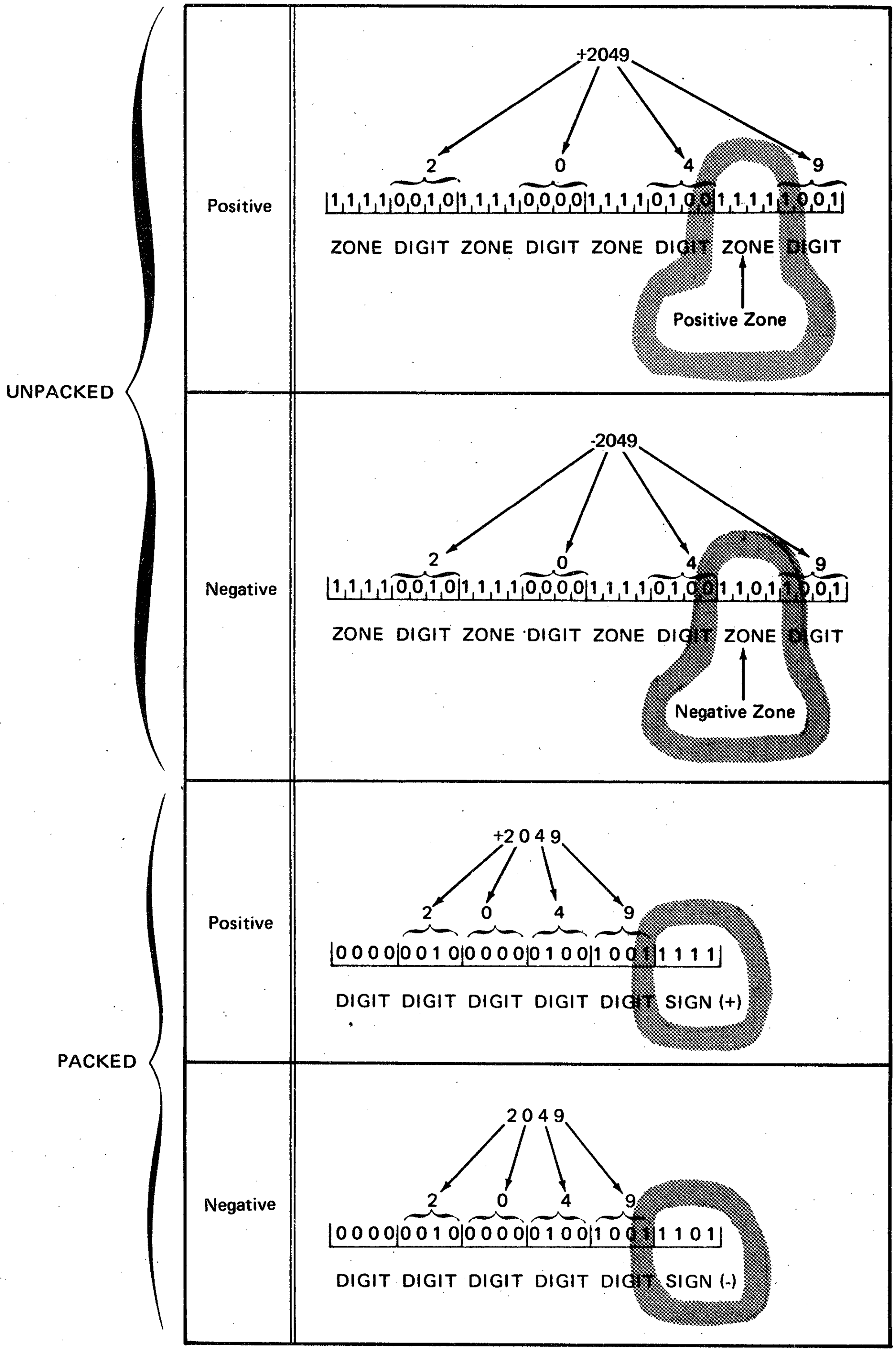
[1] Standard form. Sort accepts all four forms of the plus sign. Before sorting the file, however, the program converts all plus signs to the standard form. If a main storage dump is printed, the plus sign will always be expressed as a hexadecimal F.

Packed and Unpacked Numbers: An unpacked digit is represented by 8 bits; a packed digit is represented by 4 bits.

Unpacked Digits		
Digit	Binary Form	
0	1111	0000
1	1111	0001
2	1111	0010
3	1111	0011
4	1111	0100
5	1111	0101
6	1111	0110
7	1111	0111
8	1111	1000
9	1111	1001
	Zone Portion	Digit Portion

Packed Digits	
Digit	Binary Form
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
	Digit Portion Only

In unpacked numbers, the 4-bit sign code replaces the zone portion of the last digit in the number. In packed numbers the sign code takes up the last four bits of the number.



Columns 9-16 (Factor 1 Field Location)

Factor 1 fields identify your records. (If all of your inventory records contain an I in column 2, for example, column 2 is a factor 1 field.) The sort program identifies records you want sorted by comparing factor 1 fields with other fields in the same record, with constants, or with all or part of the program date. Columns 9 through 16 identify the locations of the factor 1 fields in the records. If there is more than one factor 1 field for the records you are describing, you must:

- Describe each field in a separate record type line.
- Put an A in column 7 of every line (except the first) to tell sort that all of the lines apply to the same record type.

Columns 9 through 12 (From) identify where the factor 1 field begins in the record. Columns 13 through 16 (To) identify where the field ends.

Length of Factor 1 Fields.

A factor 1 field can contain anywhere from 1 to 256 characters. No factor 1 field, however, can be longer than the length of the records you are working with. (For example, when you are working with 96-character records, the longest possible factor 1 field you can have is a 96-character field.) The length of factor 1 fields is also controlled by the column 8 entry.

Column 8	Maximum Factor 1 Field Length
C	256 characters[1]
Z	1 character
D	16 characters
P	8 characters[2]
U	16 characters

[1] When factor 2 is a constant, the length of the factor 1 field must not exceed 20 characters. When factor 2 is a keyword, the length of the factor 1 field must be 6 if the keyword is UDATE, and must be 2 if the keyword is UMONTH, UDAY, or UYEAR. (See *Columns 20-39* for more information.)

[2] Because the factor 1 field is packed, the field can actually represent 15 decimal digits and a sign.

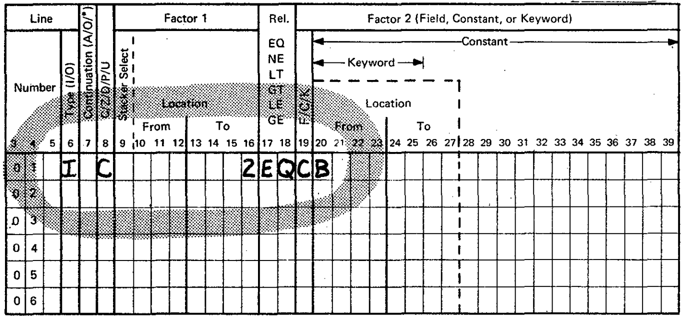
Coding Rules

1. Entries must be right-justified: the From entry must end in column 12; the To entry must end in column 16.

2. To describe factor 1 fields that are only one character long, leave columns 9 through 12 (From) blank and enter the number of the record position that contains the character in columns 13 through 16 (To).

For example, here is the entry you would make to describe the position of a factor 1 field which consists of a B in column 2.

Record Type



0 1 I C 2 EQ C B

Columns 17-18 (Relation)

The program identifies records you want to sort by comparing the factor 1 field (columns 9 through 16) with another field (columns 20 through 27) in the same record, with a constant (columns 20 through 39), or with all or part of the program date (columns 20 through 39). The other field in the same record, the constant, or the program date is called factor 2. Columns 17 and 18 tell the program what the results of the comparison must be. (If alternate collating sequence is used, both factor 1 and factor 2 are modified before the comparison is made.)

Column 17-18 Entry	Meaning
EQ[1]	Factor 1 must equal factor 2.
NE[1]	Factor 1 must not equal factor 2.
LT	Factor 1 must be less than factor 2.
GT	Factor 1 must be greater than factor 2.
LE	Factor 1 must be less than or equal to factor 2.
GE	Factor 1 must be greater than or equal to factor 2.

[1] If you want the program to compare zone portions of characters (Z in column 8), EQ and NE are the only entries you can use.

Column 19 (Field, Constant, or Keyword)

The sort program identifies records you want to sort by comparing the factor 1 field (columns 9 through 16) with another field in the same record, with a constant, or with all or part of the program date. The other field in the same record, the constant, or the program date is called factor 2. Column 19 tells the sort program whether factor 2 is another field, a constant, or a keyword that represents all or part of the program date. F in column 19 means factor 2 is a field; C means factor 2 is a constant; K means factor 2 is a keyword.

When you put an F in column 19, you use columns 20 through 27 to identify the location of the factor 2 field in the records. When you put a C in column 19, you use columns 20 through 39 for the constant. When you put a K in column 19, the keyword must begin in column 20.

Columns 20-27 (Factor 2 Field Location)

The factor 2 field must be the same length as the factor 1 field. It also must be in the same record as the factor 1 field.

Columns 20 through 27 are used to record the location of the factor 2 field. Columns 20 through 23 (From) identify the starting position of the field; columns 24 through 27 (To) identify where the field ends.

Coding Rules

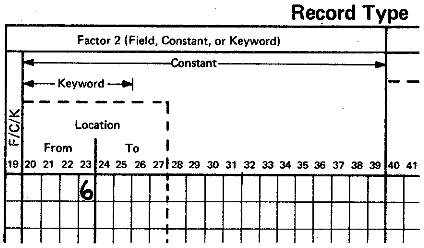
Entries must be right-justified: the From entry must end in column 23; the To entry must end in column 27.

To describe fields that are only one character long, leave columns 20 through 23 (From) blank, and enter the number of the record position that contains the character in columns 24 through 27 (To).

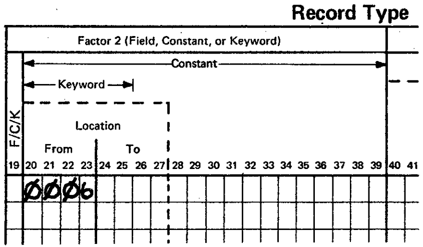
Columns 20-39 (Factor 2 Constant)

When factor 2 is a constant, you use columns 20 through 39 to write in the constant you want to use. The constant can be any arrangement of characters and blanks.

The constant must be the same length as the factor 1 field. For example, if you have a 4-position factor 1 field, your constant field must take up four positions. If your constant is the number 6, you would put the 6 in column 23, and either leave columns 20, 21, and 22 blank or fill them with zeros.



OR



If the factor 1 field contains a packed number, the length of the constant (including the sign) must be twice the length of the factor 1 field.

Alphameric Constants (Column 8 Entry is C, Z, or D)

The constant must be the same length as the factor 1 field and must always begin in column 20.

Numeric Constants (Column 8 Entry is P, U, or D)

Format: Numeric constants must be right-justified within the field length specified in factor 1 (within twice the field length if factor 1 is a packed number). For example, assume that factor 1 defines a 6-position field in the input record, and that factor 2 is the numeric constant 123. To right-justify the constant within six positions, you would have to put the constant in columns 23, 24, and 25. Leading zeros are not required. Blanks and zeros look the same to sort. In the preceding example, columns 20 through 25 could contain either 000123 or ƀƀƀ123 (with ƀ representing a blank).

Note: For character constants, a D in column 8 indicates that only the digit portion of a character will be used in the compare operations. For numeric constants, a D in column 8 indicates that the numbers are unsigned.

Signed Constants: If factor 1 is a packed number, the last character in the constant must be its sign (+ or —). If factor 1 is an unpacked number and the constant is a negative number, the last character in the constant must indicate both the numeric value of the last digit and the negative sign for the entire constant.

Coding Negative Unpacked Constants			
If Last Digit in Constant is	**Character That You Code**	**How the Number Looks Inside The Computer**	
		Zone Portion[1]	**Digit Portion**[2]
0	- (minus code)	0110	0000
1	J	1101	0001
2	K	1101	0010
3	L	1101	0011
4	M	1101	0100
5	N	1101	0101
6	O	1101	0110
7	P	1101	0111
8	Q	1101	1000
9	R	1101	1001

[1] The zone portion indicates the negative sign of the entire number.
[2] The digit portion indicates the numeric value of the last digit in the number.

For example, here are the entries you would make to sort records which have a packed negative 1 (—1) in positions 1 and 2, an unpacked negative 24 (—24) in positions 5 through 8, and an unpacked negative 10 (—10) in positions 11 through 16.

Record Type

Line					Factor 1		Rel.		Factor 2 (Field, Constant, or Keyword)				
Number	Type (I/O)	Continuation (A/O/*)	C/Z/D/P/U	Stacker Select	Location From	Location To	EQ NE LT GT LE GE	F/C/K	Constant / Keyword / Location From	Location To		Record Name	
3 4 5	6	7	8	9	10 11 12	13 14 15 16	17 18	19	20 21 22 23	24 25 26 27	28 29 30 31 32 33 34 35 36 37 38 39	40 41 42 43 44 45 46 47	48 49 50 51 52
01	I		P		1	2	EQ	C	1-			PACKED -	1
02	I	A	U		5	8	EQ	C	2M			UNPACKED	-24
03	I	A	U		11	16	EQ	C		1-		UNPACKED	-10
04													
05													
06													

Columns 20-39 (Factor 2 Keyword)

When factor 2 is a keyword, the column 8 entry must be a C. The keyword must begin in column 20; all unused columns through column 39 should be left blank. The keyword tells the sort program whether all or just part of the program date should be compared with the factor 1 field (columns 9 through 16).

Keyword[1]	Part of Program Date	Factor 1 Field Length[2]
UDATE	Entire program date	6 characters
UMONTH	Month portion of program date	2 characters
UDAY	Day portion of program date	2 characters
UYEAR	Year portion of program date	2 characters

[1] If the UDATE keyword is used, the program date must be in the same format as the date contained in the input records. If UMONTH, UDAY or UYEAR is used, the program date must be in the same format as the system date.

[2] The factor 1 field length must be 6 for UDATE, and must be 2 for UMONTH, UDAY or UYEAR.

Columns 40-72 (Comments)

Columns 40 through 72 are for your comments. If you instruct the sort program to print sequence specifications (column 27 of the header line is either blank or contains a zero), the comments will be printed along with your sequence specifications. The comments have no effect on the program's operation.

Many programmers like to use columns 40 through 47 to write the names of the records described in the record type specifications. This is why columns 40 through 47 are enclosed in dotted lines on the coding sheet.

Chapter 4. Field Specifications

Field specifications tell the sort program how to arrange and format (build) records in the work file and the output file. Possible column entries, summarized on this page and in Appendix C for quick reference, are explained in detail in this chapter.

FIELD SPECIFICATION COLUMN SUMMARY

Column Summary		
Columns	**Contents**	**Explanation**
1-2	00-99	Page number.
3-5	07x-14x	Line number. You can leave column 5 (x) blank, or enter any value to keep the specifications in ascending order.
▒ 6	F	Field specification line.
▒ 7	N	Normal control field.
	O	Opposite control field.
	F	Forced control field.
	D	Data field.
	S	Summary data field.
	*	Comment line.
▒ 8	C	Use both zone and digit portions of characters in the field.
	Z	Use only zone portion of 1-character field.
	D	Use only digit portions of characters in the field.
	P	Signed packed decimal data.
	U	Signed unpacked decimal data.
	V	Force a data character into the data field.
▒ 9-12	1-4096	Start position of a field in the record (can be blank if the field is one character long).
▒ 13-16	1-4096	End position of a field in the record.
▨ 17	Any character	Forced control fields; the character you want the program to change. Also used for summary overflow indicator fields; the character to be used for the overflow indicator.
▨ 18	Any character	Forced control fields; the character you want the program to substitute. Also used for summary overflow indicator fields; the character that the overflow field is initialized to.
▨ 19	Blank	Forced control field line is not a continuation of the preceding line.
	Any character other than blank	Forced control field line is a continuation of the preceding line.
20-22	1-256	Overflow field length entry for summary tagalong sort only.
23-39	Blank	Reserved for system use.
40-72	Any characters	Comments.

▒ Columns that must be filled in for all sort jobs.

▨ Columns that must be filled in when forced control fields are being used.

COLUMN DESCRIPTIONS

Columns 1-2 (Page Number) and 3-5 (Line Number)

Page number (columns 1 and 2) and line number (columns 3 through 5) form a 5-digit sequence number. As the sort program reads sequence specifications, it checks the sequence numbers to make sure they are not in descending order. If the numbers are in descending order (if page 02 specifications come before page 01 specifications, for example) and specifications are being issued, the sort program places a warning (the letter S) next to the line. (The S stands for sequence error.) After issuing a warning, the sort program continues reading the rest of the specification lines, then issues a message and waits for further instructions from the operator. The operator can either continue or end the job.

Because page number applies to all lines on a page, columns 1 and 2 appear only once, in the upper right corner of the page. Number the pages in ascending order.

Here is the recommended procedure for field description line numbers. Field line numbers are 07 through 14. The numbers in columns 3 and 4 are preprinted on the coding sheet. You can leave column 5 blank or enter any value to keep your specifications in ascending order. If you have more than eight field lines, use another coding sheet and start at line 07.

Use column 5 when you want to insert a specification line without renumbering the other lines. For example, to insert a specification line between lines 01070 and 01080, you can number it 01075, code it, and then continue to fill out the rest of the sheet.

Be sure any lines that are out of sequence are clearly marked. You can do this by writing a note in the margin of the page with an arrow pointing to where the insert belongs. *(Sample Job 3* in Chapter 7, *Sample Sort Jobs.)*

Column 6 (Line Type)

Column 6 contains a preprinted F, identifying the line as a field line. For addrout sorts (SORTA), field lines describe control fields that the sort program uses to sort record addresses. For tagalong sorts (SORTR) and summary tagalong sorts (SORTRS), field lines describe the fields that the sort program uses to create the records in the sorted output file. The fields can be either control fields (used to sort the records) or actual data fields. In addition, for summary tagalong sorts, field lines describe the fields that the program summarizes (adds together).

Column 7 (Field Type or Comments)

Your column 7 entry tells the sort program whether you are describing a control field, a data field, or a comment line. If you describe a control field, the column 7 entry indicates how the field is to be used.

See *Columns 7-8* to find the ways in which you can combine this entry with the column 8 entry.

Column 7 Entry	Tells Program
N	This is a normal control field. Sort this field so that the data from the field is in the sequence specified in column 18 of the header line.
O	This is an opposite control field. Sort this field so that the data from the field is in the sequence opposite that specified in column 18 of the header line.
F	This is a forced control field. Change the control field according to the entries in columns 9 through 19.
D	This is a data field.[1]
S	This is a summary data field.[2]
*	This is a comment line.

[1] Use this entry for tagalong (SORTR, SORTRS) sorts only. (If you use a D entry during an addrout sort, the line will be treated like a comment line.)

[2] Use this entry for summary tagalong (SORTRS) only. (If you use an S entry during an addrout sort, the line will be treated like a comment line. If you use an S entry during a tagalong sort, the field will be treated as a normal data field.)

Note: The first control field defined on a form for a record type is the major control field; subsequent control fields for that record type are minor control fields.

Data Fields (D in Column 7)

Data fields apply to tagalong (SORTR, SORTRS) sort jobs only. They are fields you want the sort program to include in the sorted records, but which you do not want the sort program to use in sorting the records. Within each include set (include and field lines), control field lines must be placed before data field lines.

When your file has more than one type of record:

- The number of data fields does not have to be the same for all record types.

- The total lengths of all of the data fields do not have to be the same for all record types. Sort places blanks to the right of shorter data fields so that all total lengths are equal.

Note: For a summary tagalong sort (SORTRS), you cannot determine which input record's data field will appear in the summary record.

Comment Lines (in Column 7)*

Comment lines help document the program. They do not affect program operation in any way. You can code comment lines anywhere in the sequence specifications; however, comments will be printed only if column 27 of the header line contains a zero or a blank.

Control Fields (N, O, or F in Column 7)

When your file has more than one type of record:

- The number of control fields does not have to be the same for all record types.

- The total lengths of the control fields do not have to be the same for all record types.

When records with duplicate control fields (records with all control fields equal) are sorted, their order as output is unpredictable.

Normal and Opposite Control Fields (N or O in Column 7): These control fields are fields in your input records that the sort program uses to select records for sorting. However, you can define 1-character control fields that are not in the records by using an unconditional force (see *Forced Control Fields*).

Forced Control Fields (F in Column 7): There are three types of forced control fields:

- Conditional

- Force-all

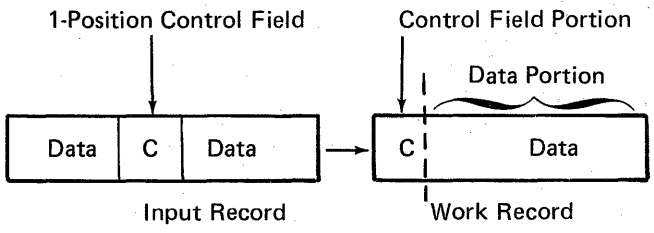
- Unconditional

Forced control fields affect the work and output records only. (It is important to remember that the sort program never changes your input records.)

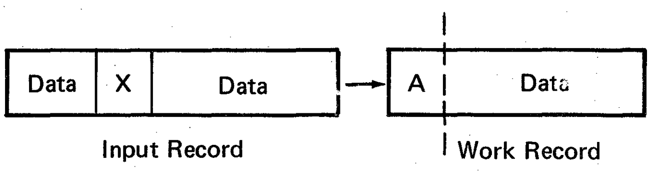
See *Column 17, Column 18,* and *Column 19* for information on how to fill out the field specifications when you are using forced control fields.

A conditional force occurs only if a control field in the input record contains a particular entry. Suppose, for example, that you want to sort a file of records, each of which has a 1-position control field. If the character in the control field is an X, you want to replace it with an A before you sort the records. To do this, you would use a conditional force. Your conditional force field would tell sort:

- Build a work record from the input record.



- If the control field contains an X, change it to an A.

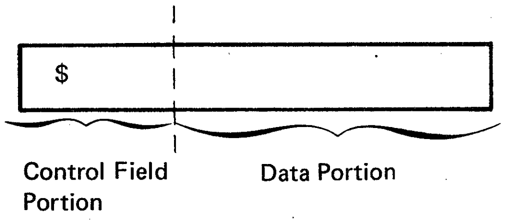


Force-all is a special type of conditional force. Force-all occurs only when a control field in an input record does *not* contain a particular entry. Suppose, for example, that you want to sort a file of records, each of which has a 1-position control field. If the character in the control field is *not* a C, F, or $, you want to put an X in the field before you sort the records. To do this, you would use a force-all line to force the X into the control field.

A force-all line follows a series of conditional force lines. For example, you may want to tell sort:

- If the control field contains a C, replace it with a 1.
- If the control field contains an F, replace it with a 2.
- If the control field contains a $, replace it with a 3.
- If the control field does *not* contain a C, an F, or a $, put an X in the control field. (You are forcing out all other possible entries by using a force-all line.)

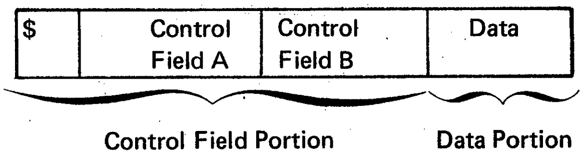
An unconditional force does not depend on entries in the input records. If, for example, you want to put a dollar sign ($) in the first position of every output record, you would use an unconditional force. Your unconditional forced control field would tell sort to put a dollar sign in the first available control field position of the work record.



Any other control fields will go after the dollar sign. Assume the input records are in this format:

Data	Control Field A	Data	Control Field B	Data

The work record will then look like this:



See *Column 19* for actual coding examples of forced control fields.

Summary Data Fields (S in Column 7)

An S in column 7 of the field specifications defines a summary data field. You can define summary fields for all three types of sort jobs, but the fields will be summarized (added together) only in summary sorts. In tagalong sorts (SORTR), summary fields are treated as normal data fields. In addrout sorts (SORTA), summary fields are treated as comments.

In a summary sort, summary data fields in the output records for all record types must be in the same position. These data fields are summarized for all record types, even if all record types are not defined as containing summary fields.

The output fields do not have to be in the same position as they were in the input records. See messages SORT-7278 and SORT-7280 in Appendix E, *Sort Programmer Messages.*

The maximum number of fields that can be summarized is 24 for each record type in a sort job.

Column 8 (C/Z/D/P/U/V)

Your column 8 entry indicates what portion of the input record characters you want the sort program to use in building and sorting the work records. The column 8 entry is critical in assuring that the sort produces the results you intend. See *Columns 7-8* to learn how you can combine this column 8 entry with the column 7 entry.

Column 8 Entry	Character Portion Used	Maximum Field Length
C	Use both zone and digit portions of the characters	256 characters
Z	Use only the zone portion of the character	1 character
D	Use only the digit portions of the characters	16 characters
P	The characters are a signed, packed decimal number	8 characters or 15 digits and 9 sign
U	The characters are a signed, unpacked decimal number	16 characters
V	Force a data character into the data field	1 character

Suppose, for example, you have a 1-character control field in your input records which can be either an *, 1, 2, or 3. The zone and digit portion of each character is:

Character	Zone	Digit
*	0101	1100
1	1111	0001
2	1111	0010
3	1111	0011

If you want the records sorted into ascending order by using the digit portion of the control field characters (by putting a D in column 8), they will be in this order:

1
2
3
*

If you want the records sorted into ascending order using both the zone and digit portions (by putting a C in column 8), they will be in this order:

*
1
2
3

Suppose you placed a Z in column 8 and wanted the records sorted into ascending order. You can then be sure that the records with an * control field will precede the records with a 1, 2, or 3 control field. Since 1, 2, and 3 have identical zone portions, records with any of these numbers as a control field will not be in any special order after the sort.

If you want to force characters into your data field, place a V in column 8 and, in column 18 specify the character to be forced. That character will be placed in the first available data field position of the work record.

As you can see, your column 8 entry can drastically affect your sorted file.

Do not confuse this column 8 entry with the column 8 entry on the record type specifications. Column 8 of the field specifications tells the sort program what portion of a character to use to sort the records. The column 8 entry on the record type specifications helps select which records you want to be sorted.

Opposite Control Fields

If you want to sort records so that some control fields are in ascending order and other control fields are in descending order, use opposite control fields. An opposite control field is sorted in ascending order (if you specify descending order on the header line), or in descending order (if you specify ascending order on the header line).

If your file contains different record types, all of which have an opposite control field in the same record position, your column 8 entries for these control fields must be one of the following:

- All Ds
- All Cs
- All Zs
- Any combination of Cs and Zs

With any other combination of entries (for example, Ds and Cs), you will not be able to predict the results of the sort.

When you use opposite control fields, sort changes them in building the work record. Therefore, you usually drop this meaningless control field information (by coding an X in column 28 on the header line) for tagalong or summary sorts. If the opposite control fields are all Ds, you do not need to drop the control field. If you wish to retain the original control field data in the output record, repeat the information as a data field.

Packed or Unpacked Control Fields (Normal or Opposite)

If you specify packed or unpacked control fields, the sort program changes the control fields while building the work record. Therefore, you may drop the control field information by coding an X in column 28 on the header line. If you wish to retain the original field data in the output record, repeat the information as a data field.

Forced Control Fields

Here are a few rules to remember when you are using forced control fields:

- You can have only one character in your forced control field.
- You can force only entire characters (zone and digit) into your control field.
- You can indicate either a conditional or an unconditional force.
- A force-all line must be preceded by a conditional force line.
- You define a forced control field by placing an F in column 7 of the field specifications.

See *Columns 9-16, Column 17, Column 18,* and *Column 19* for further information on how to complete the field specifications.

Using Control Fields to Sequence the Sorted Records

The order in which you describe control fields in the field specification lines determines the sequence of the records (tagalong sort) or the record addresses (addrout sort) in the sorted file.

Suppose each record in your file that is sorted in ascending order (A in column 18 of the header line) has a normal control field in positions 1 and 2 and an opposite control field in positions 5 through 7. Each record represents one customer's order for a separate item. The part number is in positions 1 and 2; the number of parts ordered is in positions 5 through 7. Your unsorted file might look like this:

Input Record Position

	1	2	3	4	5	6	7
0	1	7			1	2	6
1	3	4			5	1	2
2	1	7			1	2	1
3	1	5				9	6
4	3	4				1	8
5	1	7				2	0
6	3	4			2	5	0

Input Record Number **Part Number** **Number Ordered**

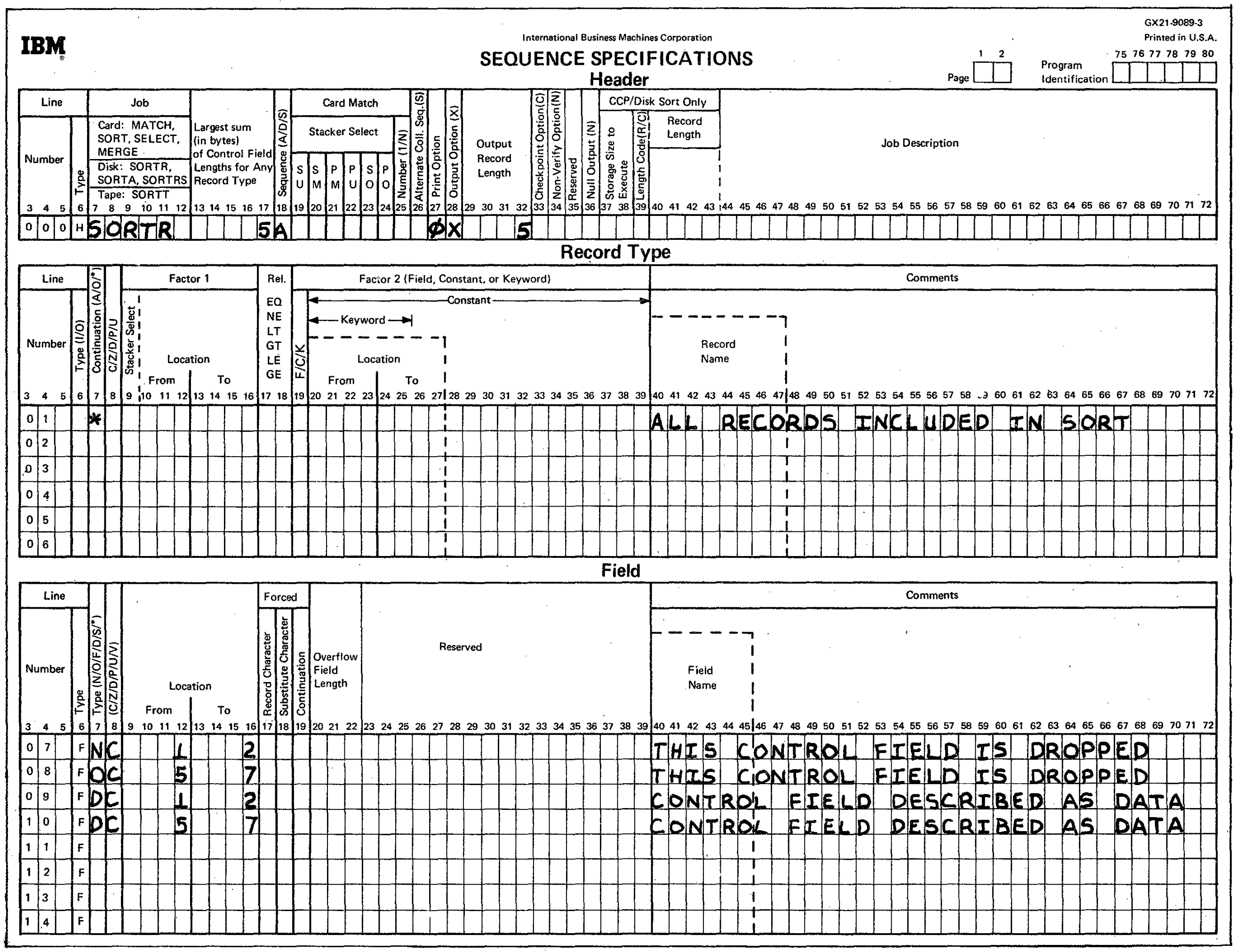
You can use the first control field to sort the records in ascending order according to the part number. You can then use the second control field to sort the number of parts ordered in descending order within each group of parts. Your sorted file would look like this:

Output Record Position

	1	2	3	4	5
0	1	5		9	6
1	1	7	1	2	6
2	1	7	1	2	1
3	1	7		2	0
4	3	4	5	1	2
5	3	4	2	5	0
6	3	4		1	8

Output Record Number **Part Number** **Number Ordered**

Include these field specifications to do the preceding sort.



IBM

International Business Machines Corporation

GX21-9089-3
Printed in U.S.A.

SEQUENCE SPECIFICATIONS

Page □□ Program Identification □□□□□□

Header

Line Number (3-5)	Type (6)	Job (7-12)	Largest sum (in bytes) of Control Field Lengths for Any Record Type (13-17)	Sequence (A/D/S) (18)	Print Option (27)	Output Option (X) (28)	Output Record Length (29-32)	Job Description (44-72)
000	H	SORTR	5	A	Ø	X	5	

Record Type

Line Number (3-5)	Type (I/O) (6)	Continuation (A/O/*) (7)	Factor 1 Location From–To	Rel.	Factor 2	Comments (40-72)
01		*				ALL RECORDS INCLUDED IN SORT
02						
03						
04						
05						
06						

Field

Line Number (3-5)	Type (6)	Type (N/O/F/D/S/*) (7)	(C/Z/D/P/U/V) (8)	Location From (9-12)	Location To (13-16)	Forced Record Character (17)	Substitute Character (18)	Continuation (19)	Overflow Field Length (20-22)	Comments (40-72)
07	F	N	C	1	2					THIS CONTROL FIELD IS DROPPED
08	F	O	C	5	7					THIS CONTROL FIELD IS DROPPED
09	F	D	C	1	2					CONTROL FIELD DESCRIBED AS DATA
10	F	D	C	5	7					CONTROL FIELD DESCRIBED AS DATA
11	F									
12	F									
13	F									
14	F									

Notice that the information occupies different positions in the input and output records. The record positions change when the control fields and data fields are moved from the input to the work and from the work to the output files. Since an opposite control field is specified, all of the control fields in the input record are dropped (columns 1 and 2 and 5 through 7). To save the control fields, specify them as data (see example).

Columns 7-8

Here are all of the possible combinations for columns 7 and 8.

Column 7	Column 8	Maximum Field Length
N or O	C	256
	Z	1
	D	16
	P	8
	U	16
F	C[1]	1
	Z[1]	1
	D[1]	1
D	C	256
	Z	1
	D	16
	P	8
	U	16
	V	1
S	C	256
	Z	1
	D	16
	P	8
	U	16
	V	1
*		Does not apply (comment line)

[1] For an unconditional force line and a force-all line, column 8 must contain a C. For a conditional force line, the column 8 entry refers to the comparison between the control field in the input record and the record character being replaced (contained in the column 17 entry) rather than to the substitute character (contained in the column 18 entry).

See column 7 and column 8 for detailed discussions of each entry.

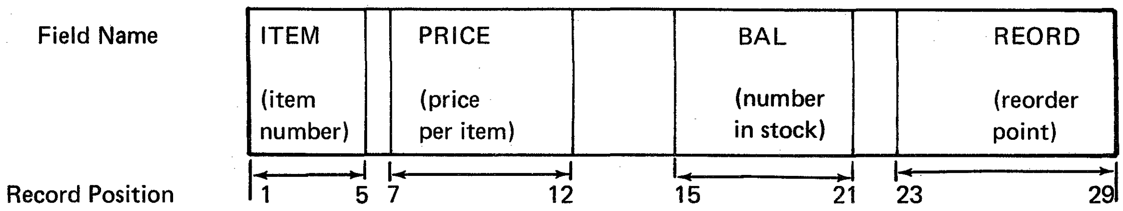
Columns 9-16 (Field Location)

Columns 9 through 16 identify the input record positions that contain the record fields. Columns 9 through 12 (From) identify the starting position of a field; columns 13 through 16 (To) identify the position in which the field ends.

The order in which you describe the control fields in the field specifications determines the order they will be in in the sorted output records.

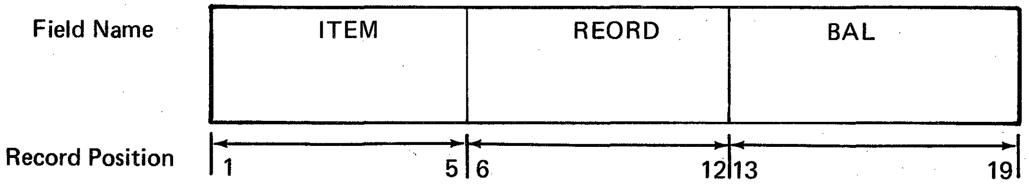
For example, suppose you have an input record that looks like this:

Input Record

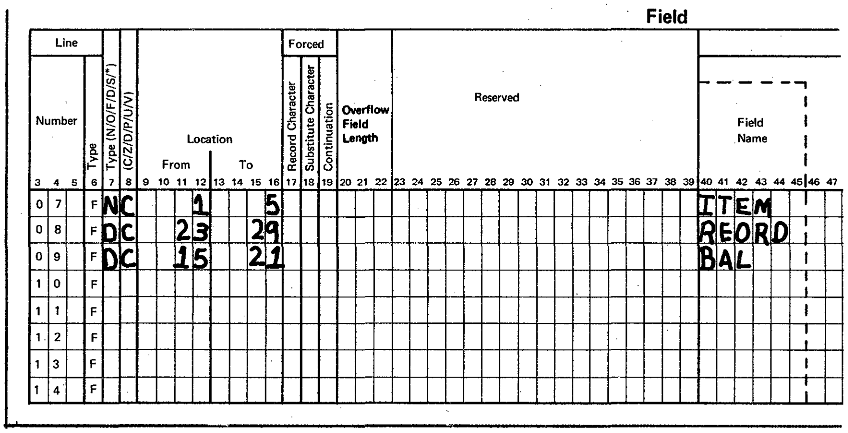


However, you want your sorted output record to look like this:

Output Record



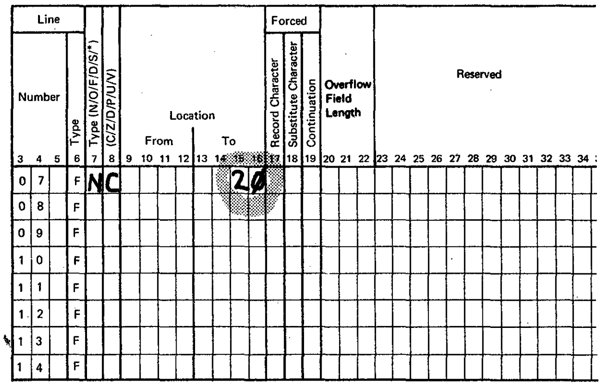
Assuming that you want to sort the records by item number, here is how you would fill out the field specifications.



Field

Line Number 3 4 5	Type 6	Type (N/O/F/D/S/*) 7	(C/Z/D/P/U/V) 8	Location From 9 10 11 12	Location To 13 14 15 16	Forced Record Character 17	Forced Substitute Character 18	Continuation 19	Overflow Field Length 20 21 22	Reserved 23–39	Field Name 40 41 42 43 44 45	46 47
0 7	F	N	C	1	5						ITEM	
0 8	F	D	C	23	29						REORD	
0 9	F	D	C	15	21						BAL	
1 0	F											
1 1	F											
1 2	F											
1 3	F											
1 4	F											

As another example, if you want the 1-character field in position 20 of your input record to be in the first position of the sorted output record, you would describe the field in the first line of your field specifications. In this example, the primary control field for the output file is determined by position 20 of the record.



Line Number (3 4 5)	Type (6)	Type (N/O/F/D/S/*) (7)	(C/Z/D/P/U/V) (8)	Location From (9 10 11 12)	Location To (13 14 15 16)	Forced Record Character (17)	Forced Substitute Character (18)	Continuation (19)	Overflow Field Length (20 21 22)	Reserved (23 24 25 26 27 28 29 30 31 32 33 34)
0 7	F	N	C		2 0					
0 8	F									
0 9	F									
1 0	F									
1 1	F									
1 2	F									
1 3	F									
1 4	F									

Field Length

The length of the field depends on the column 8 entry.

Column 8 Entry	Maximum Field Length
C	256 characters
Z	1 character
D	16 characters
P	8 characters or 15 digits and sign
U	16 characters
V	1 character

Coding Rules

1. Entries must be right-justified. The From entry must end in column 12; the To entry must end in column 16.

2. To describe fields that are only one character long, leave columns 9 through 12 (From) blank, and enter the number of the record position that contains the character in columns 13 through 16 (To).

Column 17 (Conditionally Forced Character)

See *Column 7* for a general discussion of forced control fields. You make an entry in column 17 when you want to use a conditional force. (For example, you might want to put an * in a control field only if the present entry is an A.) When you use conditional force, the input record does not change, but the work and output records will contain an * instead of an A. (See *Column 7* or *Column 18* for more information about conditional force.)

The column 17 entry tells the sort program which character in the control field (defined in columns 13 through 16) you want to replace. The program checks to see if the control field in the work record contains the character you specified in column 17. If it does, the character in column 18 replaces the control field character.

If a control field can contain any one of several characters, and you want to specify a forced character replacement for each one, you must include a column 19 entry in all of the forced field lines (except the first). See *Column 19* for instructions.

Besides being used for conditional force lines, column 17 can also be used as a summary overflow indicator field. The column 17 entry, in this case, is the character that is the overflow indicator. For more information on summary overflow indicator fields, see *Columns 20-22.*

Column 18 (Substitute Character)

See *Column 7* for general discussion of forced control fields. You make an entry in column 18 only when you are using forced control fields to sort your records. The character in column 18 either replaces the control field character you specify in column 17 or adds a new character to the control field.

Remember that the change to the control fields does not alter your input record. Also, if you want to use forced control characters you can force a 1-character field (only an entire character can be forced).

Besides being used for forced control field lines, column 18 can also be used as a summary overflow indicator field. The column 18 entry, in this case, is the character to which the overflow field is initialized. For more information on summary overflow indicator fields, see *Columns 20-22.*

Defining a Conditionally Forced Character

1. Fill in columns 1 through 6 as you would for any control field.

2. Put an F in column 7.

3. Put a C, Z, or D in column 8.

4. Leave columns 9 through 12 blank.

5. Define the position of the control field in the input record in columns 13 through 16.

6. Enter the character you want to replace in column 17.

7. Enter the character you want to replace it with in column 18. (You can use any character in column 18.)

Defining a Force-All Character

1. Fill in columns 1 through 6 as you would for any control field.

2. Put an F in column 7.

3. Put a C in column 8.

4. Leave columns 9 through 17 blank.

5. Put the character which replaces the control field in column 18.

6. Put any character in column 19. (The chracter in column 19 tells the sort program that the line refers to the same control field in the work record as the preceding line.)

If you do *not* place a force-all line after conditional force lines and the sort program does *not* find the specified characters in the control field of the input record, the sort program:

- Replaces the control field character with hexadecimal FF (if you specified ascending sequence in the header line).

- Replaces the control field character with hexadecimal 00 (if you specified descending sequence in the header line).

Defining an Unconditionally Forced Character

1. Fill in columns 1 through 6 as you would for any control field.

2. Put an F in column 7.

3. Put a C in column 8.

4. Leave columns 9 through 17 blank.

5. Put the character you are forcing in column 18.

Sample Job 4 in Chapter 7, *Sample Sort Jobs* illustrates the use of unconditionally forced characters to group record types.

Specifying an Overflow Indicator Field

1. Fill in columns 1 through 6 as you would for any control field.

2. Put an S in column 7.

3. Put a V in column 8.

4. Leave columns 9 through 16 blank.

5. Enter a character in column 17 if you do not wish to use an * as a replacement. If overflow occurs in any summary data field in the record, the initial value of the overflow field will be replaced by the character in column 17.

6. Enter a character in column 18. If none of the summary data fields overflow, the overflow field will contain the character specified in column 18.

Column 19 (Specifying Replacements for More Than One Character)

If a control field in the input record can contain any one of several characters and you want to specify forced character replacements for more than one character, use a separate line to define each character you want to replace and the forced character you want to replace it with.

Coding Rules

1. Fill in columns 1 through 6 as you would for any control field.
2. Put an F in column 7.
3. Put a C, Z, or D in column 8.
4. Leave columns 9 through 12 blank.
5. Define the control field in columns 13 through 16.
6. Enter the first character you want to replace in column 17.
7. Enter the character you want to replace it with in column 18.
8. Fill in columns 1 through 6 (of the next line) as you would for any control field.
9. Put an F in column 7.
10. Put a C, Z, or D in column 8.
11. Leave columns 9 through 12 blank.
12. Define the control field in columns 13 through 16.
13. Enter the next character you want to replace in column 17.
14. Enter the character you want to replace it with in column 18.
15. Enter any character (except blank) in column 19. Your entry tells the sort program that the line refers to the same control field in the work record as the preceding line.

Repeat steps 8 through 15 for any other characters to be replaced.

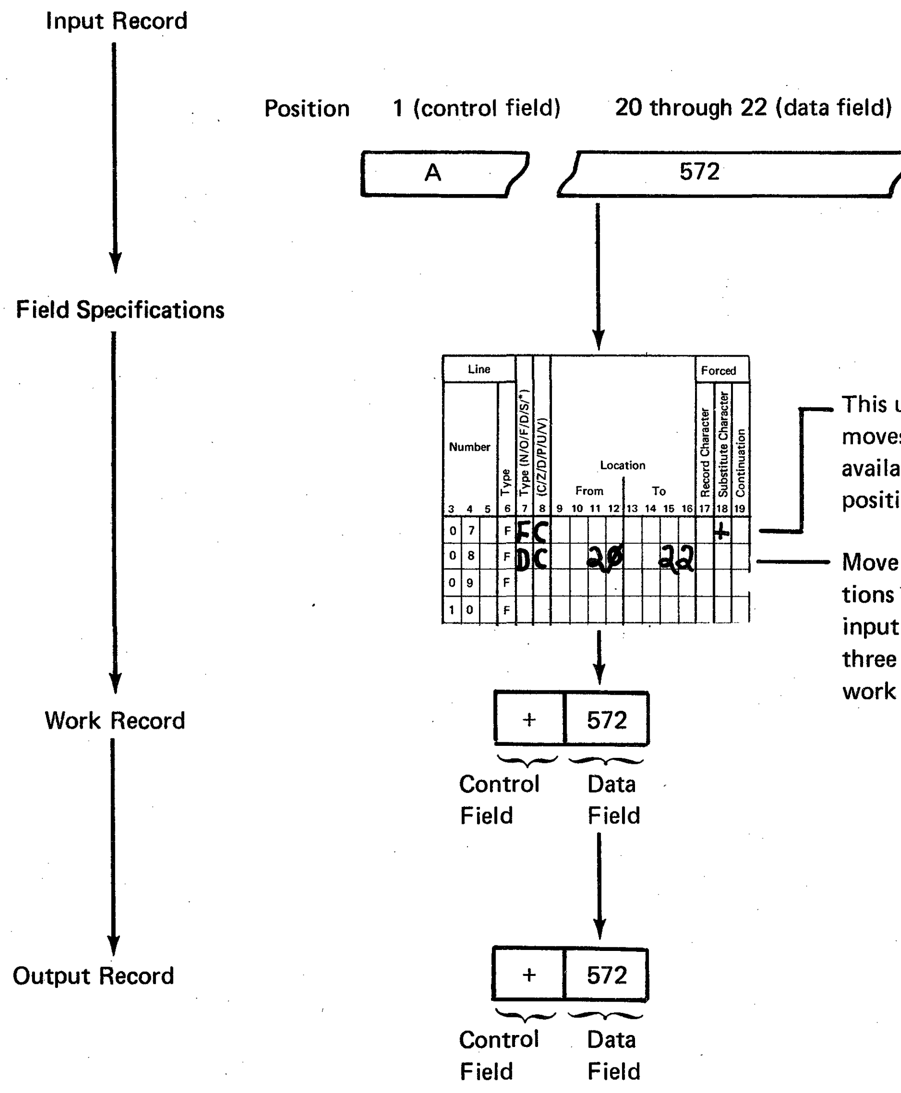
Examples of Using Forced Control Fields

You can use forced control fields to change (add to or replace) a character in a 1-position control field. The following examples represent these types of forced control characters:

- Unconditional force
- Conditional force using only forced control fields
- Conditional force using normal or opposite control fields and forced control fields
- Force-all

Unconditional Force

This example illustrates how you can unconditionally place a control field into the work and output records.



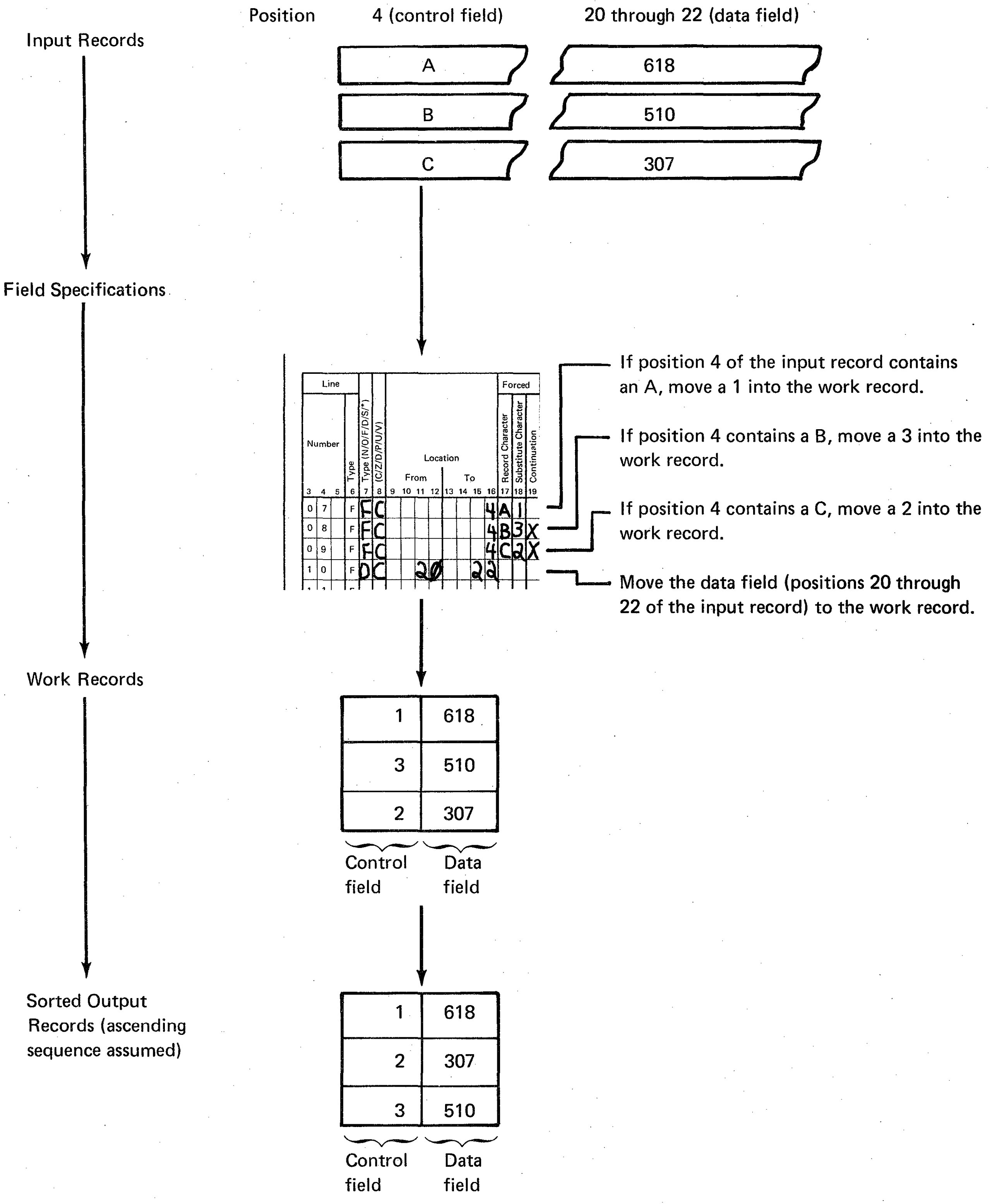
Line Number	Type	Type (N/O/F/D/S/*)	(C/Z/D/P/U/V)	Location From	Location To	Forced Record Character	Forced Substitute Character	Continuation
3 4 5	6	7	8	9 10 11 12	13 14 15 16	17	18	19
07	F	F	C				+	
08	F	D	C	2Ø	22			
09	F							
10	F							

This unconditional force moves a + into the first available work record position.

Move the data field (positions 20 through 22 of the input record) to the next three positions of the work record.

Conditional Force Using Only Forced Control Fields

This example illustrates how control field characters change according to the conditions you specify.



Line Number (3–5)	Type (6)	Type (N/O/F/D/S/*) (7)	(C/Z/D/P/U/V) (8)	Location From (9–12)	Location To (13–16)	Forced Record Character (17)	Forced Substitute Character (18)	Continuation (19)
07	F	F	C		4	A	1	
08	F	F	C		4	B	3	X
09	F	F	C		4	C	2	X
10	F	D	C	2Ø	22			

If position 4 of the input record contains an A, move a 1 into the work record.

If position 4 contains a B, move a 3 into the work record.

If position 4 contains a C, move a 2 into the work record.

Move the data field (positions 20 through 22 of the input record) to the work record.

Work Records

Control field	Data field
1	618
3	510
2	307

Sorted Output Records (ascending sequence assumed)

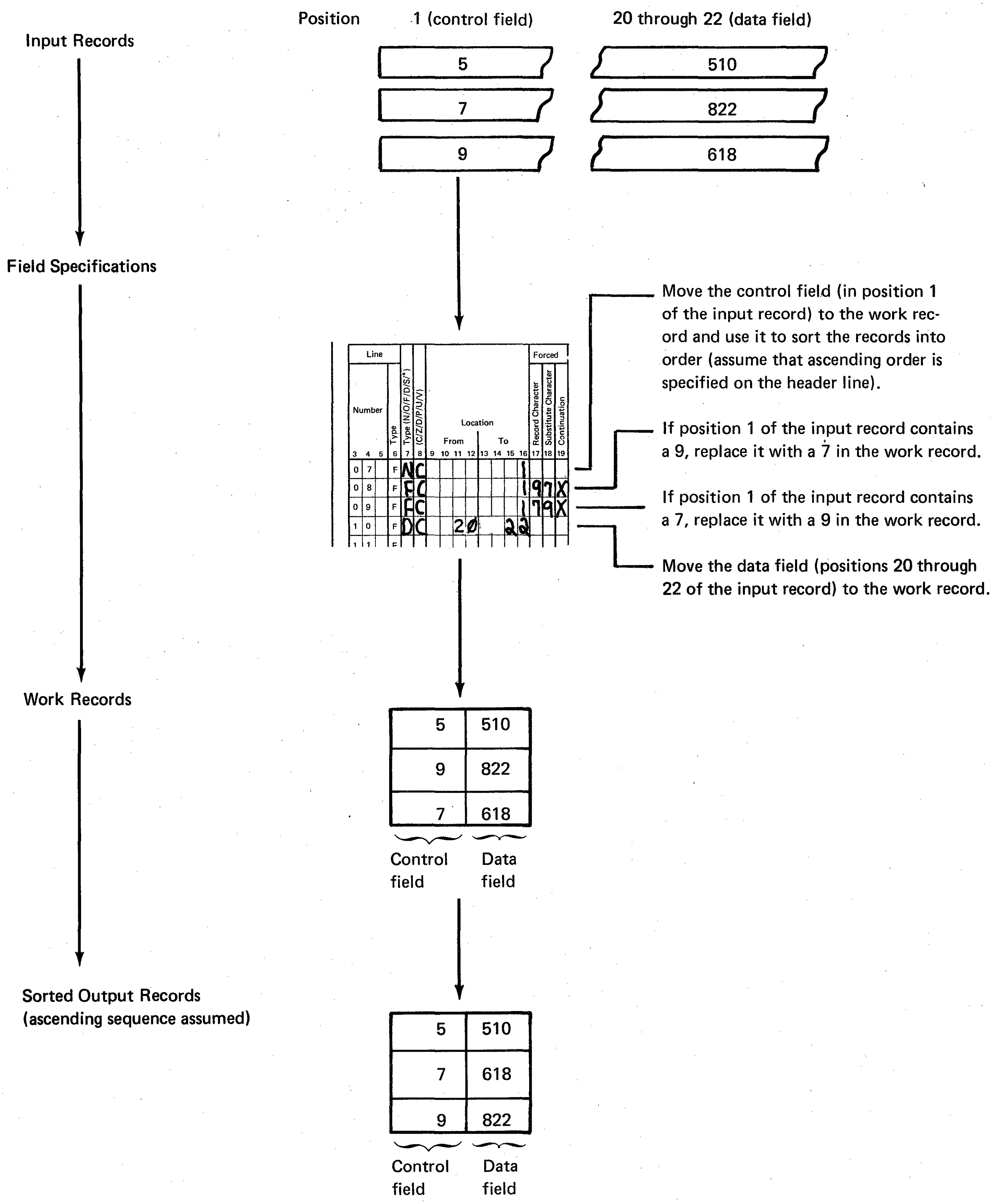
Control field	Data field
1	618
2	307
3	510

This example also shows how you can use column 19 to specify conditional replacements for more than one character. Because continuation lines are used, only one position in the work and output records is defined by the first three lines. If column 19 is left blank, each line would define a new position in the work and output records.

If you are sorting records into ascending order (A in column 18 of the header line), the sort program places hexadecimal FF into the work record before you force any characters. If you are sorting records into descending order (D in column 18 of the header line), the sort program places hexadecimal 00 into the work record before you force any characters. Therefore, if the sort program does not find an A, B, or C in position 4 of the input record, the hexadecimal 00 or FF is left unchanged in the work record.

Conditional Force Using Normal or Opposite Control Fields and Forced Control Fields

This example is similar to the previous one since the control field in the input record causes changes in the work and output records. In this case, however, the sort program first moves the control field to the work record (because the first control field is a normal control field) and then changes it if necessary. Notice that any forced character replacements are made before the records are sorted.



Line Number 3	4	5	Type 6	Type (N/O/F/D/S/*) 7	(C/Z/D/P/U/V) 8	Location From 9 10 11 12	To 13 14 15 16	Forced Record Character 17	Substitute Character 18	Continuation 19
0	7		F	N	C		1			
0	8		F	F	C		1	9	7	X
0	9		F	F	C		1	7	9	X
1	0		F	D	C	2∅	22			
1	1		F							

Move the control field (in position 1 of the input record) to the work record and use it to sort the records into order (assume that ascending order is specified on the header line).

If position 1 of the input record contains a 9, replace it with a 7 in the work record.

If position 1 of the input record contains a 7, replace it with a 9 in the work record.

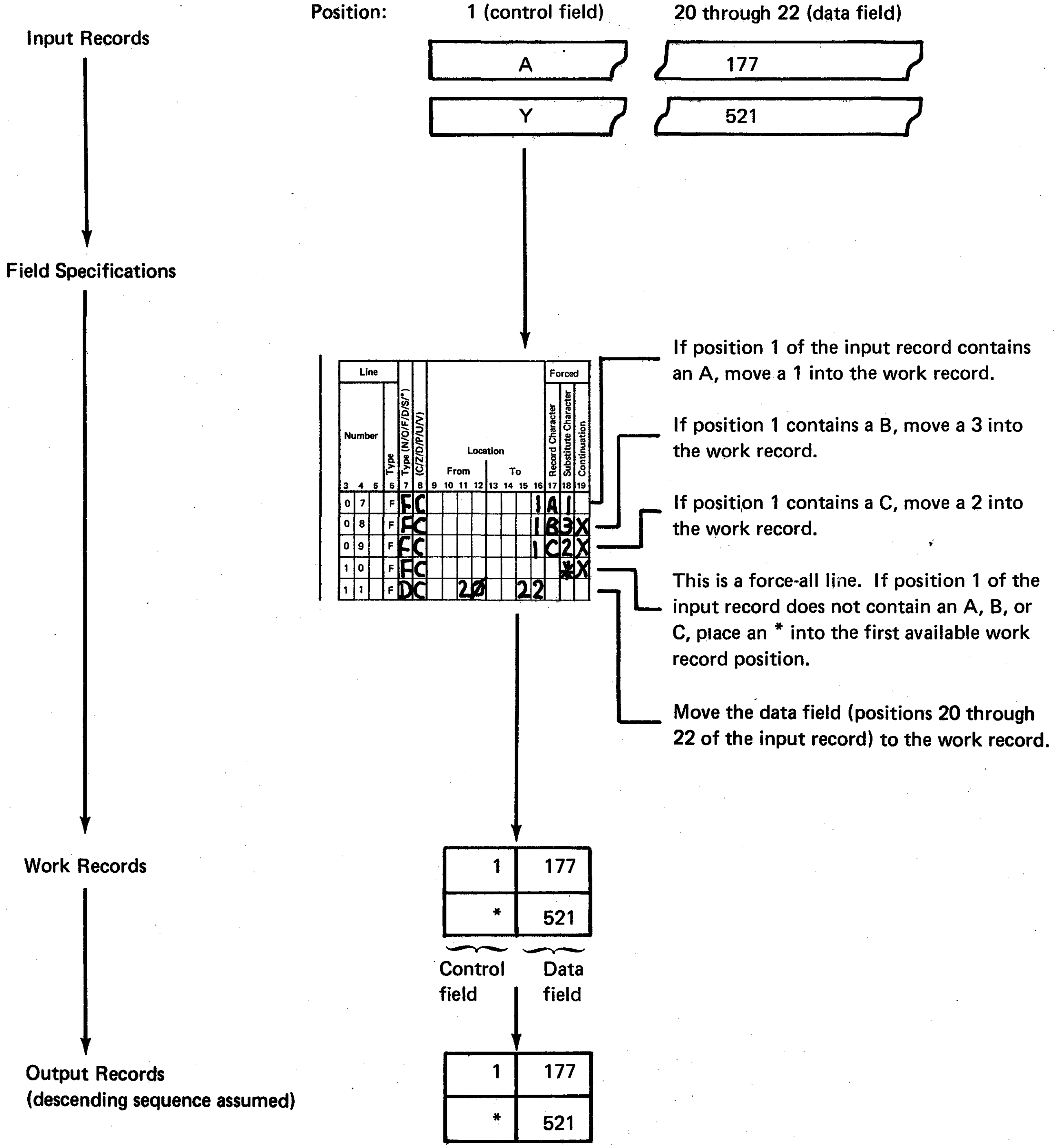
Move the data field (positions 20 through 22 of the input record) to the work record.

5	510
9	822
7	618

5	510
7	618
9	822

Force-All

Suppose your control field can contain characters other than A, B, or C. If you do not want a hexadecimal FF or 00 in the work and output records, use a force-all line.



Line														Forced		
Number			Type	Type (N/O/F/D/S/*)	(C/Z/D/P/U/V)	Location								Record Character	Substitute Character	Continuation
						From				To						
3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	7		F	F	C								1	A	1	
0	8		F	F	C								1	B	3	X
0	9		F	F	C								1	C	2	X
1	0		F	F	C										*	X
1	1		F	D	C			2	Ø			2	2			

If position 1 of the input record contains an A, move a 1 into the work record.

If position 1 contains a B, move a 3 into the work record.

If position 1 contains a C, move a 2 into the work record.

This is a force-all line. If position 1 of the input record does not contain an A, B, or C, place an * into the first available work record position.

Move the data field (positions 20 through 22 of the input record) to the work record.

Work Records

1	177
*	521

Output Records (descending sequence assumed)

1	177
*	521

Columns 20-22 (Overflow Field Length)

These columns are used only by a summary tagalong sort to eliminate the possibility of an overflow condition.

You can eliminate the possibility of an overflow condition by increasing the length of the summary data field to allow for any anticipated overflow. This can be done by coding an overflow field length entry in columns 20 through 22 of the field specifications. The overflow field length entry should be long enough to contain the sum of the summary data field length and the anticipated overflow length. The entry must be right-justified to column 22, and cannot exceed the maximum field length.

For example, if you want to summarize an unpacked field in positions 6 through 10 of your input record and you know that the output will exceed the 5-position summary field by 2 positions, specify a 7 in column 22.

Field

Line Number (3-5)	Type (6)	Type (N/O/F/D/S/*) (7)	(C/Z/D/P/U/V) (8)	Location From (9-12)	Location To (13-16)	Forced Record Character (17)	Forced Substitute Character (18)	Continuation (19)	Overflow Field Length (20-22)	Reserved (23-39)	Field Name (40-45)
07	F	S	V	6	10				7		
08	F										
09	F										
10	F										
11	F										
12	F										
13	F										
14	F										

If packed fields are summarized, columns 20 through 22 should specify the number of bytes of packed data.

For example, if you want to summarize a packed field in positions 1 through 3 of your input record and you know that the output will exceed the 3-position packed summary field (5 numbers plus sign) by 1 position, specify a 4 in column 22 (7 numbers plus sign).

Field

Line Number (3-5)	Type (6)	Type (N/O/F/D/S/*) (7)	(C/Z/D/P/U/V) (8)	Location From (9-12)	Location To (13-16)	Forced Record Character (17)	Forced Substitute Character (18)	Continuation (19)	Overflow Field Length (20-22)	Reserved (23-39)	Field Name (40-45)
07	F	S	P	1	3				4		
08	F										
09	F										
10	F										
11	F										
12	F										
13	F										
14	F										

Example of Summary Overflow Indicator Field

Sometimes the summary data field size is too small to hold the sum of the corresponding data fields. Suppose the sorted file for a mail order company looked like this:

A552	23	A552	75
Cat. No.	No. Sold	Cat. No.	No. Sold
A552	5	C200	10
Cat. No.	No. Sold	Cat. No.	No. Sold
C200	16		
Cat. No.	No. Sold		

If the company used a summary sort but specified only two positions for the summary data field, its sorted file would look like this:

A552	03	C200	26
Cat. No.	No. Sold	Cat. No.	No. Sold

The number sold for the first catalog number should be 103, but since you have only a 2-position summary field, the third digit overflowed and was lost. Since it is essential that the company knows when an overflow occurs, its programmer will define an overflow indicator field that will print a summary overflow indicator whenever an overflow occurs.

Note: You can have only one summary overflow field for each record type.

To define a summary overflow field, put an S in column 7 and a V in column 8. The overflow field will be initialized to the character specified in column 18. If overflow occurs in any summary data field in the record, the initial value of the overflow field will be replaced by the character in column 17. (If you did not put an entry in column 17, the initial value of the overflow field will be replaced by an *.) If none of the summary data fields overflow, the overflow field will contain the character specified in column 18.

Here are the entries you would make to:

- Initialize an overflow field to +.
- Replace the + with an * whenever a summary data field overflows.

F

Line Number	Type	Type (N/O/F/D/S/*)	(C/Z/D/P/U/V)	Location From	Location To	Forced Record Character	Forced Substitute Character	Continuation	Overflow Field Length	Reserved
3 4 5	6	7	8	9 10 11 12	13 14 15 16	17	18	19	20 21 22	23 24 25 26 27 28 29 30 31 32 33 34 35 36 37
0 7	F	S	V			*	+			
0 8	F									
0 9	F									
1 0	F									
1 1	F									

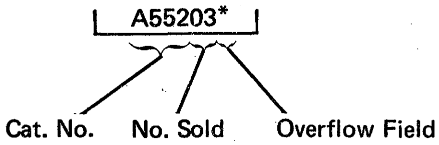
Suppose the following field specifications were used for the previous example.

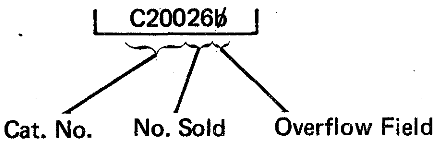
F

Line Number	Type	Type (N/O/F/D/S/*)	(C/Z/D/P/U/V)	Location From	Location To	Forced Record Character	Forced Substitute Character	Continuation	Overflow Field Length	Reserved
3 4 5	6	7	8	9 10 11 12	13 14 15 16	17	18	19	20 21 22	23 24 25 26 27 28 29 30 31 32 33 34 35 36 37
0 7	F	N	C	1	4					
0 8	F	S	C	11	12					
0 9	F	S	V			*				
1 0	F									
1 1	F									

The output records would then contain the catalog number, the number of items sold (a summary data field), and the overflow field. The overflow field is initially blank, but it will contain an * after an overflow.

The summary output records would then look like this:





You can also use the V entry in column 8 with a D entry in column 7. The character you specify in column 18 is unconditionally forced into your work record. See *Sample Job 6* in Chapter 7, *Sample Sort Jobs* for an example of using these entries.

Columns 23-39 (Reserved)

These columns are not used. Leave them blank.

Columns 40-72 (Comments)

Columns 40 through 72 are for your comments. If you instructed the sort program to print sequence specifications (column 27 of the header line is either blank or contains a zero), comments are printed along with your sequence specifications. The comments have no effect on program operation.

Columns 40 through 45 are enclosed in dotted lines on the coding sheet, because many programmers like to use these columns to write the names of fields they described in the field lines.

Chapter 5. How to Run Sort

To run any sort job, you must:

- Know the types of files you will use (input file, work file, output file).
- Supply operation control language (OCL) and sequence specifications to run your job.

OCL statements to load and execute a sort job can be entered into the system in two ways: by executing your own sort procedure stored on disk, and by entering your own sort procedure through the keyboard.

This chapter describes how you set up your files (or use existing files), and how you specify what you want done with the information that your files contain or will contain.

Note: Sorting can also be invoked through the Data File Utility (see the *IBM System/32 Utilities Program Product Reference Manual—Data File Utility,* SC21-7600.)

FILE INFORMATION

File Characteristics

Input File

The input file is an area on disk that contains records to be sorted. Any indexed, sequential, or direct file with fixed length records can be used as input to the sort program; for example, an RPG II file or a file from a previous sort job. If you use the direct access method to create a file for the sort program, you might want to test for and omit blank records because direct file areas are set to blanks before the file is created.

Work File

The work file is an area on disk that is set aside as a work area. The sort program uses this area as working storage during the sorting process. You can either specify the work file area on disk or you can let the sort program automatically allocate it.

You should usually give a work file scratch status (RETAIN-S) because the work file is created during a sort and usually has no function after the sort job is completed.

Output File

The output file contains the sorted file. If the output file exists before the sort, it cannot have permanent status (RETAIN-P) because the sort program cannot write into a permanent file. However, if the output file is built during the sort, it can have either temporary (RETAIN-T), or permanent (RETAIN-P) status. RETAIN-S should not be used since the file would be deleted at the end of the sort job.

Automatic Work File Allocation

If you omit the work file FILE statement from your sort OCL statements, the sort program automatically allocates a scratch area on the disk. This scratch area is the remaining free disk space.

File Programming Considerations

You must consider several file characteristics when setting up your files or selecting the type of sort you will use. Some basic considerations are:

	Addrout Sort (SORTA)	Tagalong Sorts (SORTR, SORTRS)
Work file size	Work file need only be big enough to contain control fields and relative record numbers.	Work file must be big enough to contain the entire file to be sorted. (See Appendix A for information on how to calculate work file size.)
Sort output	Relative record numbers only.	Control fields, data fields, and summary data fields. (See *Types of Sort: Addrout, Tagalong, and Summary Tagalong* in Chapter 1.)
Input file overlay	Output file should not overlay the input file.[1]	Output file can overlay the input file; that is, the input file can also be the output file.[2]
Sort job running time	Usually shorter than SORTR or SORTRS.	Usually longer than SORTA.

[1] If your input file is overlaid, you must have a backup copy of your input file that you can access using your addrout sort output.

[2] The space for the output file can be the same as the space for the input file. However, you should not overlay a file you want to sort unless you first make a backup copy of your input file. When you use the input file as the output file, you must specify (for your output file) the location and size.

RUNNING SORT BY USING YOUR OWN SORT PROCEDURES

File Information Needed When Using Your Own Procedures

The following text contains general file information. For detailed FILE statement parameter descriptions, see the *IBM System/32 System Control Programming Reference Manual,* GC21-7593.

If you enter your own OCL statements, you must supply sort with the following information:

About the Input File:

Name of the file you will sort. The FILE statement NAME must be INPUT. Use the LABEL keyword to give the sort program the name of your file if it is not INPUT. For example:

```
// FILE NAME-INPUT,LABEL-MASTER
```

where MASTER is the name of your input file.

About the Work File (Optional):

- Name of your work file. The FILE statement NAME must be WORK. Use the LABEL keyword to give the sort program the name of your file if it is not WORK. For example:

  ```
                                   {BLOCKS-  }
  // FILE NAME-WORK,LABEL-SORTWORK,{         }number,RETAIN-S
                                   {RECORDS- }
  ```

  where SORTWORK is the name of your work file.

- Number of blocks or records in the work file. In the work file FILE statement, the number after either BLOCKS or RECORDS indicates the total number of blocks or records in the work file. See Appendix A for information on how to calculate the number for either BLOCKS or RECORDS.

- Status of the work file. RETAIN-S (scratch at end-of-sort job) indicates work file status. The work file is usually a scratch file since you normally do not need the information in the work file after a sort job is run.

About the Output File:

- Name of the sorted file. The FILE statement NAME must be OUTPUT. Use the LABEL keyword to give the sort program the name of your file if it is not OUTPUT. For example:

  // FILE NAME-OUTPUT,LABEL-SORTMAST,$\left\{\begin{matrix}\text{BLOCKS-}\\ \text{RECORDS-}\end{matrix}\right\}$number,RETAIN-$\left\{\begin{matrix}\text{T}\\ \text{P}\end{matrix}\right\}$

  where SORTMAST is the name of your output file.

- Number of blocks or records in the output file. In the output file FILE statement, the number after either BLOCKS or RECORDS indicates that total number of blocks or records in the output file. See Appendix A for information on how to calculate the number for either BLOCKS or RECORDS.

- Status of the output file. RETAIN-T (temporary) or RETAIN-P (permanent) indicates output file status. A file built during the sort job can have either of these parameters; however, an output file that exists before the sort job is run cannot have permanent status because you cannot write into an existing permanent file. RETAIN-S should not be used since the file would be deleted at the end of the sort job.

  Note: If the work or output file you want to use is an existing temporary file (RETAIN-T), you must supply the location where the file begins and its originally allocated size.

Source Member Information Needed When Using Your Own Procedures

A source member can contain the sort sequence specifications for a sort job.

The format of the SOURCE statement is:

```
// SOURCE source member name
```

For a detailed description of source members and for information on how source members can be created using the $MAINT system utility program, see the *IBM System/32 System Control Programming Reference Manual,* GC21-7593.

Source members and procedures can also be created and updated by the Source Entry Utility (SEU). For information on how to use SEU, see the *IBM System/32 Utilities Program Product Reference Manual–Source Entry Utility,* SC21-7605.

Executing Sort Procedures Stored on Disk

If you use a sort procedure that is on disk, sort sequence specifications can either be contained in a source member or can be contained within the procedure.

To execute the procedure, enter either:

```
name
```

or

```
// INCLUDE name
```

where name is the name of the sort procedure.

If the sort sequence specifications are contained in a source member, your procedure includes:

```
// LOAD #GSORT
// FILE NAME-INPUT,...
// FILE NAME-WORK,... (optional)
// FILE NAME-OUTPUT,...
// RUN
// SOURCE source member name
```

where the referenced source member includes:

```
(header specification)
(record type specifications)
(field specifications)
// END
```

The sort sequence specifications can be included in your sort procedure which then includes:

```
// LOAD #GSORT
// FILE NAME-INPUT
// FILE NAME-WORK,... (optional)
// FILE NAME-OUTPUT,...
// RUN
(header specification)
(record type specifications)
(field specifications)
// END
```

Note: The procedure must contain an END statement after the last sort sequence specification statement.

Entering Sort Procedures through a Keyboard

If you enter the OCL for a sort job through a keyboard, the sort sequence specifications can be in a source member or you can key them in with the sort procedure OCL.

When sort sequence specifications are contained in a source member, the following OCL statements are keyed:

```
// LOAD #GSORT
// FILE NAME-INPUT,...
// FILE NAME-WORK,... (optional)
// FILE NAME-OUTPUT,...
// RUN
// SOURCE source member name
```

where the referenced source member includes:

```
(header specification)
(record type specifications)
(field specifications)
// END
```

The sort sequence specifications can be keyed in with the sort procedure OCL statements as follows:

```
// LOAD #GSORT
// FILE NAME-INPUT,...
// FILE NAME-WORK,... (optional)
// FILE NAME-OUTPUT,...
// RUN
(header specification)
(record type specifications)
(field specifications)
// END
```

Note: The END statement indicates to the sort program that the last sort sequence specification statement has been entered.

Two FILE statements are required for any offline multivolume file. For multivolume file statements, refer to the *IBM System/32 System Control Programming Reference Manual*, GC21-7593. (The sort program does not support offline multivolume work files.)

Note: Offline multivolume files cannot be processed while the sort program is running as an inquiry program.

Chapter 6. Performance Considerations

The time required to run a sort job can vary greatly, and depends on the following factors:

- Number of records to be sorted
- Record size
- Number of sequence specifications
- Whether or not alternate collating sequence is being used
- Disk location of the input, work, and output files
- Use of offline multivolume files
- Whether or not automatic work file allocation is being used
- Order of records in the input file
- Use of the nonverify option

The following text describes how each factor affects the time required to run a sort job.

NUMBER OF RECORDS TO BE SORTED

The more records you want to sort, the longer the sort job takes to run. For the most efficient sort job, therefore, use the record omit option whenever possible.

RECORD SIZE

The larger the records, the longer it takes to sort a file. Therefore, if program run time is important to you, do not include unnecessary fields in your sort job. Every field you do not include decreases the record size by the length of that field. Remember, sort builds a work record for each input record included in the sort.

NUMBER OF SEQUENCE SPECIFICATIONS

The more sequence specifications you use, the less main storage space will be available for records.

- The code required by the select/build routine for any sort job uses about 100 bytes of storage.
- Each sequence specification uses about 25 bytes of storage. (Remember that comment lines use no main storage.)
- If you specify includes and/or omits (referring to packed data) for your sort job, the select/build routine will use an additional 200 bytes of storage.
- If you specify field statements (referring to packed data) for your sort job, the select/build routine will use an additional 80 bytes of storage.
- An alternate collating sequence uses an additional 375 bytes of storage.

Your sort job may run faster if you use additional sequence specifications to exclude records you do not want to sort.

ALTERNATE COLLATING SEQUENCE

Use of an alternate collating sequence increases the time it takes to run a sort job (see *Alternate Collating Sequence* in Appendix B).

DISK LOCATION OF FILES

Because the sort program must move each record you want to sort at least two times (first from the input file to the work file, then from the work file to the output file), file location affects how long it takes to run a job. The distance the disk arm must travel between the input and work files and between the work and output files can be minimized to reduce running time. The following shows how the input, work, and output files for a sort job can be placed on a disk system. Each method shown will usually run slower than the method or methods that precede it:

	Input	Work	Output	Disk	← (fastest method)
	Input	Output	Work	Disk	
	Output	Input	Work	Disk	

OFFLINE MULTIVOLUME FILES

Use of offline multivolume files (diskette files) increases the time needed to run a sort job because diskette files are copied to disk before they are used.

AUTOMATIC WORK FILE ALLOCATION

Automatic work file allocation usually increases the time needed to run a sort job because an automatically allocated work file is not arranged for a fast sort run. If you want to minimize sort running time, use a well-planned work file and the work file FILE statement rather than automatic work file allocation.

ORDER OF RECORDS IN THE INPUT FILE

The order of input file records can affect how long it takes to run a sort job. Suppose that two files have about the same number of records to be sorted. Suppose further that the two files contain identical information; however, more of the records in the first file are in the sequence you want than is true of the records in the second file. The sort program usually takes less time to sort the records in the first file than in the second file if the same sequence specifications are used to sort both files.

NONVERIFY OPTION

If an N is entered in column 34 of the sort header specification, the sort program does not verify data written on the work file. The amount of performance improvement depends on file size, number of records, and file location.

Chapter 7. Sample Sort Jobs

Each of the seven jobs in this section has:

1. An introduction explaining its purpose.

2. Filled-out sequence specification sheets.

3. Discussion of the sequence specifications.

The first six jobs use the following files and input records. (Sample Job 7 uses other files and input records, which are explained in the introduction to that job.)

Files	Records in Files	Contents of Records
Inventory File	Inventory Records	● Quantities of items in stock ● Number of transactions for each stock item
Transaction File	Issue Records	Shipments of items to customers
	Receipt Records	Purchases of more items from suppliers (vendors)
	Adjustment Records	Corrections to inventory quantities

The formats of the four types of records are shown in the following charts.

Record Formats for the Inventory File

	Record Positions	Field Names	Contents[1]	
Inventory Record	1-6	ITEM	Number of stock item to which the record applies.	**Item Information**
	7-10	CLASS	Class of the item (men's clothing, jewelry, and so on).	
	11-24	DESC	Description of the item.	
	25-27	LOC	Location of the warehouse where the item is stored.	**Stock Status and Reorder Information**
	28-29	UNIT	Unit of measure for ordering the item (by the pound, by the dozen, and so on).	
	30-34	COST	Cost of the item per unit.	
	35-39	PRICE	Price (per unit) at which the item is sold.	
	40-45	REORD	Reorder point (lowest stock quantity allowed).	
	46-50	QTY	Number of units to order when reordering.	
	51-56	BAL	Number of units in stock.	
	57		Not used.	
	58-62	ORDER	Number of units ordered, but not yet received.	
	63-68		Not used.	
	69-74	AVAIL	Number of units available (BAL + ORDER).	
	75-80	PORDER	Purchase order number.	
	81-86	PDATE	Date of purchase order.	
	87-91	VEND1	First vendor.	
	92-96	VEND2	Second vendor.	
	97-99	TRANS	Number of transactions for this period.	**Transaction Activity**
	100-105	TDATE	Date of last transaction.	
	106-111	ISSUES	Number of units issued during this period.	

[1] Any unsigned numeric field (packed or unpacked) can be treated as character information.

Record Formats for the Transaction File

	Record Positions	Field Names	Contents[1]
Issue Record	1	ID	I (identifies this record as an Issue record).
	2-7	ITEM	Number of the stock item to which record applies.
	8-12	QTY	Number of units ordered (5 dozen, 32 pounds, and so on).
	13-19	CUST	Customer number.
	20-27	INV	Invoice number.
	28-33	DATE	Date of the order.
	34	DIS	Discount information.
	35-39	PRICE	Price of the item per unit.
	40-42	CON1	The constant 001.
	43-50	COST	Total cost of items ordered.
Receipt Record	1	ID	R (identifies this record as a Receipt record).
	2-5	CLASS	Class of the item ordered (men's clothing, jewelry, and so on).
	6-11	ITEM	Number of the item ordered
	12-16	PRICE	Price of the item per unit (by the dozen, by the pound, and so on).
	17-22	PORDER	Purchase order number.
	23-27	QTY	Number of units ordered.
Adjustment Record	1	ID	A (identifies this record as an Adjustment record).
	2-7	ITEM	Number of the stock item to which record applies.
	8	CODE	Adjustment code (identifies reason for the adjustment).
	9-13	QTY	Number of units (dozens, pounds, and so on) being added or subtracted.
	14-17	OK	Authorization for the adjustment.
	18-23	DATE	Date of the adjustment.

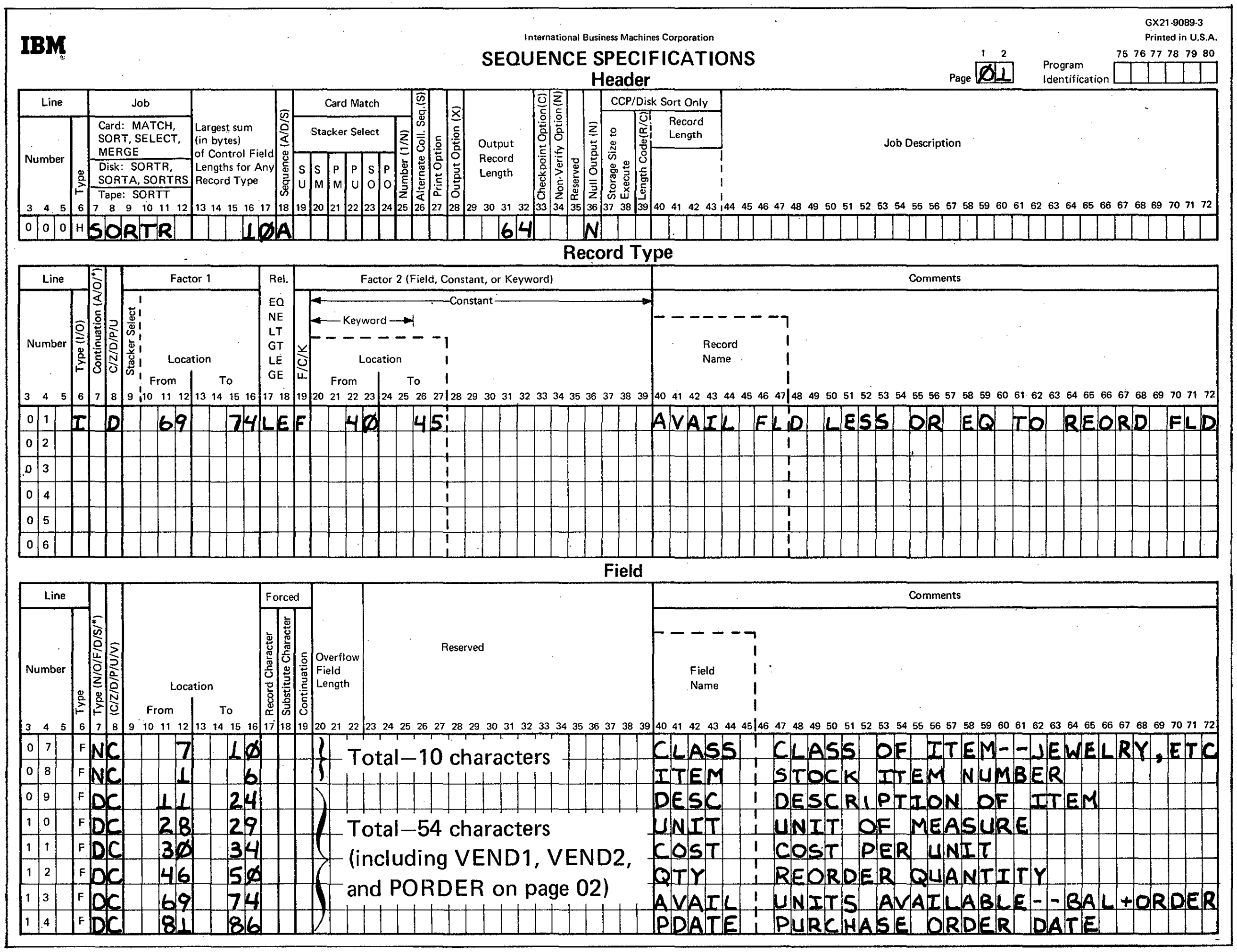
[1] Any unsigned numeric field (packed or unpacked) can be treated as character information.

SAMPLE JOB 1: PRODUCING A FILE OF INVENTORY INFORMATION FOR REORDERING STOCK ITEMS

- The program selects inventory for stock items that need to be reordered. These are items for which the quantity on hand (AVAIL field) is less than or equal to the reorder point (REORD field).

- Output records are to contain information needed to reorder the items. The information is in these fields: CLASS, ITEM, DESC, UNIT, COST, QTY, AVAIL, PDATE, VEND1, VEND2, and PORDER.

- Output records are to be in ascending order by class (CLASS field) and by item numbers (ITEM field) within each class.



IBM

International Business Machines Corporation

SEQUENCE SPECIFICATIONS

GX21-9089-3 Printed in U.S.A.

Page Ø1 Program Identification

Header

Line Number	Type	Job (Card: MATCH, SORT, SELECT, MERGE; Disk: SORTR, SORTA, SORTRS; Tape: SORTT)	Largest sum (in bytes) of Control Field Lengths for Any Record Type	Sequence (A/D/S)	Output Record Length	Null Output (N)	Job Description
000	H	SORTR	1Ø	A	64	N	

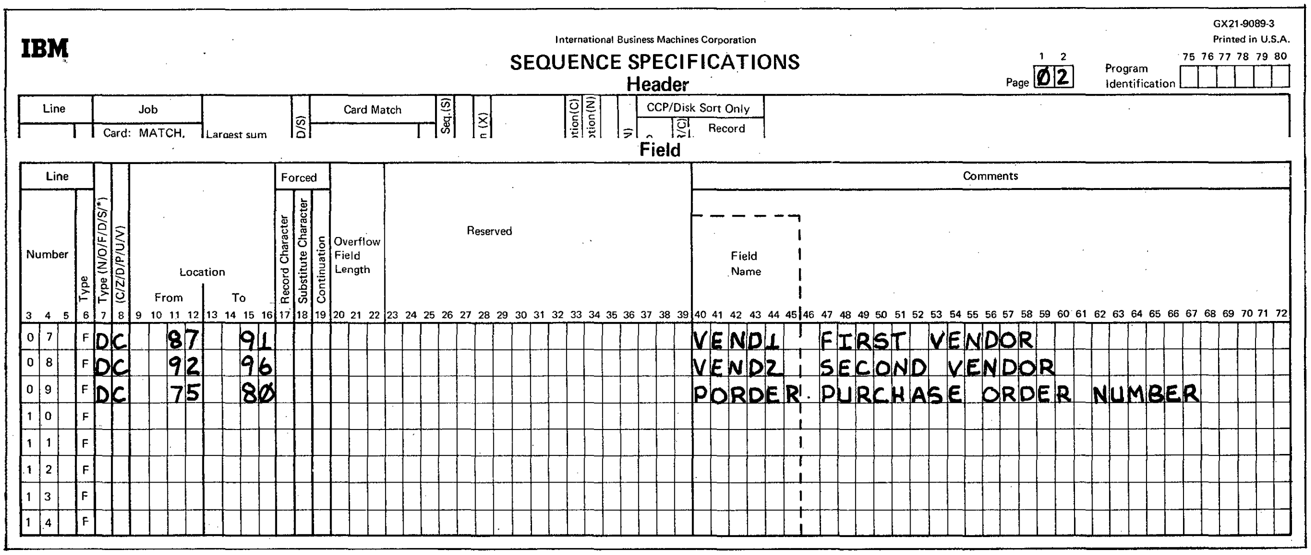
Record Type

Line Number	Type (I/O)	Continuation (A/O/*)	C/Z/D/P/U	Stacker Select	Factor 1 Location From	To	Rel. (EQ NE LT GT LE GE)	F/C/K	Factor 2 Location From	To	Comments (Record Name)
01	I		D		69	74	LE	F	4Ø	45	AVAIL FLD LESS OR EQ TO REORD FLD
02											
03											
04											
05											
06											

Field

Line Number	Type	Type (N/O/F/D/S/*)	(C/Z/D/P/U/V)	Location From	To	Record Character	Substitute Character	Continuation	Overflow Field Length / Reserved	Field Name	Comments
07	F	N	C	7	1Ø				Total—10 characters	CLASS	CLASS OF ITEM--JEWELRY, ETC
08	F	N	C	1	6					ITEM	STOCK ITEM NUMBER
09	F	D	C	11	24				Total—54 characters (including VEND1, VEND2, and PORDER on page 02)	DESC	DESCRIPTION OF ITEM
10	F	D	C	28	29					UNIT	UNIT OF MEASURE
11	F	D	C	3Ø	34					COST	COST PER UNIT
12	F	D	C	46	5Ø					QTY	REORDER QUANTITY
13	F	D	C	69	74					AVAIL	UNITS AVAILABLE--BAL+ORDER
14	F	D	C	81	86					PDATE	PURCHASE ORDER DATE

Because there is not enough space on the first coding sheet for the last three fields, they are coded on a second sheet. To show the proper sequence for the last three fields, put a 02 in columns 1 and 2 of the second coding sheet.



IBM

International Business Machines Corporation

SEQUENCE SPECIFICATIONS

GX21-9089-3 Printed in U.S.A.

Page Ø2 Program Identification

Header

Field

Line Number	Type	Type (N/O/F/D/S/*)	(C/Z/D/P/U/V)	Location From	To	Record Character	Substitute Character	Continuation	Overflow Field Length	Reserved	Field Name	Comments
07	F	D	C	87	91						VEND1	FIRST VENDOR
08	F	D	C	92	96						VEND2	SECOND VENDOR
09	F	D	C	75	8Ø						PORDER	PURCHASE ORDER NUMBER
10	F											
11	F											
12	F											
13	F											
14	F											

Header Specification

Columns 1-2

01 identifies this sheet as the first sequence specifications sheet.

Column 6

H identifies the line as a header line.

Columns 7-12

SORTR identifies the job as a tagalong sort.

Columns 13-17

The control fields used to sort records in this job are CLASS and ITEM. The total of their lengths is 10, which is the entry for columns 13 through 17.

Column 18

The A stands for ascending order. Output records are to be sorted into ascending order by CLASS and ITEM fields.

Columns 29-32

The output records contain eleven fields, including two control fields (CLASS and ITEM) and nine data fields. Control fields are not being dropped (column 28 is blank). Therefore, the length of the output record is the total length (64) of all eleven fields described in field specifications.

Column 36

N in column 36 indicates that the sort program will create an empty output file if no records are found in the input file or if no records are selected for sorting, and will *not* issue message SORT-7724 NO INPUT RECORDS INCLUDED.

Record Type Specifications

Column 6

I identifies the line as an include line. The input records described by the line are used by the program. All other input records are omitted.

Column 8

Both factor 1 and factor 2 fields are numeric positive fields. The D in column 8 causes zeros and blanks to be considered equal if leading zeros were not recorded.

Columns 9-16, 17-18, and 20-27

Inventory records for which the number of units available (AVAIL) is less than or equal to the reorder point (REORD) are selected:

- Columns 9 through 16 identify the location of the AVAIL field (record positions 69 through 74).
- Columns 20 through 27 identify the location of the REORD field (record positions 40 through 45).
- Columns 17 and 18 define the comparison made between the two fields.

Column 19

An F in column 19 indicates that factor 1 is compared to the field located by the entries in columns 20 through 27.

Columns 40 through 72

Columns 40 through 72 are comment columns.

Field Specifications

Column 6

F identifies the lines as field specification lines.

Column 7

The first two fields described (CLASS and ITEM) are normal control fields. The rest are data fields. All are included in the output record (column 28 of the header line is blank).

The records are sorted into ascending order by the CLASS field. Within each class, the item numbers (ITEM field) are in ascending order.

CLASS Sequence	ITEM Sequence	Sequence of Records in Sorted Output File
Lowest Class No.	Lowest item no.	Lowest class no. - Lowest item no.
	Next higher item no.	Lowest class no. - Next higher item no.
	Third higher item no.	Lowest class no. - Third higher item no.
	.	.
	.	.
	.	.
	Highest item no.	Lowest class no. - Highest item no.
Each Higher Class No.	Lowest item no.	Class no. - Lowest item no.
	Next higher item no.	Class no. - Next higher item no.
	Third higher item no.	Class no. - Third higher item no.
	.	.
	.	.
	.	.
	Highest item no.	Class no. - Highest item no.
Highest Class No.	Lowest item no.	Highest class no. - Lowest item no
	Next higher item no.	Highest class no. - Next higher item no.
	Third higher item no.	Highest class no. - Third higher item no.
	.	.
	.	.
	.	.
	Highest item no.	Highest class no. - Highest item no.

Column 8

Zone and digit portions of all characters in all fields are used as they appear in the input records.

Columns 9-16

Columns 9 through 16 identify the locations of fields in the input records.

Columns 40-72

Columns 40 through 72 are comment columns.

SAMPLE JOB 2: PRODUCING A RELATIVE RECORD ADDRESS FILE FOR AN ENTIRE INVENTORY FILE

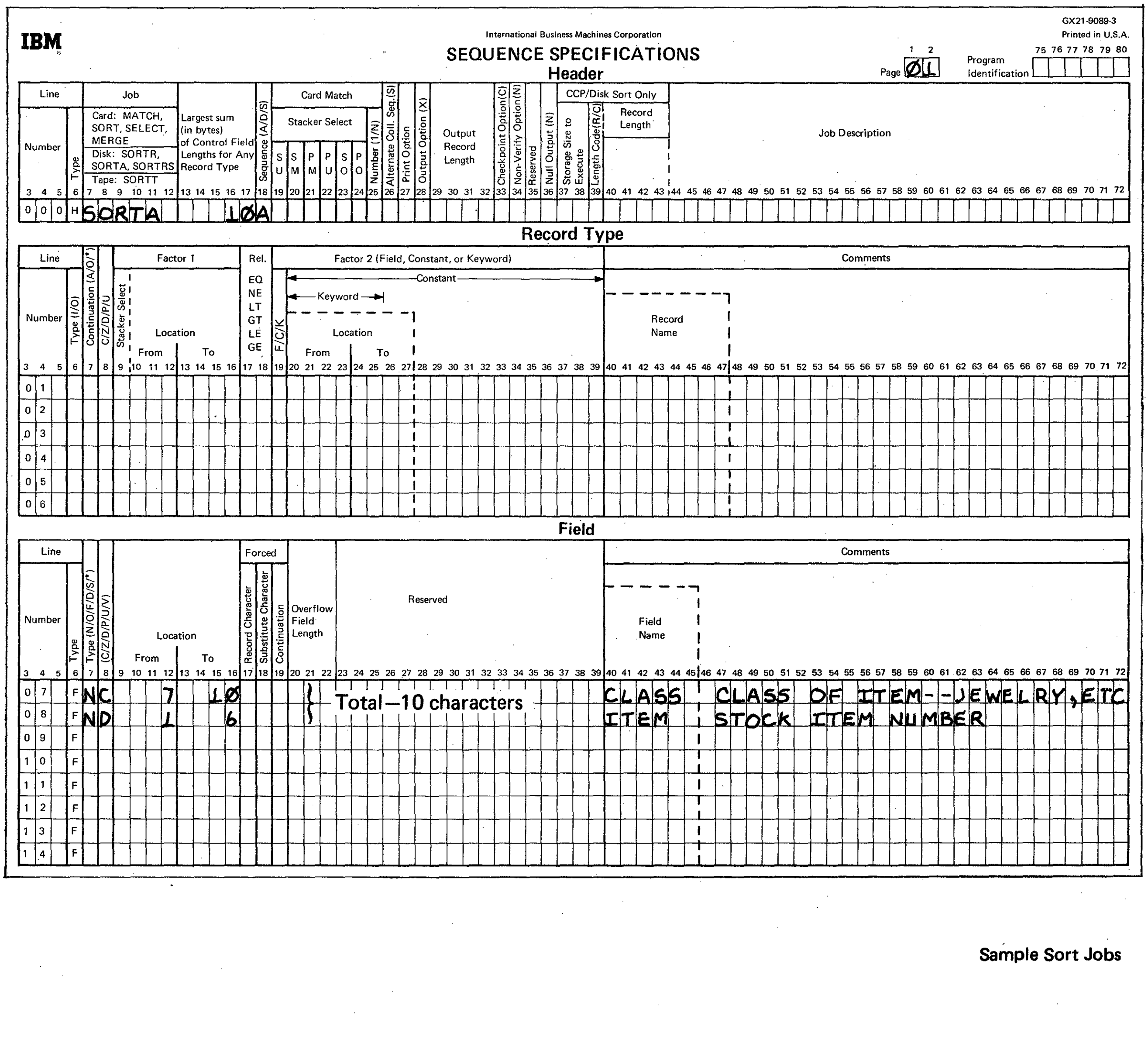
- Assume that the inventory file is an indexed file. The ITEM field is the record key used in the index. (The record keys are in ascending order.)

- This job creates a record address file containing relative record numbers of all records in the file. It sorts them into ascending order by the CLASS field. Within each class, addresses are also sorted into ascending order by ITEM field.

- As a result of this job, inventory records can be processed consecutively in either of two ways:

  1. By using the file index, which is in ascending order by ITEM field.

  2. By using the record address file created by the sort program, which is in ascending order by CLASS and also by ITEM within each class.



IBM

International Business Machines Corporation

SEQUENCE SPECIFICATIONS

GX21-9089-3
Printed in U.S.A.

Page Ø1

Program Identification

Header

Line Number (3-5)	Type (6)	Job (7-12) Card: MATCH, SORT, SELECT, MERGE; Disk: SORTR, SORTA, SORTRS; Tape: SORTT	Largest sum (in bytes) of Control Field Lengths for Any Record Type (13-17)	Sequence (A/D/S) (18)	Card Match Stacker Select SU SM PM PU SO PO (19-24)	Number (1/N) (25)	Alternate Coll. Seq.(S) (26)	Print Option (27)	Output Option (X) (28)	Output Record Length (29-32)	Checkpoint Option(C) (33)	Non-Verify Option(N) (34)	Reserved (35)	Null Output (N) (36)	CCP/Disk Sort Only: Storage Size to Execute (37-38)	Length Code(R/C) (39)	Record Length (40-43)	Job Description (44-72)
000	H	SORTA	1Ø	A														

Record Type

Line Number (3-5)	Type (I/O) (6)	Continuation (A/O/*) (7)	C/Z/D/P/U (8)	Stacker Select (9)	Factor 1 Location From (10-12)	Factor 1 Location To (13-16)	Rel. EQ NE LT GT LE GE (17-18)	F/C/K (19)	Factor 2 (Field, Constant, or Keyword) Location From (20-23)	To (24-27)	(28-39)	Comments: Record Name (40-47)	Comments (48-72)
01													
02													
03													
04													
05													
06													

Field

Line Number (3-5)	Type (6)	Type (N/O/F/D/S/*) (7)	(C/Z/D/P/U/V) (8)	Location From (9-12)	Location To (13-16)	Forced: Record Character (17)	Forced: Substitute Character (18)	Continuation (19)	Overflow Field Length (20-22)	Reserved (23-39)	Comments: Field Name (40-45)	Comments (46-72)
07	F	N	C	7	1Ø				}	Total—10 characters	CLASS	CLASS OF ITEM--JEWELRY,ETC
08	F	N	D	1	6				}		ITEM	STOCK ITEM NUMBER
09	F											
10	F											
11	F											
12	F											
13	F											
14	F											

Header Specification

Column 6

H identifies the line as a header line.

Columns 7-12

SORTA identifies the job as an addrout sort.

Columns 13-17

The control fields used to sort addresses in this job are CLASS and ITEM. The total of their lengths is 10, which is the entry for columns 13 through 17.

Column 18

The A stands for ascending order. The records are sorted into ascending order by item class (CLASS field), and by item number (ITEM field) within each class.

Record Type Specifications

Because all input records are being used and all have the same field specifications, no record type specifications are needed (include-all is implied).

Field Specifications

Column 6

F identifies the lines as field specification lines.

Column 7

Both fields (CLASS and ITEM) are normal control fields. The record addresses are sorted into ascending order by CLASS field and by ITEM field within each class:

CLASS Sequence	ITEM Sequence	Sequence of Records in Sorted Output File
Lowest Class No.	Lowest item no. Next higher item no. Third higher item no. . . . Highest item no.	Lowest class no. - Lowest item no. Lowest class no. - Next higher item no. Lowest class no. - Third higher item no. . . . Lowest class no. - Highest item no.
Each Higher Class No.	Lowest item no. Next higher item no. Third higher item no. . . . Highest item no.	Class no. - Lowest item no. Class no. - Next higher item no. Class no. - Third higher item no. . . . Class no. - Highest item no.
Highest Class No.	Lowest item no. Next higher item no. Third higher item no. . . . Highest item no.	Highest class no. - Lowest item no. Highest class no. - Next higher item no. Highest class no. - Third higher item no. . . . Highest class no. - Highest item no.

Column 8

C means that both zone and digit portions of the characters in the CLASS control field are used in the sorting process.

D means that only the digit portions of the characters in the ITEM control field are used in the sorting process. This causes blanks to be treated as zeros.

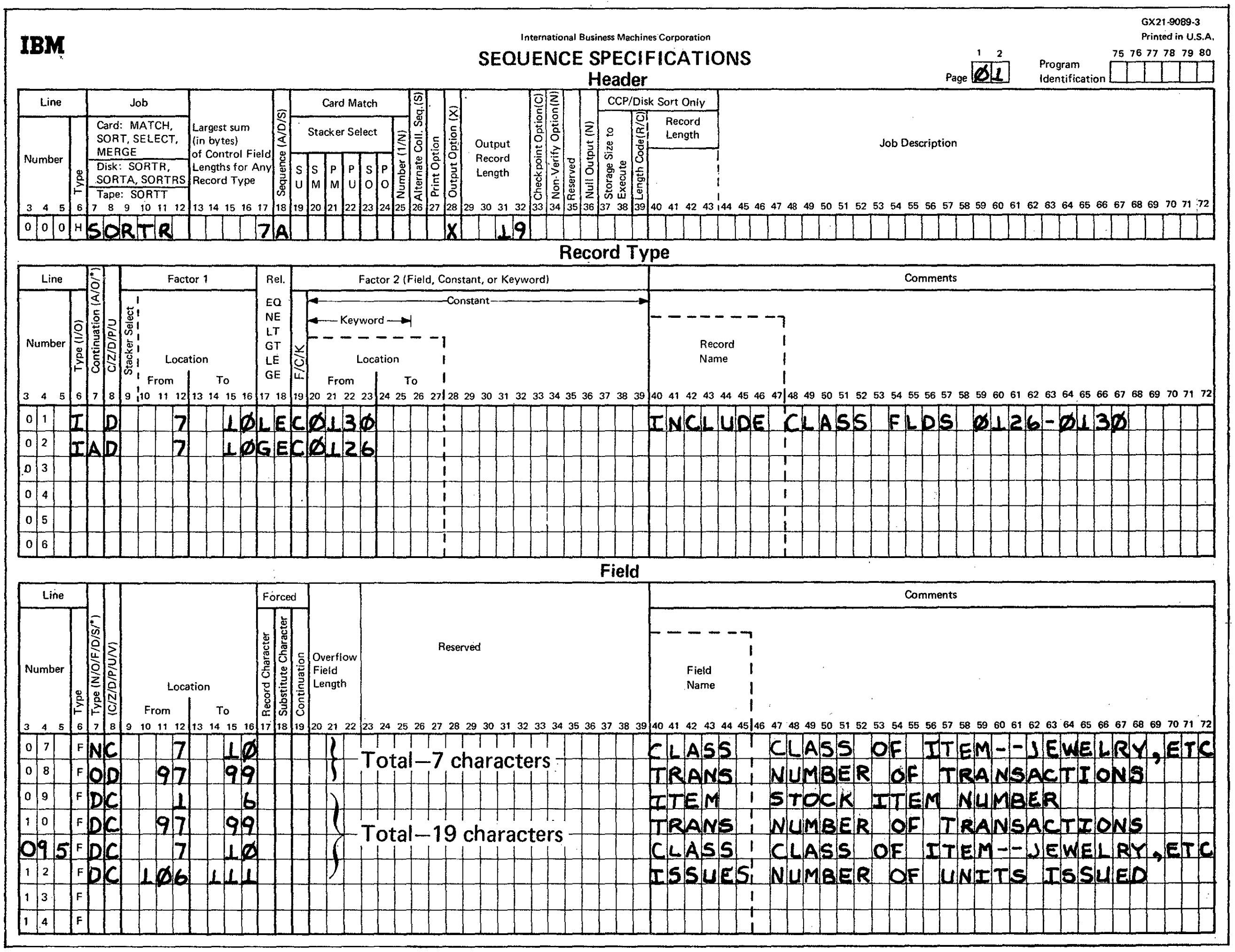
Columns 9-16

The CLASS control field is located in positions 7 through 10 of the input records.

The ITEM control field is located in positions 1 through 6 of the input records.

SAMPLE JOB 3: PRODUCING A FILE OF INVENTORY INFORMATION ABOUT CERTAIN ITEMS

- The program selects inventory records for stock items in classes 0126 through 0130.
- Output records are to contain only inventory information that indicates the activity (transactions) involving the stock items. The information is in these fields: ITEM, CLASS, TRANS, and ISSUES.
- Output records are to be sorted in ascending order by class (CLASS field), and in descending order by number of transactions (TRANS field) within each class.



IBM

International Business Machines Corporation

SEQUENCE SPECIFICATIONS

GX21-9089-3
Printed in U.S.A.

Page 1 2: Ø1

Program Identification 75 76 77 78 79 80:

Header

Line Number	Type	Job (Card: MATCH, SORT, SELECT, MERGE / Disk: SORTR, SORTA, SORTRS / Tape: SORTT)	Largest sum (in bytes) of Control Field Lengths for Any Record Type	Sequence (A/D/S)	Card Match Stacker Select SU	SM	PM	PU	SO	PO	Number (1/N)	Alternate Coll. Seq.(S)	Print Option	Output Option (X)	Output Record Length	Checkpoint Option(C)	Non-Verify Option(N)	Reserved	Null Output (N)	CCP/Disk Sort Only Storage Size to Execute	Length Code(R/C)	Record Length	Job Description
3 4 5	6	7 8 9 10 11 12	13 14 15 16 17	18	19	20	21	22	23	24	25	26	27	28	29 30 31 32	33	34	35	36	37 38	39	40 41 42 43	44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72
000	H	SORTR	7	A										X	19								

Record Type

Line Number	Type (I/O)	Continuation (A/O/*)	C/Z/D/P/U	Stacker Select	Factor 1 Location From	To	Rel. EQ NE LT GT LE GE	F/C/K	Factor 2 (Field, Constant, or Keyword) Location From	To		Comments Record Name	
3 4 5	6	7	8	9	10 11 12	13 14 15 16	17 18	19	20 21 22 23	24 25 26 27	28 29 30 31 32 33 34 35 36 37 38 39	40 41 42 43 44 45 46 47	48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72
01	I		D		7	1Ø	LE	C	Ø13Ø			INCLUDE	CLASS FLDS Ø126-Ø13Ø
02	I	A	D		7	1Ø	GE	C	Ø126				
03													
04													
05													
06													

Field

Line Number	Type	Type (N/O/F/D/S/*)	(C/Z/D/P/U/V)	Location From	To	Forced Record Character	Substitute Character	Continuation	Overflow Field Length	Reserved	Comments Field Name	
3 4 5	6	7	8	9 10 11 12	13 14 15 16	17	18	19	20 21 22	23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39	40 41 42 43 44 45	46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72
07	F	N	C	7	1Ø					Total—7 characters	CLASS	CLASS OF ITEM--JEWELRY,ETC
08	F	O	D	97	99						TRANS	NUMBER OF TRANSACTIONS
09	F	D	C	1	6					Total—19 characters	ITEM	STOCK ITEM NUMBER
10	F	D	C	97	99						TRANS	NUMBER OF TRANSACTIONS
Ø95	F	D	C	7	1Ø						CLASS	CLASS OF ITEM--JEWELRY,ETC
12	F	D	C	1Ø6	111						ISSUES	NUMBER OF UNITS ISSUED
13	F											
14	F											

Header Specification

Column 6

H identifies the line as a header line.

Columns 7-12

SORTR identifies the job as a tagalong sort.

Columns 13-17

The control fields used to sort records in this job are CLASS and TRANS. The total of their lengths is 7, which is the entry for columns 13 through 17.

Column 18

The A stands for ascending order. One control field (CLASS) is defined as a normal control field in field specifications. The other control field (TRANS) is an opposite control field. The records, therefore, are sorted into ascending order by item class (CLASS field), and into descending order by number of transactions (TRANS field) within each class.

Column 28

X indicates that control fields are being dropped from the output records because the TRANS field is an opposite control field. The program must convert the information from the TRANS field into a special form so that the records can be sorted properly. The program does not return the information to its original form after sorting. The information, therefore, would be meaningless.

The information from these two fields, however, is necessary in the output records. Therefore, the fields are also defined as data fields (see the explanation of column 7 of the field specifications for this example).

Columns 29-32

Because control fields are dropped from the output records, the length of the output records is the total length of the fields defined as data fields in field specifications. The length is 19, which is the entry in columns 29 through 32.

Record Type Specifications

Column 6

I identifies the record type lines as include lines. The input records described by these lines are used by the program. All other input records are omitted.

Column 7

The A in column 7 of line 02 means that line 02 is a continuation of the definition of the record described in line 01. The A means AND. An input record must meet all conditions listed in lines 01 and 02 before it is included.

Column 8

D in column 8 of lines 01 and 02 means that the digit portions of the characters in positions 7 through 10 of the records (factor 1) are compared with the digit portions of the constants 0126 and 0130 (factor 2). Leading blanks in positions 7 through 10 are treated like zeros. Therefore, ␢126 (␢ is a blank) and 0126 are considered equal.

Columns 9-16, 17-18, and 20-39

Inventory records that indicate the activity (transactions) of stock classes 0126 through 0130 are to be selected:

- Columns 9 through 16 (factor 1) identify the location of the CLASS field (record positions 7 through 10). The CLASS field contains 4-digit class numbers.
- Columns 20 through 23 list constants (C in column 19), that in this example define the upper and lower limits of the classes being selected from the inventory file.
- Columns 17 through 18 define the relationship between the input record field being used to build the file (factor 1) and the constants in factor 2. All inventory classes greater than or equal to (GE) 0126 and less than or equal to (LE) 0130 are to be included in the sorted file.

Column 19 and Columns 20-39

C in column 19 indicates that factor 1 is compared to a constant in columns 20 through 39. The constant in line 01 is 0130. The constant in line 02 is 0126.

Field Specifications

Column 6

F in column 6 of lines 07 through 12 indicates that these lines are field specification lines.

Column 7

Column 7 indicates the types of fields being described. The fields described in lines 07 and 08 (CLASS and TRANS, respectively) are control fields. CLASS is a normal control field. TRANS is an opposite control field. Output records are sorted so that CLASS fields are in the order indicated in column 18 of the header specifications (ascending order). Records having the same CLASS field are in descending order according to the TRANS field.

The fields described in lines 09 through 12 are data fields. They are included in the output record but have no effect on the sequence of the records.

Note that the CLASS and TRANS fields are described once as control fields and once as data fields because TRANS is an opposite control field. Its contents in the control field portion of the output records are changed in the sorting process and are meaningless for later processing. Therefore the control fields are dropped from the output records (X in column 28 of the header specifications) and CLASS and TRANS are defined as data fields so that they still appear in the output records.

The form of the output record is as follows:

	ITEM	CLASS	TRANS	ISSUES
Positions	1 ←→ 6	7 ←→ 10	11 ←→ 13	14 ←→ 19

Column 8

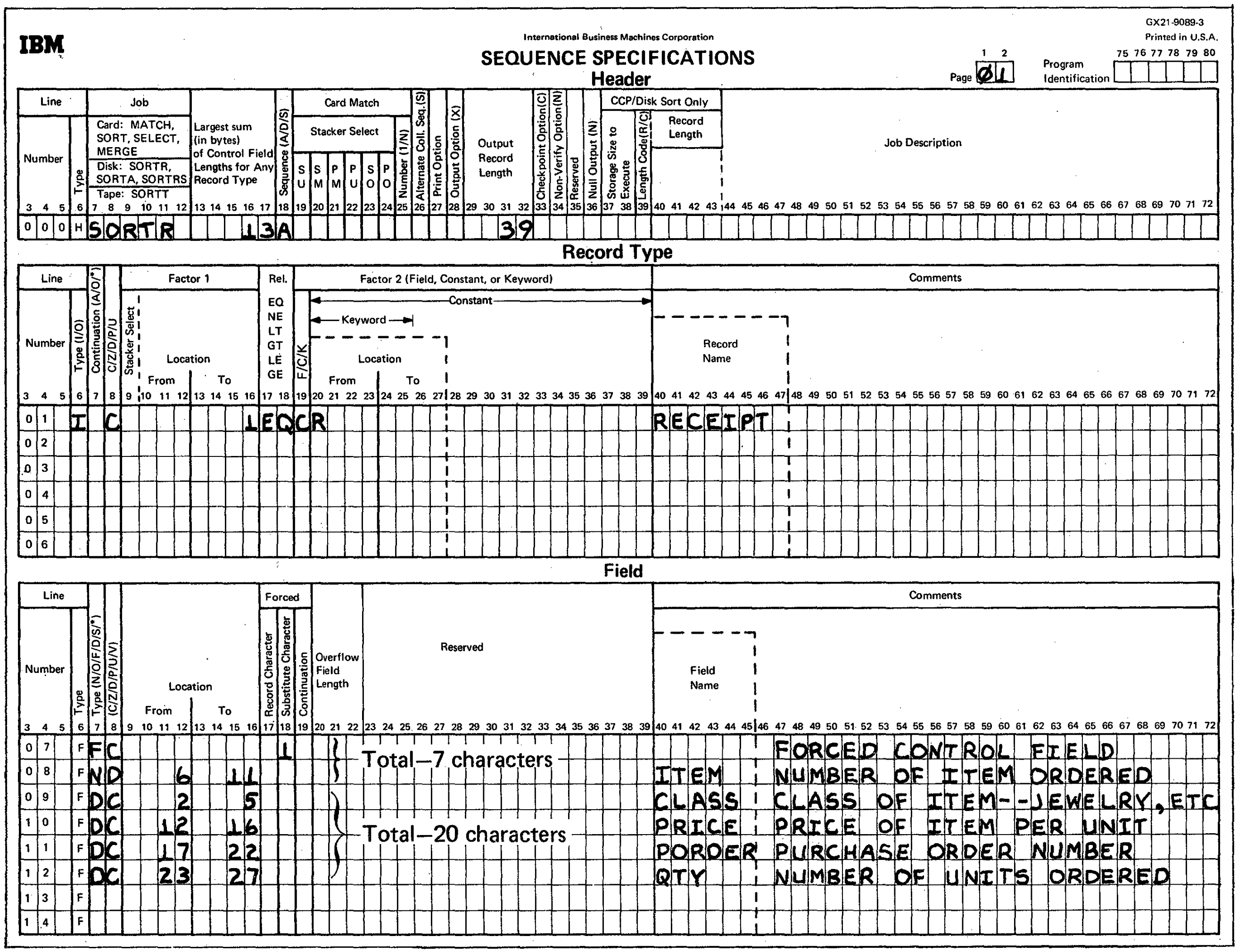
C in column 8, line 07, and in column 8, lines 9 through 12 means that both the zone and digit portions of all characters in all fields are used as they appear in the input records. D in column 8, line 08, means that only the digit portions of the characters in the TRANS control field are used in the sorting process.

Columns 9-16

Columns 9 through 16 identify the locations of fields in the input records.

SAMPLE JOB 4: PRODUCING A HISTORY FILE OF TRANSACTION INFORMATION

- The program selects records for all transactions in the transaction file.
- The entire file of input records is used to create output records. The output records are used to produce a history of the activity involving various stock.
- Output records are sorted by record type: RECEIPT records first, ISSUE records second, and ADJUST (adjustment) records last. Each type of record is sorted into ascending order by item number (ITEM field). The ISSUE records also have a subcontrol field—DATE OF ORDER.



IBM

International Business Machines Corporation

GX21-9089-3
Printed in U.S.A.

SEQUENCE SPECIFICATIONS

Page ØI (1 2)

Program Identification (75 76 77 78 79 80)

Header

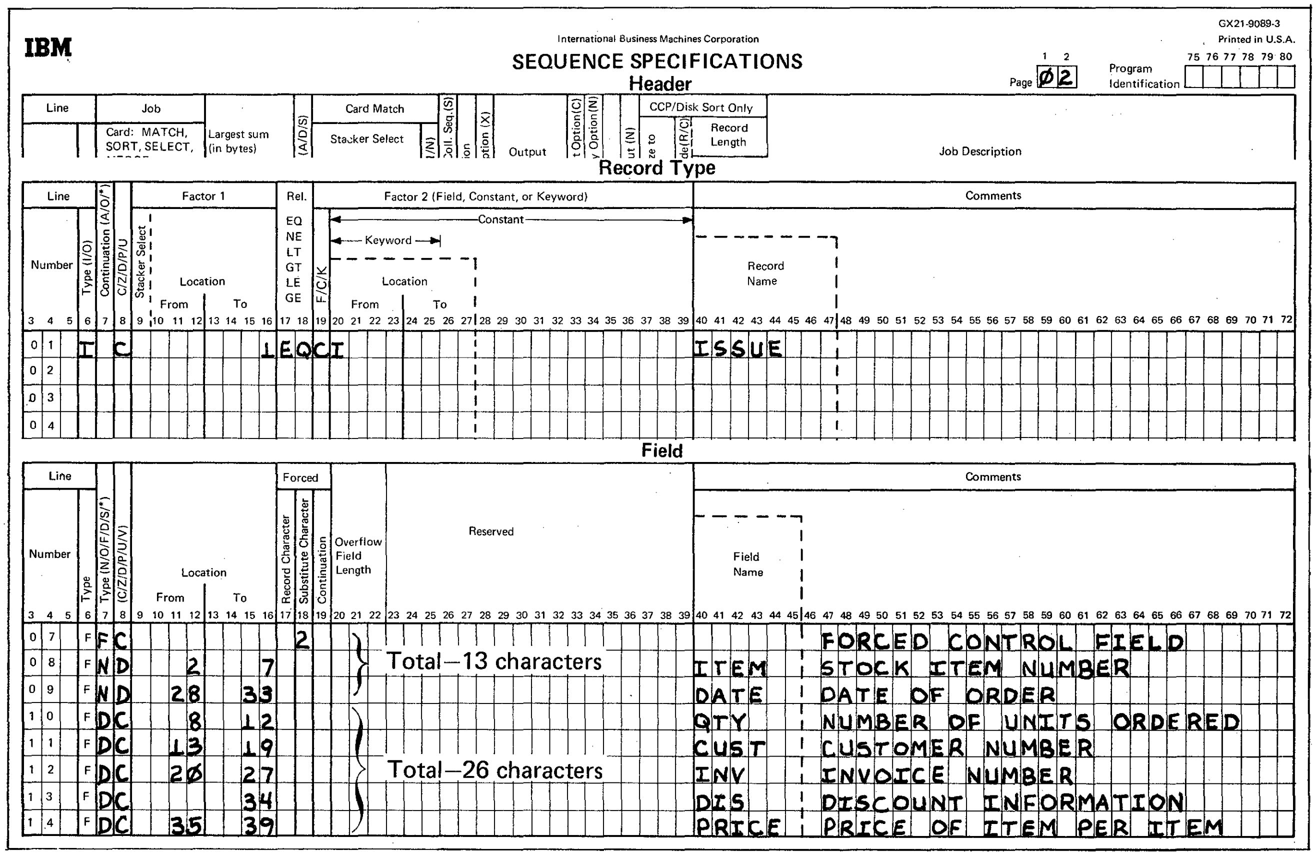
Line Number (3-5)	Type (6)	Job (7-12) Card: MATCH, SORT, SELECT, MERGE; Disk: SORTR, SORTA, SORTRS; Tape: SORTT	Largest sum (in bytes) of Control Field Lengths for Any Record Type (13-17)	Sequence (A/D/S) (18)	Card Match Stacker Select S U (19)	S M (20)	P M (21)	P U (22)	S O (23)	P O (24)	Number (1/N) (25)	Alternate Coll. Seq. (S) (26)	Print Option (27)	Output Option (X) (28)	Output Record Length (29-32)	Checkpoint Option (C) (33)	Non-Verify Option (N) (34)	Reserved (35)	Null Output (N) (36)	CCP/Disk Sort Only Storage Size to Execute (37-38)	Length Code (R/C) (39)	Record Length (40-43)	Job Description (44-72)
000	H	SORTR	13	A											39								

Record Type

Line Number (3-5)	Type (I/O) (6)	Continuation (A/O/*) (7)	C/Z/D/P/U (8)	Stacker Select (9)	Factor 1 Location From (10-12)	To (13-16)	Rel. EQ NE LT GT LE GE (17-18)	F/C/K (19)	Factor 2 (Field, Constant, or Keyword) Location From (20-23) / To (24-27) / Constant / Keyword (20-39)	Comments Record Name (40-46)	(47-72)
01	I		C			1	EQ	C	R	RECEIPT	
02											
03											
04											
05											
06											

Field

Line Number (3-5)	Type (6)	Type (N/O/F/D/S/*) (7)	C/Z/D/P/U/V (8)	Location From (9-12)	To (13-16)	Forced Record Character (17)	Substitute Character (18)	Continuation (19)	Overflow Field Length (20-22)	Reserved (23-39)	Comments Field Name (40-45)	(46-72)
07	F	F	C			1				Total—7 characters		FORCED CONTROL FIELD
08	F	N	D	6	11						ITEM	NUMBER OF ITEM ORDERED
09	F	D	C	2	5					Total—20 characters	CLASS	CLASS OF ITEM--JEWELRY, ETC
10	F	D	C	12	16						PRICE	PRICE OF ITEM PER UNIT
11	F	D	C	17	22						PORDER	PURCHASE ORDER NUMBER
12	F	D	C	23	27						QTY	NUMBER OF UNITS ORDERED
13	F											
14	F											



IBM

International Business Machines Corporation

SEQUENCE SPECIFICATIONS

GX21-9089-3 Printed in U.S.A.

Page ∅2

Program Identification

Header

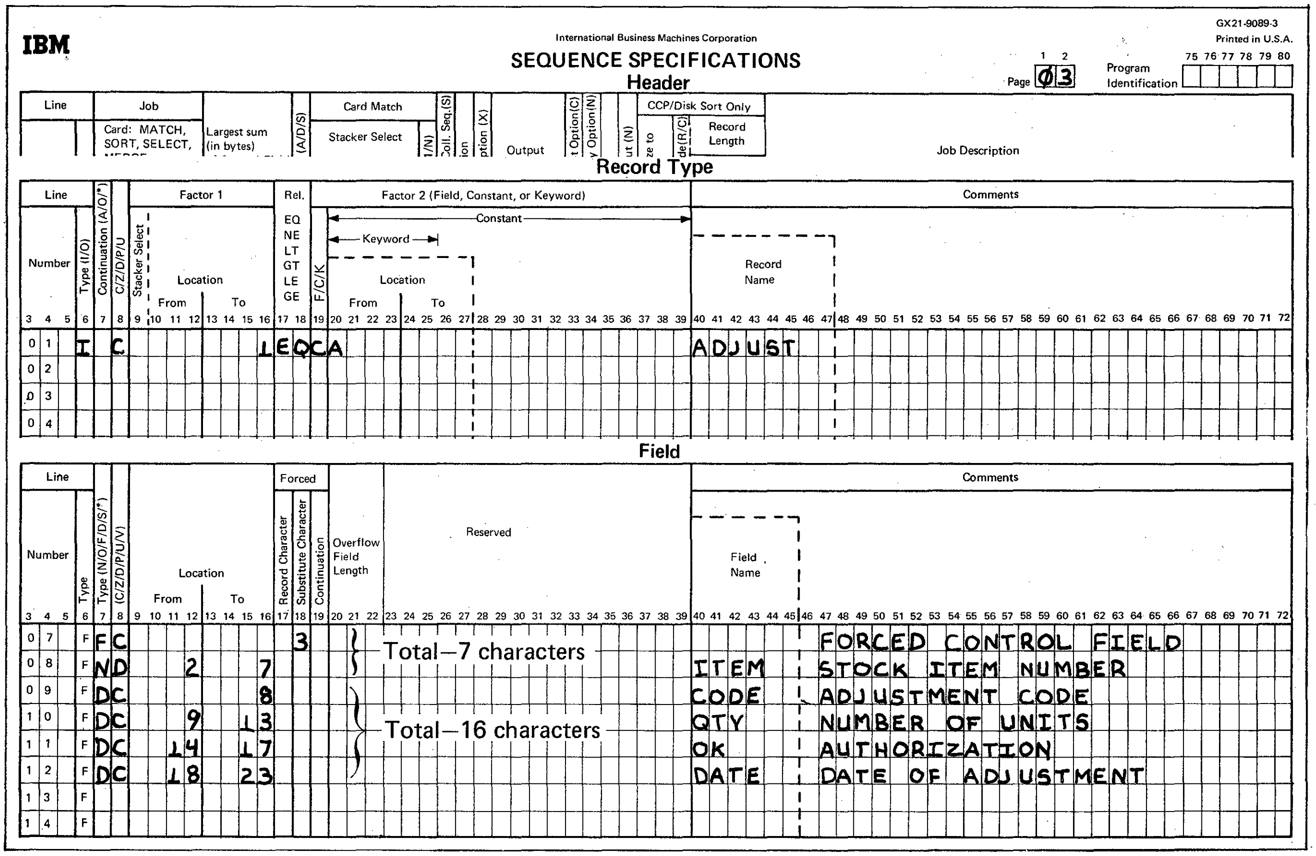
Line | Job (Card: MATCH, SORT, SELECT, MERGE) | Largest sum (in bytes) | (A/D/S) | Card Match Stacker Select | Output | CCP/Disk Sort Only: Record Length | Job Description

Record Type

Line Number	Type (I/O)	Continuation (A/O/*)	C/Z/D/P/U	Stacker Select	Factor 1 Location From	Factor 1 Location To	Rel.	F/C/K	Factor 2 (Field, Constant, or Keyword)	Record Name	Comments
01	I		C			1	EQ	C	I	ISSUE	
02											
03											
04											

Field

Line Number	Type	Type (N/O/F/D/S/*)	(C/Z/D/P/U/V)	Location From	Location To	Forced Record Character	Forced Substitute Character	Continuation	Overflow Field Length	Reserved	Field Name	Comments
07	F	F	C				2			Total—13 characters		FORCED CONTROL FIELD
08	F	N	D	2	7						ITEM	STOCK ITEM NUMBER
09	F	N	D	28	33						DATE	DATE OF ORDER
10	F	D	C	8	12					Total—26 characters	QTY	NUMBER OF UNITS ORDERED
11	F	D	C	13	19						CUST	CUSTOMER NUMBER
12	F	D	C	2∅	27						INV	INVOICE NUMBER
13	F	D	C		34						DIS	DISCOUNT INFORMATION
14	F	D	C	35	39						PRICE	PRICE OF ITEM PER ITEM



IBM

International Business Machines Corporation

SEQUENCE SPECIFICATIONS

GX21-9089-3 Printed in U.S.A.

Page ∅3

Program Identification

Header

Line | Job (Card: MATCH, SORT, SELECT, MERGE) | Largest sum (in bytes) | (A/D/S) | Card Match Stacker Select | Output | CCP/Disk Sort Only: Record Length | Job Description

Record Type

Line Number	Type (I/O)	Continuation (A/O/*)	C/Z/D/P/U	Stacker Select	Factor 1 Location From	Factor 1 Location To	Rel.	F/C/K	Factor 2 (Field, Constant, or Keyword)	Record Name	Comments
01	I		C			1	EQ	C	A	ADJUST	
02											
03											
04											

Field

Line Number	Type	Type (N/O/F/D/S/*)	(C/Z/D/P/U/V)	Location From	Location To	Forced Record Character	Forced Substitute Character	Continuation	Overflow Field Length	Reserved	Field Name	Comments
07	F	F	C				3			Total—7 characters		FORCED CONTROL FIELD
08	F	N	D	2	7						ITEM	STOCK ITEM NUMBER
09	F	D	C		8					Total—16 characters	CODE	ADJUSTMENT CODE
10	F	D	C	9	13						QTY	NUMBER OF UNITS
11	F	D	C	14	17						OK	AUTHORIZATION
12	F	D	C	18	23						DATE	DATE OF ADJUSTMENT
13	F											
14	F											

Header Specification

Column 6

H identifies the line as a header line.

Columns 7-12

SORTR identifies the job as a tagalong sort.

Columns 13-17

Three types of records are described: one on each sheet. The total length of the control fields for each type of record is 7, 13, and 7 characters, respectively. The largest of these totals (13) is the entry used in columns 13 through 17.

Column 18

The A stands for ascending order. It indicates the order into which output records (described in the three sets of field lines) will be sorted.

Columns 29-32

The total length of control fields for each type of record described is 7, 13, and 7 respectively. The total length of data fields for each type of record is 20, 26, and 16 respectively. The largest control-field total (13) is added to the largest data-field total (26) and the result (39) is used in columns 29 through 32.

Record Type Specifications

Column 6

Column 6 in line 01 of each of the three sheets contains an I indicating that the lines are include lines. The input records described by these lines are used by the program. All other input records are omitted.

Column 8, Columns 9-16, Columns 17-18, Column 19, and Columns 20-39

Column 8 in line 01 of each of the three sheets contains a C indicating the the entire character in position 1 of the input records is to be compared with the constants R, I, and A respectively. Only input records with one of those characters in position 1 will be included in the job.

In the record type lines, columns 9 through 39 have the following meanings:

- Columns 9 through 16 identify the input-record position (position 1) being compared against R, I, and A.
- Columns 17 through 18 indicate the comparison result that determines whether an input record is used (the character in position 1 must equal R, I, or A).
- C in column 19 of each line indicates that the characters against which the input-record character is compared are constants supplied in columns 20 through 39.
- Columns 20 through 39 supply the constants (R, I, and A, respectively) used in the comparison.

Field Specifications

Field specifications differ for the three types of records being described. The records, therefore, are described on separate coding sheets.

Column 6

F in column 6 of lines 07 through 12 on sheet 01, lines 07 through 14 on sheet 02, and lines 07 through 12 on sheet 03 indicate that the lines are field lines.

Column 7

Column 7 indicates the types of fields being described for each of the three types of records. The records described by sheets 01 and 03 (RECEIPT and ADJUST records respectively) have two control fields. The records described by sheet 02 (ISSUE records) have three. The remaining fields described are data fields.

These diagrams show the formats of the output records that will be created from each of the three types of input records. The number underneath the field identification blocks indicate the positions information will occupy in the output record.

Receipt Records (sheet 01)

Control Field			Data				
1	ITEM	binary zeros	CLASS	PRICE	PORDER	QTY	6 blanks
1	2←→7	8←→13	14←→17	18←→22	23←→28	29←→33	34←→39

Issue Records (sheet 02)

Control Field			Data				
2	ITEM	DATE	QTY	CUST	INV	DIS	PRICE
1	2←→7	8←→13	14←→18	19←→25	26←→33	34	35←→39

Adjust Records (sheet 03)

Control Field				Data			
3	ITEM	binary zeros	CODE	QTY	OK	DATE	10 blanks
1	2←→7	8←→13	14	15←→19	20←→23	24←→29	30←→39

The first control field in each type of output record is a forced control field. The number 1 will begin every RECEIPT output record, 2 will begin ISSUE records, and 3 will begin ADJUST records. This forces all RECEIPT output records to be first, all ISSUE records second, and all ADJUST records third.

The output records for this job must be the same length. The total number of characters in the control fields described for RECEIPT and ADJUST was less than the total for ISSUE records (the control field length for ISSUE on the header is used to assign the absolute length of the control field). Therefore, the program added binary zeros at the end of the control fields for RECEIPT and ADJUST output records. Blanks were added at the end of the data portion of RECEIPT and ADJUST output records to make the total length of the records equal to the total length of ISSUE output records.

Column 8

C means that both zone and digit portions of the characters in the fields are used. D means that only the digit portions of the characters in the ITEM control field are used in the sorting process.

Columns 9-16 and Column 18

Columns 9 through 16 of the field lines identify the locations of the fields in the input records.

The first control field for each type of record is a forced control field. On each sheet, it is described by an unconditional force line. For this reason, columns 9 through 16 are blank in those lines. Column 18 identifies the character to be used in the forced control field: 1 in output records created from RECEIPT records, 2 in ISSUE output records, and 3 in ADJUST output records.

SAMPLE JOB 5: PRODUCING A LIST OF INVENTORY ITEMS REQUIRING ADJUSTMENTS

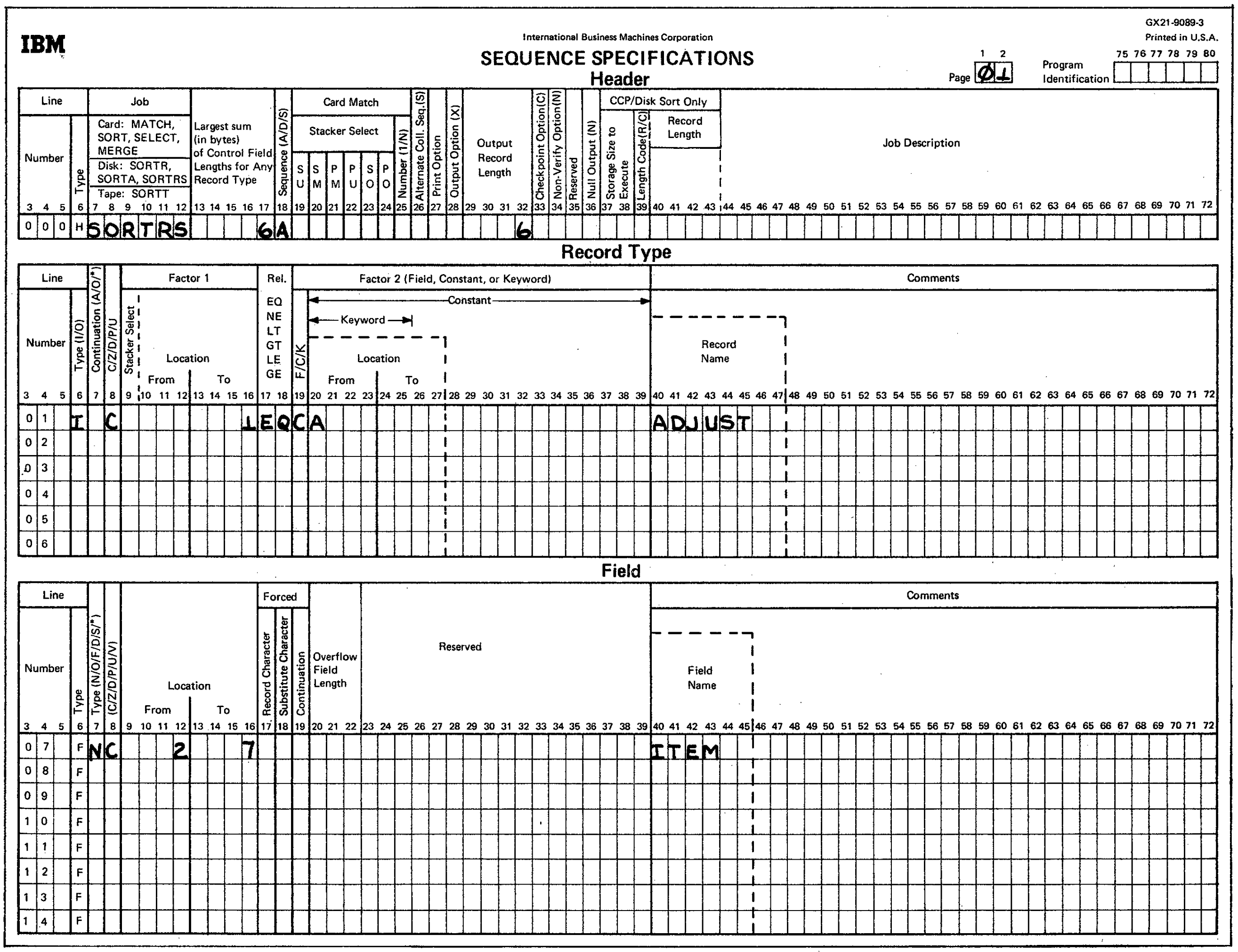
- The program selects all adjustment records in the transaction file.

- The output record is a control field consisting of the stock item number only.

- Since the output record consists of no data field and, in particular, no summary data fields, the result of the SORTRS job will be to eliminate all duplicate adjustment stock item numbers from the sorted file.

- The output records, each consisting of a unique stock item number that had an adjustment, are to be in ascending order.

- The output provides a list of all stock items that had an adjustment.



IBM

GX21-9089-3
Printed in U.S.A.

International Business Machines Corporation

SEQUENCE SPECIFICATIONS

Page (1 2): Ø1

Program Identification (75 76 77 78 79 80):

Header

Line Number (3 4 5)	Type (6)	Job (7–12) Card: MATCH, SORT, SELECT, MERGE Disk: SORTR, SORTA, SORTRS Tape: SORTT	Largest sum (in bytes) of Control Field Lengths for Any Record Type (13–17)	Sequence (A/D/S) (18)	Card Match Stacker Select SU (19)	SM (20)	PM (21)	PU (22)	SO (23)	PO (24)	Number (1/N) (25)	Alternate Coll. Seq.(S) (26)	Print Option (27)	Output Option (X) (28)	Output Record Length (29–32)	Checkpoint Option(C) (33)	Non-Verify Option(N) (34)	Reserved (35)	Null Output (N) (36)	CCP/Disk Sort Only Storage Size to Execute (37 38)	Length Code(R/C) (39)	Record Length (40–43)	Job Description (44–72)
000	H	SORTRS	6	A											6								

Record Type

Line Number (3 4 5)	Type (I/O) (6)	Continuation (A/O/*) (7)	C/Z/D/P/U (8)	Stacker Select (9)	Factor 1 Location From (10–12)	To (13–16)	Rel. EQ NE LT GT LE GE (17 18)	F/C/K (19)	Factor 2 (Field, Constant, or Keyword) Location From (20–23)	To (24–27)	(28–39)	Comments Record Name (40–47)	(48–72)
01	I		C			1	EQ	C	A			ADJUST	
02													
03													
04													
05													
06													

Field

Line Number (3 4 5)	Type (6)	Type (N/O/F/D/S/*) (7)	(C/Z/D/P/U/V) (8)	Location From (9–12)	To (13–16)	Forced Record Character (17)	Substitute Character (18)	Continuation (19)	Overflow Field Length (20–22)	Reserved (23–39)	Comments Field Name (40–45)	(46–72)
07	F	N	C	2	7						ITEM	
08	F											
09	F											
10	F											
11	F											
12	F											
13	F											
14	F											

Header Specification

Column 6

H identifies the line as a header line.

Columns 7-12

SORTRS identifies the job as a summary sort.

Columns 13-17

Only one control field is used in this job: ITEM. Its length is 6, which is the entry for columns 13 through 17.

Column 18

The A stands for ascending order.

Columns 29-32

The output record consists of one control field only. The output record length is 6, the same as the control field length.

Record Type Specifications

Column 6

I identifies the record type line 01 as an include line. Input records described by this line are to be sorted. Since this is the only include set, all other records are omitted.

Column 7

Line 01 is the first (and only) include line of this include set consisting of lines 01 and 07. Therefore, column 7 must be blank.

Column 8

C in column 8 tells sort to use both the zone and digit portions of the factor 1 and factor 2 fields described in line 01.

Columns 13-16, 17-18, 19, and 20

Only adjustment records containing an A in position 1 are to be included in the sort.

Field Specifications

Column 6

F in column 6 of line 07 indicates that this is a field specification line.

Columns 7 and 8

NC in columns 7 and 8 of line 07 indicates that the ITEM field described is a normal control field consisting of both the zone and digit portions of the characters in the ITEM field of the input records included in the sort.

Columns 9-16

Columns 9 through 16 of line 07 identify the location of the ITEM field (having a length of 6) that is to be used as the control field.

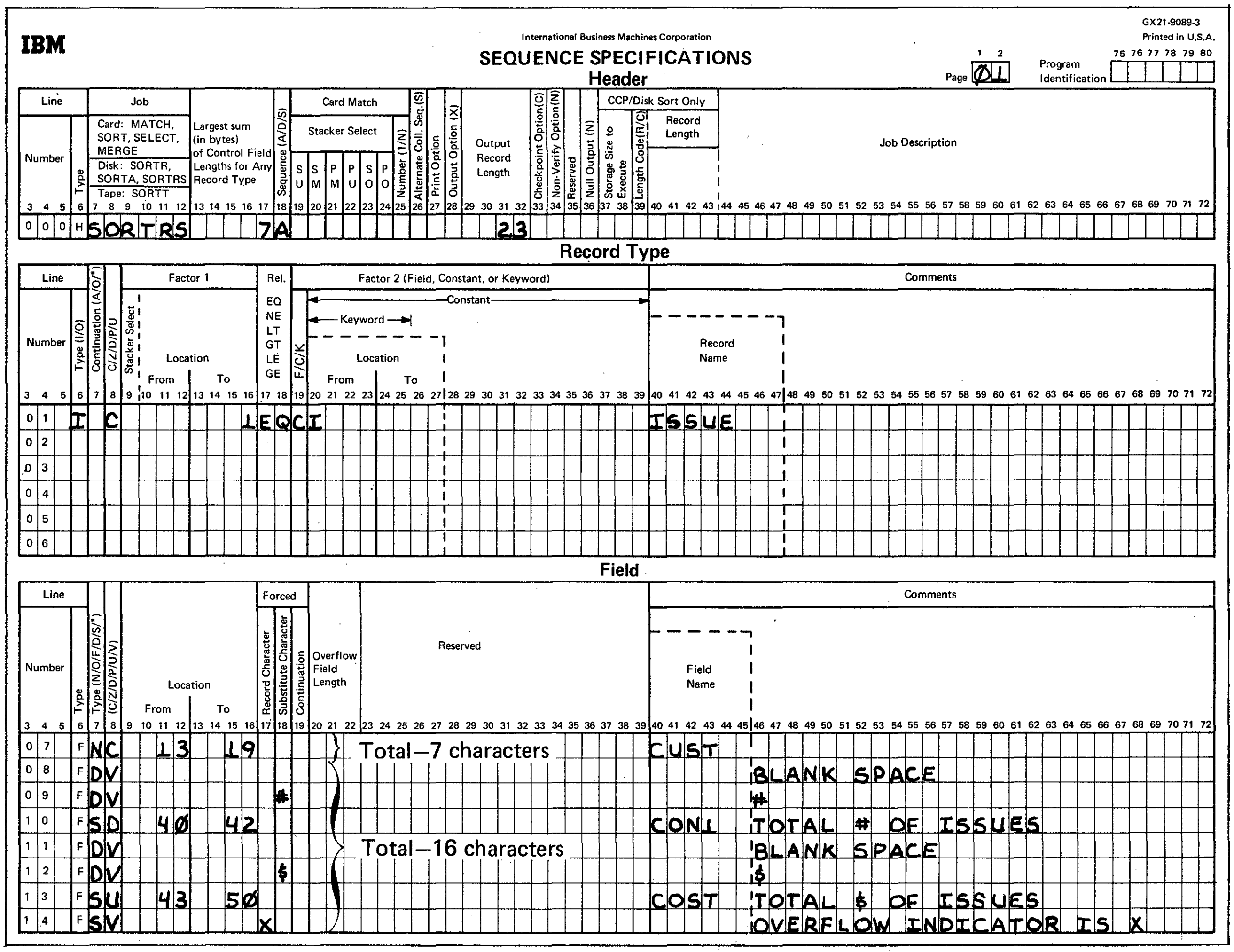
SAMPLE JOB 6: PRODUCING A SUMMARY HISTORY FILE OF ISSUES BY CUSTOMER NUMBER

- The program selects all ISSUE records from the transaction file.

- Using a summary sort (SORTRS), a file is to be formed consisting of each customer to whom any stock item was issued.

- Each output record consists of a unique customer number, the total number of issues made to that number, and the total dollar value of all the issues.

- The output records are to be in ascending order by customer number.



IBM

International Business Machines Corporation

GX21-9089-3
Printed in U.S.A.

SEQUENCE SPECIFICATIONS

Page ØI (1 2)

Program Identification (75 76 77 78 79 80)

Header

Line Number (3 4 5)	Type (6)	Job (Card: MATCH, SORT, SELECT, MERGE; Disk: SORTR, SORTA, SORTRS; Tape: SORTT) (7–12)	Largest sum (in bytes) of Control Field Lengths for Any Record Type (13–17)	Sequence (A/D/S) (18)	Card Match Stacker Select SU SM PM PU SO PO (19–24)	Number (1/N) (25)	Alternate Coll. Seq.(S) (26)	Print Option (27)	Output Option (X) (28)	Output Record Length (29–32)	Checkpoint Option(C) (33)	Non-Verify Option(N) (34)	Reserved (35)	Null Output (N) (36)	CCP/Disk Sort Only: Storage Size to Execute (37–38)	Length Code(R/C) (39)	Record Length (40–43)	Job Description (44–72)
000	H	SORTRS	7	A						23								

Record Type

Line Number (3 4 5)	Type (I/O) (6)	Continuation (A/O/*) (7)	C/Z/D/P/U (8)	Stacker Select (9)	Factor 1 Location From (10–12)	Factor 1 Location To (13–16)	Rel. EQ NE LT GT LE GE (17–18)	F/C/K (19)	Factor 2 (Field, Constant, or Keyword) Location From (20–23)	Location To (24–27)	Constant (28–39)	Comments: Record Name (40–47)	(48–72)
01	I		C			1	EQ	C	I			ISSUE	
02													
03													
04													
05													
06													

Field

Line Number (3 4 5)	Type (6)	Type (N/O/F/D/S/*) (7)	(C/Z/D/P/U/V) (8)	Location From (9–12)	Location To (13–16)	Forced Record Character (17)	Forced Substitute Character (18)	Continuation (19)	Overflow Field Length (20–22)	Reserved (23–39)	Comments: Field Name (40–45)	(46–72)
07	F	N	C	13	19					Total—7 characters	CUST	
08	F	D	V									BLANK SPACE
09	F	D	V				#					#
10	F	S	D	4Ø	42						CON1	TOTAL # OF ISSUES
11	F	D	V							Total—16 characters		BLANK SPACE
12	F	D	V				$					$
13	F	S	U	43	5Ø						COST	TOTAL $ OF ISSUES
14	F	S	V			X						OVERFLOW INDICATOR IS X

Header Specification

Column 6

H identifies the line as a header line.

Columns 7-12

SORTRS identifies the job as a summary sort.

Columns 13-17

The work record as well as the output record consists of a control field made up of the customer number. Its length is 7, which is the entry for columns 13 through 17.

Column 18

The A indicates that the control field is to be sorted in ascending order.

Columns 29-32

The control field is not to be dropped when writing the output file. The length of the output record is the control field (seven characters as specified in line 07) added to the data fields (16 characters as specified in lines 08 through 14). This totals 23 characters, and is the entry for columns 29 through 32.

Record Type Specifications

Column 6

I identifies the record type line 01 as an include line. Input records described by this line are to be sorted. Since this is the only include set, all other input records are omitted.

Column 7

Line 01 is the first (and only) include line of this include set consisting of lines 07 through 14; therefore, column 7 must be blank.

Columns 8, 16, 17-18, 19, and 20

Only ISSUE records containing the full character I in position 1 of the input records are to be included in the sort.

Field Specifications

Column 6

F in column 6 of lines 07 through 14 indicates that the lines are field lines.

Columns 7 and 8

NC in line 07 specifies that this is the control field to be sorted in the sequence specified in column 18 of the header line (ascending order) using both the zone and digit portions of the characters.

Lines 08, 09, 11, and 12 specify unconditional 1-character force fields.

Line 10 indicates that this is a summary data field. When the data in the input record (positions 40 through 42) is used to build a work record, only the digit portions of the characters will be used because CON1 is an unsigned decimal number. Using a D in column 8 with a signed number causes the absolute value of that signed number to be used.

Line 13 also is specified as a summary data field. However, U in column 8 indicates that this is signed unpacked decimal data. The sign is to be moved to the work record and used in subsequent summary additions.

Line 14 is specified as the summary overflow indicator byte.

Columns 9-16

Entries are placed in the From and To columns for lines 07, 10, and 13 to indicate the control field (CUST), digit summary data field (CON1), and the signed unpacked decimal summary field (COST), respectively.

For lines 08, 09, 11, and 12, the entries are blank since these are unconditional 1-character force fields.

For line 14 the entries are blank since this line is an overflow indicator specification.

Column 17

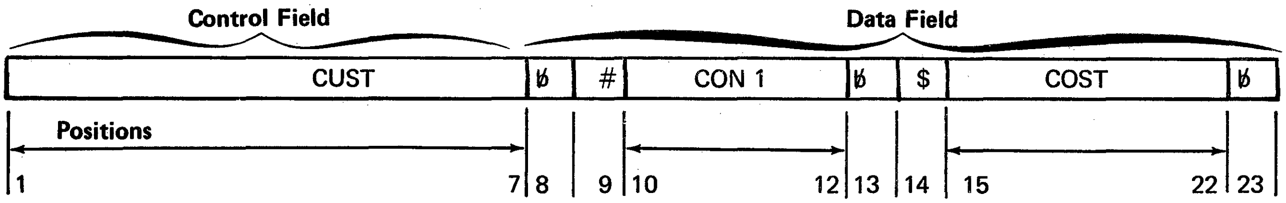
The X in line 14 defines the field as an overflow indicator field.

Column 18

This column is only used for lines 08, 09, 11, 12, and 14. The line numbers and the forced value in the data fields are:

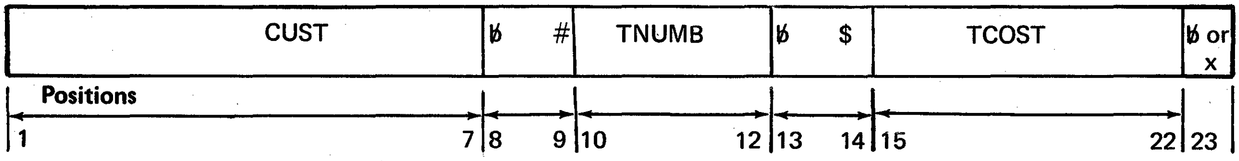
08	ƀ (blank)
09	#
11	ƀ (blank)
12	$
14	ƀ (blank)

The initial work record created from the ISSUE records will look like this :



Since this is a summary sort, the CON1 and COST fields will be summarized for records having identical control fields (CUST). These portions 10 through 12 (CON1) in the work records will be summarized to reflect the total number of issues (TNUMB) for a given customer number. Likewise, positions 15 through 22 (COST) will be summarized to reflect the total cost of all issues (TCOST).

The output record will look like this:



A blank in position 23 indicates that TNUMB and TCOST are valid sums. X in position 23 indicates that overflow occurred when the sort program was summarizing TNUMB and/or TCOST for this customer number, resulting in an invalid sum or sums.

SAMPLE JOB 7: SORTING SUMMARY HISTORY FILE BY DESCENDING TOTAL DOLLARS ISSUED (NEGATIVE VALUES POSSIBLE)

- The input for this job is the output file created by *Sample Job 6.* Each input record has the following format:

Record Position	Field Name	Contents
1-7	CUST	Customer number
8-9		Constants: ƀ#
10-12	TNUMB	Total number of issues to customer
13-14		Constants: ƀ$
15-22	TCOST	Total cost of all issues to customer
23	OFLOW	Overflow indicator: ƀ – TNUMB and TCOST are valid sums X – Invalid sum or sums; overflow occurred while summarizing TNUMB and/or TCOST

- Only records with valid TNUMB and TCOST will be used.

- A record sort of the valid records is desired; that is, the format of the output record is to be identical to the format of the input record.

- The records are to be sorted by total dollars issued (TCOST) in descending order. TCOST could be negative for some customer numbers. For this reason, a sort that considers the sign must occur.

IBM

International Business Machines Corporation

GX21-9089-3
Printed in U.S.A.

SEQUENCE SPECIFICATIONS

Page 1 2: Ø1

Program Identification 75 76 77 78 79 80: □□□□□□

Header

Line Number (3-5)	Type (6)	Job (7-12) Card: MATCH, SORT, SELECT, MERGE; Disk: SORTR, SORTA, SORTRS; Tape: SORTT	Largest sum (in bytes) of Control Field Lengths for Any Record Type (13-17)	Sequence (A/D/S) (18)	Card Match Stacker Select SU (19)	SM (20)	PM (21)	PU (22)	SO (23)	PO (24)	Number (1/N) (25)	Alternate Coll. Seq.(S) (26)	Print Option (27)	Output Option (X) (28)	Output Record Length (29-32)	Checkpoint Option(C) (33)	Non-Verify Option(N) (34)	Reserved (35)	Null Output (N) (36)	CCP/Disk Sort Only Storage Size to Execute (37-38)	Length Code(R/C) (39)	Record Length (40-43)	Job Description (44-72)
000	H	SORTR	8	D										X	23								

Record Type

Line Number (3-5)	Type (I/O) (6)	Continuation (A/O/*) (7)	C/Z/D/P/U (8)	Stacker Select (9)	Factor 1 Location From (10-12)	Factor 1 Location To (13-16)	Rel. EQ NE LT GT LE GE (17-18)	F/C/K (19)	Factor 2 (Field, Constant, or Keyword) (20-39)	Comments Record Name (40-72)
01	O		C			23	EQ	C	X	INVALID SUMS
02	I									INCLUDE-ALL POSITIVE TCOST
03		*								INFORMATION
04										
05										
06										

Field

Line Number (3-5)	Type (6)	Type (N/O/F/D/S/*) (7)	(C/Z/D/P/U/V) (8)	Location From (9-12)	Location To (13-16)	Forced Record Character (17)	Substitute Character (18)	Continuation (19)	Overflow Field Length (20-22)	Reserved (23-39)	Comments Field Name (40-72)
07	F	N	U	15	22						NORMAL UNPACKED
08	F	D	C	1	23						ENTIRE RECORD-DATA FIELD
09	F										
10	F										
11	F										
12	F										
13	F										
14	F										

Header Specification

Column 6

H identifies the line as a header line.

Columns 7-12

SORTR identifies the job as a tagalong sort.

Columns 13-17

One type of record is specified to be included. The total length of the control fields for this type is 8 (the entry in column 17).

Column 18

D stands for descending order. The largest positive number comes first; the largest negative number (the negative number having the largest absolute value) comes last.

For example, the absolute values of −19 and −99 are 19 and 99. −99 is more negative than −19 since its absolute value is greater than 19.

Column 28

X indicates that the control field is to be dropped when forming the output file.

Columns 29-32

When the control field is dropped, only the data portion remains. This data portion is the entire input record. Thus 23 is entered in columns 29 through 32.

Record Type Specifications

Columns 6 and 7

Two record types are specified. One type is to be omitted (O in column 6). One is to be included. Each record type is specified by one record type line. Since these lines are the first (as well as the last) record type lines in their omit set and include set, column 7 must be blank. Note that the asterisk in column 7, line 03, indicates that this line is only a comment line.

Columns 8, 9-16, 17-18, 19, and 20-39

For line 01, these columns specify that if position 23 in the input record contains the character constant X, the record is to be omitted.

Line 02 indicates that all records not omitted by line 01 are to be included. For each record included, a work record is formed by the field specifications, lines 07 and 08. Note that the control field is an 8-position unpacked field (line 07).

For a more complete discussion of numeric data, see *Columns 8 and Columns 9-16* under *Column Descriptions* in Chapter 3.

Field Specifications

Column 6

F in column 6 of lines 07 and 08 indicates that these lines are field specification lines.

Columns 7, 8, 9-16, and 18

- Line 07 indicates unpacked data in the control field from positions 15 through 22 of the input record. (This control field is dropped because an X is coded in column 28 of the header specifications.)

- Line 08 indicates that the entire record is included as data.

Appendix A. Calculating the Sizes of Files for Sort

INPUT FILE

Use this formula to calculate the size (in blocks) of an input file. Round the result up to the nearest whole number:

$$\text{BLOCKS} = \frac{\text{number of records in the file} \times \text{input record length}}{\text{number of bytes in a block (2,560)}}$$

OUTPUT FILE

Use this formula to calculate how many blocks the output file requires; round the result up to the nearest whole number:

$$\text{BLOCKS} = \frac{\text{number of records that will be selected for sorting} \times \text{output record length}}{\text{number of bytes in a block (2,560)}}$$

If you use the RECORDS parameter in the output file statement:

$$\text{RECORDS} = \text{number of records that will be selected for sorting}$$

WORK FILE

Use either of the following formulas to approximate how many blocks or records to specify on the FILE statement for the work file; whichever formula you use, round your results up to the nearest whole number:

$$\text{BLOCKS} = 2 + \left(\frac{\text{number of records that will be selected for sorting} \times \text{work record length} \times 1.1}{\text{number of bytes in a block (2,560)}}\right)$$

or

$$\text{RECORDS} = \text{number of records that will be selected for sorting} + \left(\frac{2 \times \text{number of bytes in a block (2,560)}}{\text{work record length}}\right)$$

Note: Sort includes an 8-byte work block vector in every work block written to the work file. Therefore, the work file must be larger than a file that would hold the exact number of records to be sorted. This is why both formulas approximate the number of records or blocks to specify on the FILE statement for the work file.

Work Record Length

Type of Sort	Type of Output	Work Record Length
SORTA	Address output	Length of control fields + 3
SORTR or SORTRS	Control fields only	Length of control fields as specified in the header line
	Control fields and data	Length of data + length of control fields. (This is the output record length specified in the header line.)
	Data only	Length of data + length of control fields. (Output record length specified in the header line plus the control field length specified in the header line.)

Appendix B. Collating Sequence

Standard EBCDIC Collating Sequence

When Both Zone and Digit Portions of Characters are Used		
Order in the Sequence	Character	Corresponding Hexadecimal Number [2]
1 (lowest)	blank	40
2	¢	4A
3	.	4B
4	<	4C
5	(	4D
6	+	4E
7	|	4F
8	&	50
9	!	5A
10	$	5B
11	*	5C
12	)	5D
13	;	5E
14	¬	5F
15	- (minus)	60
16	/	61
17	,	6B
18	%	6C
19	_ (underscore)	6D
20	>	6E
21	?	6F
22	:	7A
23	#	7B
24	@	7C
25	' (apostrophe)	7D
26	=	7E
27	"	7F
28	A	C1
29	B	C2
30	C	C3
31	D	C4
32	E	C5
33	F	C6
34	G	C7
35	H	C8
36	I	C9
37	}	D0
38	J	D1
39	K	D2
40	L	D3
41	M	D4
42	N	D5
43	O	D6
44	P	D7
45	Q	D8
46	R	D9
47	S	E2
48	T	E3
49	U	E4
50	V	E5
51	W	E6
52	X	E7
53	Y	E8
54	Z	E9
55	0	F0
56	1	F1
57	2	F2
58	3	F3
59	4	F4
60	5	F5
61	6	F6
62	7	F7
63	8	F8
64 (highest)	9	F9

When Only the Zone Portion of Characters is Used		
Order in the[1] Sequence	Character	Corresponding Hexadecimal Number [2]
1 (lowest)	¢	4A
	.	4B
	<	4C
	(	4D
	+	4E
	|	4F
2	!	5A
	$	5B
	*	5C
	)	5D
	;	5E
	¬	5F
3	/	61
	,	62
	%	6C
	_ (underscore)	6D
	>	6E
	?	6F
4	:	7A
	#	7B
	@	7C
	' (apostrophe)	7D
	=	7E
	"	7F
5	&	50
	A	C1
	B	C2
	C	C3
	D	C4
	E	C5
	F	C6
	G	C7
	H	C8
	I	C9
6	- (minus)	60
	}	D0
	J	D1
	K	D2
	L	D3
	M	D4
	N	D5
	O	D6
	P	D7
	Q	D8
	R	D9
7	S	E2
	T	E3
	U	E4
	V	E5
	W	E6
	X	E7
	Y	E8
	Z	E9
8 (highest)	blank	40
	0	F0
	1	F1
	2	F2
	3	F3
	4	F4
	5	F5
	6	F6
	7	F7
	8	F8
	9	F9

When Only the Digit Portion of Characters is Used		
Order in the[1] Sequence	Character	Corresponding Hexadecimal Number [2]
1 (lowest)	blank	40
	&	50
	- (minus)	60
	}	D0
	0	F0
2	/	61
	A	C1
	J	D1
	1	F1
3	B	C2
	K	D2
	S	E2
	2	F2
4	C	C3
	L	D3
	T	E3
	3	F3
5	D	C4
	M	D4
	U	E4
	4	F4
6	E	C5
	N	D5
	V	E5
	5	F5
7	F	C6
	O	D6
	W	E6
	6	F6
8	G	C7
	P	D7
	X	E7
	7	F7
9	H	C8
	Q	D8
	Y	E8
	8	F8
10	I	C9
	R	D9
	Z	E9
	9	F9
11	¢	4A
	!	5A
	:	7A
12	.	4B
	$	5B
	,	6B
	#	7B
13	<	4C
	*	5C
	%	6C
	@	7C
14	(	4D
	)	5D
	_ (underscore)	6D
	' (apostrophe)	7D
15	+	4E
	;	5E
	>	6E
	=	7E
16 (highest)	|	4F
	¬	5F
	?	6F
	"	7F

[1] When several characters share the same position in the sequence, they are considered equal. For example, if you are using only the digit portions of characters H, Q, Y and 8 (position #9) are considered equal.

[2] This is the number you use in ALTSEQ statements to identify a character that you want to shift to a different order in the sequence.

STANDARD COLLATING SEQUENCE

The Standard EBCDIC Collating Sequence chart shows the three standard collating sequences:

- Standard collating sequence when both zone and digit portions of characters are used.
- Standard collating sequence when only the zone portion of characters is used.
- Standard collating sequence when only the digit portion of characters is used.

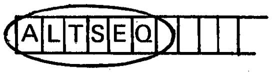
ALTERNATE COLLATING SEQUENCE

If you want to change the standard collating sequence, you must code an S in column 26 of the header specification and code ALTSEQ statements. You can code as many ALTSEQ statements as you need. Each one must begin with ALTSEQ and can contain a maximum of 96 positions.

Note: Do not use a packed or unpacked factor 1 in an include or omit record type specification (P or U in column 8) if you specify an alternate collating sequence.

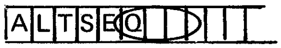
Coding Rules

1. Code ALTSEQ in the first six positions to tell sort you want to alter the standard collating sequence.



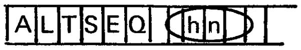
```
ALTSEQ
```

2. Leave the next two positions blank.



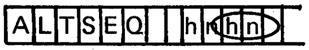
```
ALTSEQ
```

3. Enter the hexadecimal equivalent[1] of the character you are taking out of its normal sequence.



```
ALTSEQ  hn
```

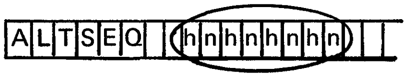
4. Enter the hexadecimal equivalent[1] of the value that the character specified in columns 9 and 10 is to assume in the collating sequence.



```
ALTSEQ  hnhn
```

5. Enter as many pairs of 3 and 4 as the number of characters you are taking out of normal sequence.

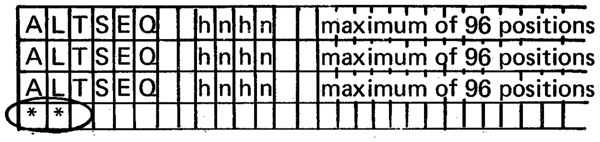
6. Leave no space between sets of hexadecimal numbers.



```
ALTSEQ  hnhnhnhn
```

7. When you reach the end of one statement, you can continue on the next specification line (follow rules 1 through 6).

8. Enter a double asterisk in positions 1 and 2 to indicate the end of the ALTSEQ statements.



```
ALTSEQ  hnhn  maximum of 96 positions
ALTSEQ  hnhn  maximum of 96 positions
ALTSEQ  hnhn  maximum of 96 positions
**
```

[1] See Standard EBCDIC Collating Sequence chart for hexadecimal equivalents of all characters.

Input Order When ALTSEQ Statements Are Used

ALTSEQ statements follow the header line. When you use an alternate collating sequence, your input must be in this order:

1. OCL statements
2. Sequence specifications
   a. Header line
   b. ALTSEQ statements
   c. **
   d. Record type and field lines as required
3. // END

Programming Considerations

When you move a character into the sequence position normally assigned to another character, both the new and the original character occupy the same position and are considered equal. (See the first sample ALTSEQ statement.) If you do not want the two characters to be equal, you must also move the character that normally occupies that position. (See the second ALTSEQ statement.)

Effect of ALTSEQ Statements on Other Coding

ALTSEQ statements can change:

- Factor 1 and factor 2.
- Normal and opposite control fields.
- Control field characters before they are replaced or added to by forced fields.

ALTSEQ statements never change data fields in records or forced control field characters.

Sample ALTSEQ Statements

To Change the Sequence of One Character

ALTSEQ Statements

```
ALTSEQ  40F0
* *
```

Explanation Blank (40) is moved to the position in the collating sequence occupied by zero (F0). Blanks and zeros, therefore, are considered equal.

To Change the Sequence of Several Characters

ALTSEQ Statements

```
ALTSEQ  5BC8C8C9C9D1
* *
```

Explanation 5BC8 moves $ (5B) to the position occupied by H (C8).
C8C9 moves H (C8) to the position occupied by I (C9).
C9D1 moves I (C9) to the position occupied by J (D1).
The $, therefore, has been inserted between G and H, as follows:
A, B, C, D, E, F, G, $, H, I, (I and J occupy the same position).

PROGRAMMING AIDS

You can use a Translation Table and Alternate Collating Sequence Coding Sheet to keep track of changes you make to your collating sequence. A sample sheet is included for your information.

IBM

International Business Machines Corporation

Form GX21-9096
Printed in U.S.A.

TRANSLATION TABLE AND ALTERNATE COLLATING SEQUENCE CODING SHEET

Code	Graphic	Entry	Replaced By/Takes Place Of
00000000		00	
00000001		01	
00000010		02	
00000011		03	
00000100		04	
00000101		05	
00000110		06	
00000111		07	
00001000		08	
00001001		09	
00001010		0A	
00001011		0B	
00001100		0C	
00001101		0D	
00001110		0E	
00001111		0F	
00010000		10	
00010001		11	
00010010		12	
00010011		13	
00010100		14	
00010101		15	
00010110		16	
00010111		17	
00011000		18	
00011001		19	
00011010		1A	
00011011		1B	
00011100		1C	
00011101		1D	
00011110		1E	
00011111		1F	
00100000		20	
00100001		21	
00100010		22	
00100011		23	
00100100		24	
00100101		25	
00100110		26	
00100111		27	
00101000		28	
00101001		29	
00101010		2A	
00101011		2B	
00101100		2C	
00101101		2D	
00101110		2E	
00101111		2F	
00110000		30	
00110001		31	
00110010		32	
00110011		33	
00110100		34	
00110101		35	
00110110		36	
00110111		37	
00111000		38	
00111001		39	
00111010		3A	
00111011		3B	
00111100		3C	
00111101		3D	
00111110		3E	
00111111		3F	
01000000	Blank	40	▨
01000001		41	
01000010		42	
01000011		43	
01000100		44	
01000101		45	
01000110		46	
01000111		47	
01001000		48	
01001001		49	
01001010	¢	4A	▨
01001011	•	4B	▨
01001100	<	4C	▨
01001101	(	4D	▨
01001110	+	4E	▨
01001111	|	4F	▨
01010000	&	50	▨
01010001		51	
01010010		52	
01010011		53	
01010100		54	
01010101		55	
01010110		56	
01010111		57	
01011000		58	
01011001		59	
01011010	!	5A	▨
01011011	$	5B	▨
01011100	*	5C	▨
01011101	)	5D	▨
01011110	;	5E	▨
01011111	¬	5F	▨
01100000	-	60	▨
01100001	/	61	▨
01100010		62	
01100011		63	
01100100		64	
01100101		65	
01100110		66	
01100111		67	
01101000		68	
01101001		69	
01101010		6A	
01101011	,	6B	▨
01101100	%	6C	▨
01101101	_	6D	▨
01101110	>	6E	▨
01101111	?	6F	▨
01110000		70	
01110001		71	
01110010		72	
01110011		73	
01110100		74	
01110101		75	
01110110		76	
01110111		77	
01111000		78	
01111001		79	
01111010	:	7A	▨
01111011	#	7B	▨
01111100	@	7C	▨
01111101	'	7D	▨
01111110	=	7E	▨
01111111	"	7F	▨
10000000		80	
10000001		81	
10000010		82	
10000011		83	
10000100		84	
10000101		85	
10000110		86	
10000111		87	
10001000		88	
10001001		89	
10001010		8A	
10001011		8B	
10001100		8C	
10001101		8D	
10001110		8E	
10001111		8F	
10010000		90	
10010001		91	
10010010		92	
10010011		93	
10010100		94	
10010101		95	
10010110		96	
10010111		97	
10011000		98	
10011001		99	
10011010		9A	
10011011		9B	
10011100		9C	
10011101		9D	
10011110		9E	
10011111		9F	
10100000		A0	
10100001		A1	
10100010		A2	
10100011		A3	
10100100		A4	
10100101		A5	
10100110		A6	
10100111		A7	
10101000		A8	
10101001		A9	
10101010		AA	
10101011		AB	
10101100		AC	
10101101		AD	
10101110		AE	
10101111		AF	
10110000		B0	
10110001		B1	
10110010		B2	
10110011		B3	
10110100		B4	
10110101		B5	
10110110		B6	
10110111		B7	
10111000		B8	
10111001		B9	
10111010		BA	
10111011		BB	
10111100		BC	
10111101		BD	
10111110		BE	
10111111		BF	
11000000		C0	
11000001	A	C1	▨
11000010	B	C2	▨
11000011	C	C3	▨
11000100	D	C4	▨
11000101	E	C5	▨
11000110	F	C6	▨
11000111	G	C7	▨
11001000	H	C8	▨
11001001	I	C9	▨
11001010		CA	
11001011		CB	
11001100		CC	
11001101		CD	
11001110		CE	
11001111		CF	
11010000	}	D0	▨
11010001	J	D1	▨
11010010	K	D2	▨
11010011	L	D3	▨
11010100	M	D4	▨
11010101	N	D5	▨
11010110	O	D6	▨
11010111	P	D7	▨
11011000	Q	D8	▨
11011001	R	D9	▨
11011010		DA	
11011011		DB	
11011100		DC	
11011101		DD	
11011110		DE	
11011111		DF	
11100000		E0	
11100001		E1	
11100010	S	E2	▨
11100011	T	E3	▨
11100100	U	E4	▨
11100101	V	E5	▨
11100110	W	E6	▨
11100111	X	E7	▨
11101000	Y	E8	▨
11101001	Z	E9	▨
11101010		EA	
11101011		EB	
11101100		EC	
11101101		ED	
11101110		EE	
11101111		EF	
11110000	0	F0	▨
11110001	1	F1	▨
11110010	2	F2	▨
11110011	3	F3	▨
11110100	4	F4	▨
11110101	5	F5	▨
11110110	6	F6	▨
11110111	7	F7	▨
11111000	8	F8	▨
11111001	9	F9	▨
11111010		FA	
11111011		FB	
11111100		FC	
11111101		FD	
11111110		FE	
11111111		FF	

□ Hexadecimal numbers of characters in the standard collating sequences.

▨ Where you record the hexadecimal number of the character you are going to put in that relative position in the sequence.

HEADER SPECIFICATION COLUMN SUMMARY

Column Summary

Columns	Entries	Explanation
1-2	00-99	Page number.
3-5	000	Header line number.
6	H	Header line identification.
7-12	SORTA	Addrout sort job.
	SORTR	Tagalong sort job.
	SORTRS	Summary tagalong sort job.
13-17	1-256	Largest sum (in bytes) of control field lengths for any record type.
18	A	Records in sorted file to be in ascending order by control fields.
	D	Records in sorted file to be in descending order by control fields.
	S	Not used in System/32 sort jobs.
19-25	Blank	Not used in System/32 sort jobs.
26	Blank	Use standard collating sequence in compare operations.
	S	Use an alternate collating sequence in compare operations. ALTSEQ statements will define the collating sequence to be used.
27	0 or Blank	Print: Sequence specifications. Diagnostic messages. Program-status messages. Action messages. Formatted messages.
	1	Print: Program-status messages. Action messages. Formatted messages.
	2	Print: Action messages. Formatted messages.
	3	Print nothing (except formatted messages).
28	Blank	Keep control fields in output records in tagalong sort jobs.
	X	Drop control fields from output records in tagalong sort jobs.
29-32	1-4096	Length (in bytes) of output records in tagalong sort jobs.
33	Blank	Not used in System/32 sort jobs.
34	N	Data written on the work file will not be verified.
	Blank or character other than N	Data written on the work file will be verified.
35	Blank	Reserved for system use.
36	Blank	Operator message SORT-7724 will be issued.
	N	Operator message SORT-7724 will not be issued.
37-39	Blank	Not used in System/32 sort jobs.
40-72	Any characters	Comments.

▒ Columns that must be filled in for all sort jobs.

▨ Columns that must be filled in for a tagalong sort.

RECORD TYPE SPECIFICATION COLUMN SUMMARY

Column Summary

Columns	Entries	Explanation
1-2	00-99	Page number.
3-5	01x-06x	Line number. You can leave column 5 (x) blank, or enter any value to keep the specifications in ascending order.
▨ 6	I	Include line.
	O	Omit line.
7	Blank	This is the first line of a set of I *or* O record type lines.
	A	AND lines. These specifications continue the definition of the record described on the preceding line.
	O	OR line. These specifications define a different type of record than the one on the previous line.
	*	Comment line.
▨ 8	C	Use both zone and digit portions of characters.
	Z	Use only zone portion of a 1-character field.
	D	Use only digit portions of characters.
	P	Signed packed decimal data.
	U	Signed unpacked decimal data.
▨ 9-12	1-4096	The input record position in which the factor 1 field begins (blank if field is only one position long).
▨ 13-16	1-4096	The input record position in which the factor 1 field ends.
▨ 17-18	EQ	Factor 1 must equal factor 2.
	NE	Factor 1 must not equal factor 2.
	LT	Factor 1 must be less than factor 2.
	GT	Factor 1 must be greater than factor 2.
	LE	Factor 1 must be less than or equal to factor 2.
	GE	Factor 1 must be greater than or equal to factor 2.
▨ 19	F	Factor 2 is another field in the same input record.
	C	Factor 2 is a constant.
	K	Factor 2 is a keyword.
▨ 20-23	1-4096	The input record position in which the factor 2 field begins (blank if field is only one position long).
▨ 24-27	1-4096	The input record position in which the factor 2 field ends.
▨ 20-39	Any characters	The factor 2 constant.
40-72	Any characters	Comments.

▨ Columns that must be filled in.

FIELD SPECIFICATION COLUMN SUMMARY

Column Summary

Columns	Contents	Explanation
1-2	00-99	Page number.
3-5	07x-14x	Line number. You can leave column 5 (x) blank, or enter any value to keep the specifications in ascending order.
▒ 6	F	Field specification line.
▒ 7	N	Normal control field.
	O	Opposite control field.
	F	Forced control field.
	D	Data field.
	S	Summary data field.
	*	Comment line.
▒ 8	C	Use both zone and digit portions of characters in the field.
	Z	Use only zone portion of 1-character field.
	D	Use only digit portions of characters in the field.
	P	Signed packed decimal data.
	U	Signed unpacked decimal data.
	V	Force a data character into the data field.
▒ 9-12	1-4096	Start position of a field in the record (can be blank if the field is one character long).
▒ 13-16	1-4096	End position of a field in the record.
▨ 17	Any character	Forced control fields; the character you want the program to change. Also used for summary overflow indicator fields; the character to be used for the overflow indicator.
▨ 18	Any character	Forced control fields; the character you want the program to substitute. Also used for summary overflow indicator fields; the character that the overflow field is initialized to.
▨ 19	Blank	Forced control field line is not a continuation of the preceding line.
	Any character other than blank	Forced control field line is a continuation of the preceding line.
20-22	1-256	Overflow field length entry for summary tagalong sort only.
23-39	Blank	Reserved for system use.
40-72	Any characters	Comments.

▒ Columns that must be filled in for all sort jobs.

▨ Columns that must be filled in when forced control fields are being used.

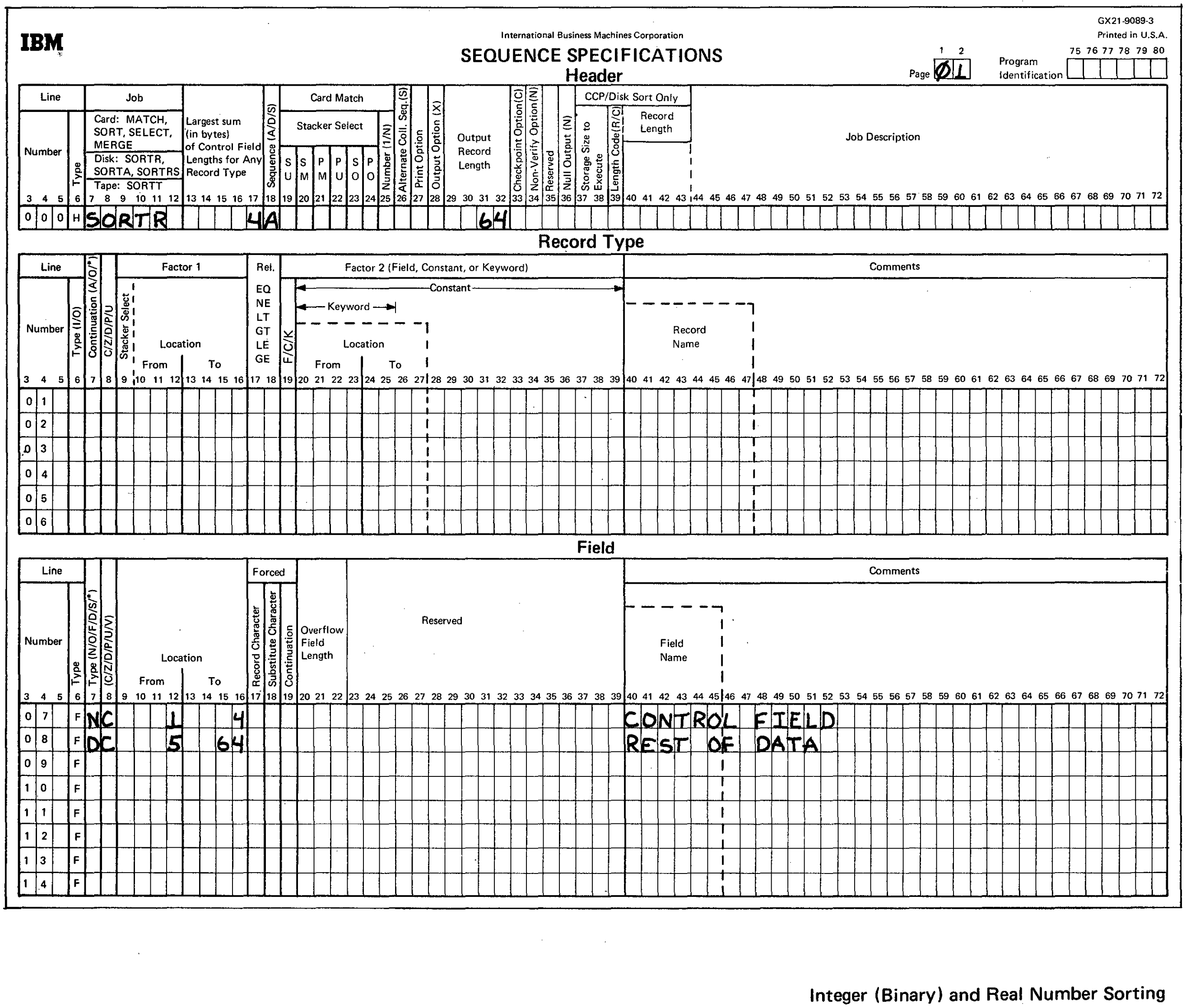
Appendix D. Integer (Binary) and Real Number Sorting

This appendix indicates how either positive and negative integers or positive and negative real numbers in control fields can be used to sort records in a file.

Coded sequence specifications sheets for sample sort jobs are included, followed by a brief discussion of the entries made on the sheets.

Note: For detailed discussion of sequence specifications sheet entries, see Chapters 2, 3, and 4; see also Chapter 7, *Sample Sort Jobs.*

SAMPLE SORT JOB 1. SORTING ON A CONTROL FIELD THAT CONTAINS EITHER A POSITIVE INTEGER OR A POSITIVE REAL NUMBER



IBM

International Business Machines Corporation

SEQUENCE SPECIFICATIONS

GX21-9089-3
Printed in U.S.A.

Page (1 2) Ø1 Program Identification (75 76 77 78 79 80)

Header

Line Number (3 4 5)	Type (6)	Job — Card: MATCH, SORT, SELECT, MERGE; Disk: SORTR, SORTA, SORTRS; Tape: SORTT (7-12)	Largest sum (in bytes) of Control Field Lengths for Any Record Type (13-17)	Sequence (A/D/S) (18)	Card Match — Stacker Select S U M, P M U, S O, P O (19-24)	Number (1/N) (25)	Alternate Coll. Seq.(S) (26)	Print Option (27)	Output Option (X) (28)	Output Record Length (29-32)	Checkpoint Option(C) (33)	Non-Verify Option(N) (34)	Reserved (35)	Null Output (N) (36)	CCP/Disk Sort Only — Storage Size to Execute (37 38)	Length Code(R/C) (39)	Record Length (40-43)	Job Description (44-72)
0 0 0	H	SORTR	4	A						64								

Record Type

Line Number (3 4 5)	Type (I/O) (6)	Continuation (A/O/*) (7)	C/Z/D/P/U (8)	Factor 1 — Stacker Select (9)	Location From (10-12)	To (13-16)	Rel. EQ NE LT GT LE GE (17 18)	F/C/K (19)	Factor 2 (Field, Constant, or Keyword) — Constant; Keyword; Location From (20-23)	To (24-27)	(28-39)	Comments — Record Name (40-47)	(48-72)
0 1													
0 2													
0 3													
0 4													
0 5													
0 6													

Field

Line Number (3 4 5)	Type (6)	Type (N/O/F/D/S/*) (7)	(C/Z/D/P/U/V) (8)	Location From (9-12)	To (13-16)	Forced — Record Character (17)	Substitute Character (18)	Continuation (19)	Overflow Field Length (20-22)	Reserved (23-39)	Comments — Field Name (40-45)	(46-72)
0 7	F	N	C	1	4						CONTROL FIELD	
0 8	F	D	C	5	64						REST OF DATA	
0 9	F											
1 0	F											
1 1	F											
1 2	F											
1 3	F											
1 4	F											

Header Specification

Column 6

H identifies the line as a header line.

Columns 7-12

SORTR identifies the job as a tagalong sort.

Columns 13-17

The control field that contains either a positive integer or a positive real number has a length of 4 (the entry in column 17).

Column 18

The A specifies ascending order. The smallest positive integer or positive real number comes first; the largest positive integer or positive real number comes last in the sort sequence.

Columns 29-32

The control field is not dropped when writing to the output file (column 28 is blank). Therefore, the length of the output record equals the length of the control field (4 bytes) added to the length of the data field (60 characters) specified by lines 07 and 08. This output record length totals 64, which is the entry for columns 29 through 32.

Record Type Specifications

Because all input records are being used and all have the same field specifications, no record type specifications are needed (include-all is implied).

Field Specifications

Column 6

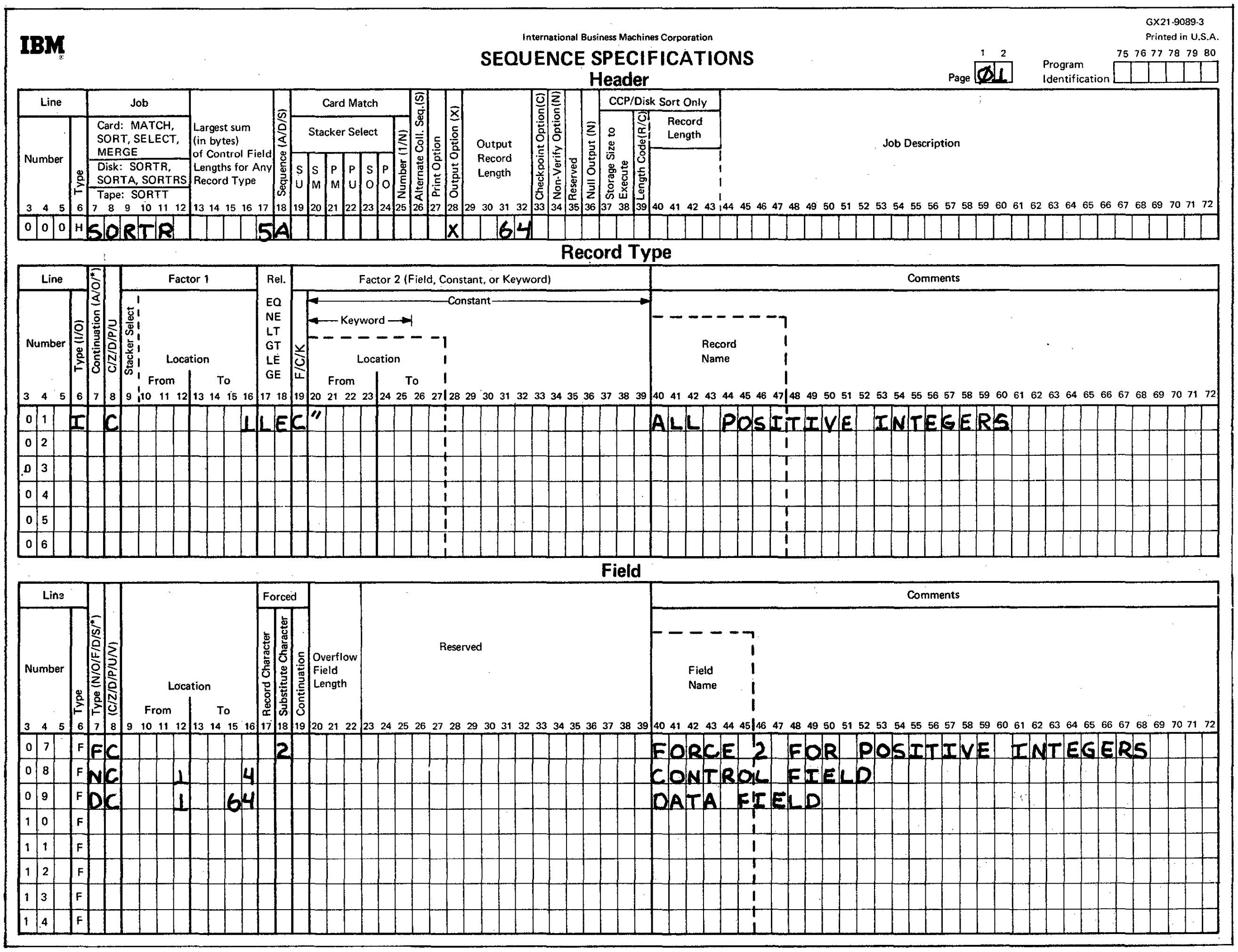
F identifies the lines as field specification lines.

Columns 7, 8, and 9-16

Line 07 indicates that the control field is in positions 1 through 4 of the input record. This control field is included in the sorted output file (column 28 of the header line is blank).

Line 08 indicates that the other 60 input record characters are included as data in the sorted output file.

SAMPLE SORT JOB 2. SORTING ON A CONTROL FIELD THAT CONTAINS ONLY A POSITIVE OR NEGATIVE INTEGER



IBM

International Business Machines Corporation

GX21-9089-3
Printed in U.S.A.

SEQUENCE SPECIFICATIONS

Page (1 2) Ø1 Program Identification (75 76 77 78 79 80)

Header

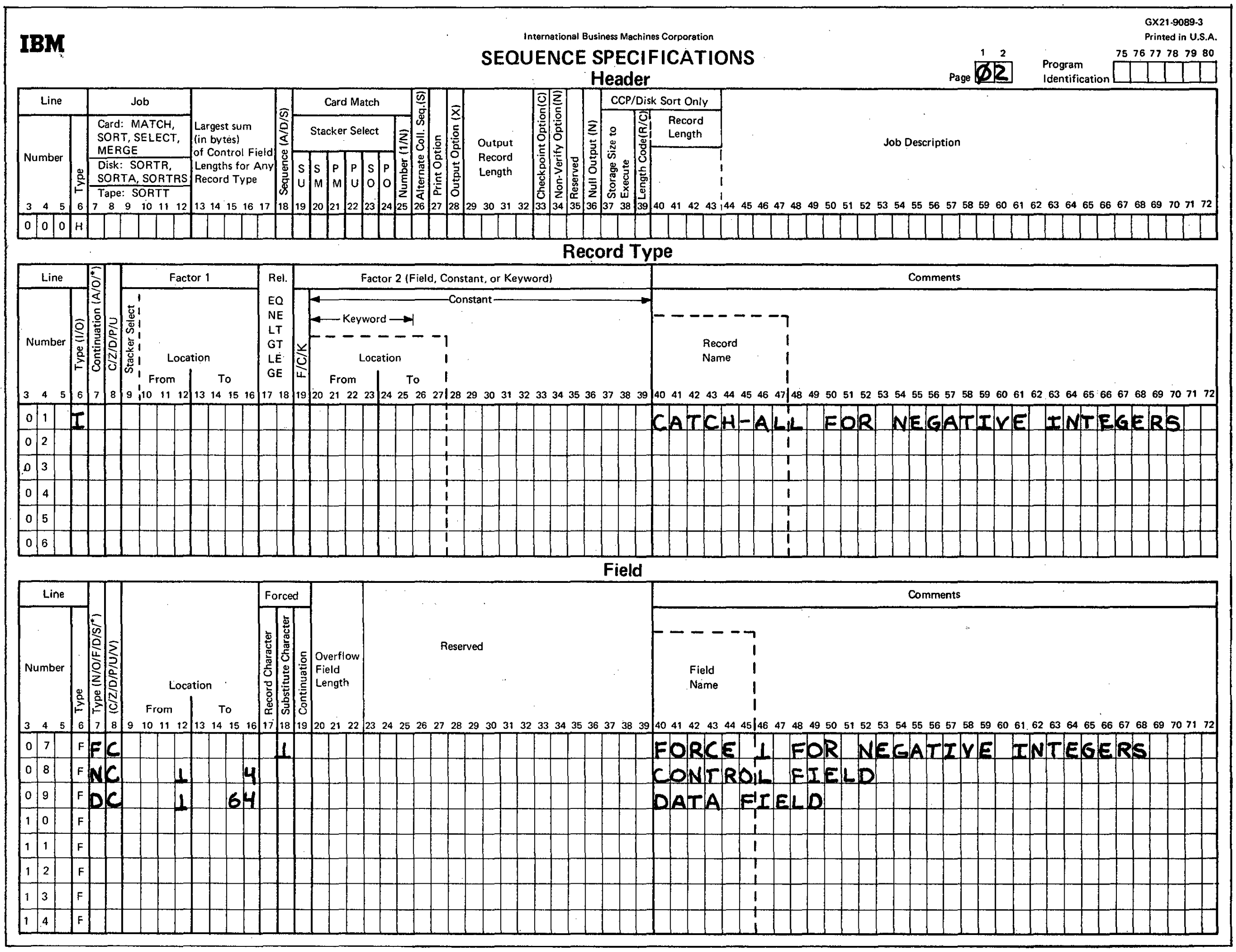
Line Number (3 4 5)	Type (6)	Job (7–12) Card: MATCH, SORT, SELECT, MERGE / Disk: SORTR, SORTA, SORTRS / Tape: SORTT	Largest sum (in bytes) of Control Field Lengths for Any Record Type (13–17)	Sequence (A/D/S) (18)	Card Match: Stacker Select SU SM PM PU SO PO (19–24)	Number (1/N) (25)	Alternate Coll. Seq.(S) (26)	Print Option (27)	Output Option (X) (28)	Output Record Length (29–32)	Checkpoint Option(C) (33)	Non-Verify Option(N) (34)	Reserved (35)	Null Output (N) (36)	CCP/Disk Sort Only: Storage Size to Execute (37 38)	Length Code(R/C) (39)	Record Length (40–43)	Job Description (44–72)
000	H	SORTR	5	A					X	64								

Record Type

Line Number (3 4 5)	Type (I/O) (6)	Continuation (A/O/*) (7)	C/Z/D/P/U (8)	Stacker Select (9)	Factor 1 Location From (10–12)	Factor 1 Location To (13–16)	Rel. EQ NE LT GT LE GE (17 18)	F/C/K (19)	Factor 2 (Field, Constant, or Keyword) Constant / Keyword / Location From (20–23) To (24–27) (20–39)	Comments: Record Name (40–72)
01	I		C			1	LE	C	"	ALL POSITIVE INTEGERS
02										
03										
04										
05										
06										

Field

Line Number (3 4 5)	Type (6)	Type (N/O/F/D/S/*) (7)	(C/Z/D/P/U/V) (8)	Location From (9–12)	Location To (13–16)	Forced Record Character (17)	Forced Substitute Character (18)	Continuation (19)	Overflow Field Length (20–22)	Reserved (23–39)	Comments: Field Name (40–72)
07	F	F	C				2				FORCE 2 FOR POSITIVE INTEGERS
08	F	N	C	1	4						CONTROL FIELD
09	F	D	C	1	64						DATA FIELD
10	F										
11	F										
12	F										
13	F										
14	F										



IBM

International Business Machines Corporation

SEQUENCE SPECIFICATIONS

GX21-9089-3 Printed in U.S.A.

Page Ø2

Program Identification

Header

Line Number	Type	Job	Largest sum	Sequence (A/D/S)	Card Match	Output Record Length	Job Description
000	H						

Record Type

Line Number	Type (I/O)	Factor 1	Rel.	Factor 2	Comments (Record Name)
01	I				CATCH-ALL FOR NEGATIVE INTEGERS
02					
03					
04					
05					
06					

Field

Line Number	Type	Type (N/O/F/D/S/*)	(C/Z/D/P/U/V)	Location From	Location To	Record Character	Substitute Character	Continuation	Overflow Field Length	Comments (Field Name)
07	F	F	C				1			FORCE 1 FOR NEGATIVE INTEGERS
08	F	N	C	1	4					CONTROL FIELD
09	F	D	C	1	64					DATA FIELD
10	F									
11	F									
12	F									
13	F									
14	F									

Header Specification

Column 6

H identifies the line as a header line.

Columns 7-12

SORTR identifies the job as a tagalong sort.

Columns 13-17

The total length of the control field (for either a positive or a negative integer) is 5 bytes (the entry in column 17).

Column 18

The A stands for ascending order. Since negative integers are stored in two's complement form, the smallest negative integer comes first; the largest positive integer comes last in the sort sequence (for example: –2, –1, 0, 1, 2).

Column 28

X indicates that the control field is dropped when writing to the output file.

Columns 29-32

When the control field is dropped, only the data portion remains. The data portion in this sample job is the entire input record. Therefore, 64 is entered in columns 29 through 32.

Record Type Specifications

Column 6

Two record types are specified (positive integers and negative integers). Both record types are to be included (I in column 6 of line 01, pages 01 and 02).

Columns 8, 9-16, 17-18, 19, and 20-39

On line 01 (page 01), these columns specify that if position 1 (the leftmost byte of the 4-byte control field from the input record) is less than or equal to the character constant " (see note), the work record will have a character 2 forced into the first byte of its 5-byte control field (line 07 on page 01). This include line selects only positive integers. Line 01 (page 02), which is an include-all line, indicates that all records not described by the first include line (page 01) are also included in the sort. The first include line (line 01, page 01) selects all positive integers; however, this page 02 include-all line selects all negative integers, which will have a character 1 forced into the first byte of their work record control field (line 07 on page 02). All negative integers are forced in front of all positive integers in the sort sequence.

Note: The special character " has a bit configuration of 01111111, which is equivalent to the high order byte of the maximum positive integer.

Field Specifications

Column 6

F identifies the lines on pages 01 and 02 as field specification lines.

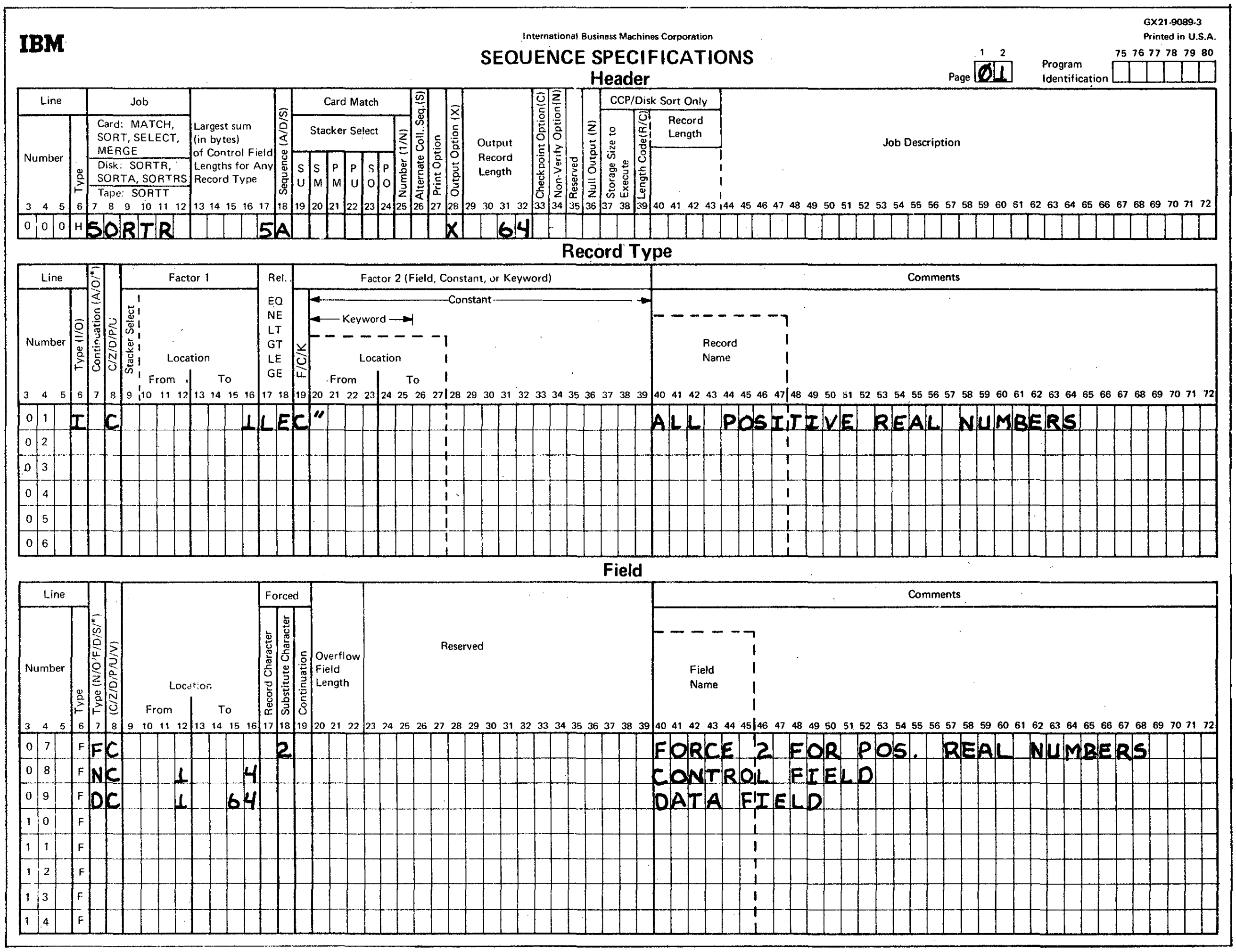
Columns 7, 8, 9-16, and 18

On line 07 (pages 01 and 02), these columns specify that for positive integers (line 07 on page 01), a character 2 will be forced into the first byte of the work record control field; for negative integers (line 07 on page 02), a character 1 will be forced into the first byte of the work record control field.

On line 08 (pages 01 and 02), columns 7, 8, and 9 through 16 specify that the control field is in positions 1 through 4 of the input records. (The control field is not included in the output records because header specification column 28, page 01, contains an X.)

On line 09 (pages 01 and 02), columns 7, 8, and 9 through 16 specify that 64 input record characters are included as data in the sorted output file.

SAMPLE SORT JOB 3. SORTING ON A CONTROL FIELD THAT CONTAINS ONLY A POSITIVE OR NEGATIVE REAL NUMBER



IBM

International Business Machines Corporation

SEQUENCE SPECIFICATIONS

GX21-9089-3
Printed in U.S.A.

Page (1 2) ØL

Program Identification (75 76 77 78 79 80)

Header

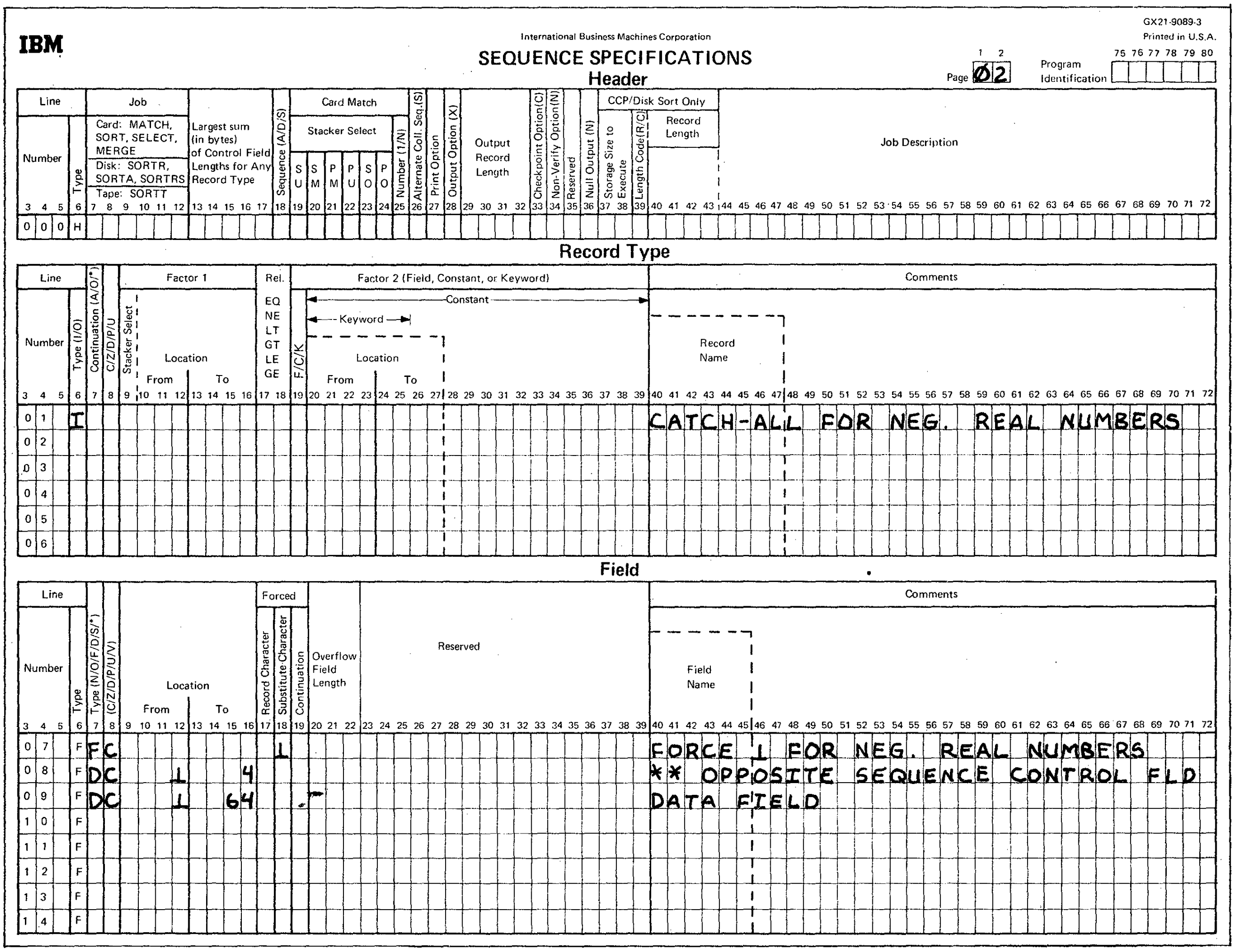
Line Number (3-5)	Type (6)	Job (7-12) Card: MATCH, SORT, SELECT, MERGE; Disk: SORTR, SORTA, SORTRS; Tape: SORTT	Largest sum (in bytes) of Control Field Lengths for Any Record Type (13-17)	Sequence (A/D/S) (18)	Card Match Stacker Select SUM, SMM, PMU, PUO, SOO, PO (19-24)	Number (1/N) (25)	Alternate Coll. Seq. (S) (26)	Print Option (27)	Output Option (X) (28)	Output Record Length (29-32)	Checkpoint Option (C) (33)	Non-Verify Option (N) (34)	Reserved (35)	Null Output (N) (36)	CCP/Disk Sort Only: Storage Size to Execute (37-38)	Length Code (R/C) (39)	Record Length (40-43)	Job Description (44-72)
000	H	SORTR	5	A					X	64								

Record Type

Line Number (3-5)	Type (I/O) (6)	Continuation (A/O/*) (7)	C/Z/D/P/U (8)	Stacker Select (9)	Factor 1 Location From (10-12)	Factor 1 Location To (13-16)	Rel. EQ NE LT GT LE GE (17-18)	F/C/K (19)	Factor 2 (Field, Constant, or Keyword) (20-39)	Comments: Record Name (40-72)
01	I		C			1	LE	C	"	ALL POSITIVE REAL NUMBERS
02										
03										
04										
05										
06										

Field

Line Number (3-5)	Type (6)	Type (N/O/F/D/S/*) (7)	(C/Z/D/P/U/V) (8)	Location From (9-12)	Location To (13-16)	Forced: Record Character (17)	Substitute Character (18)	Continuation (19)	Overflow Field Length (20-22)	Reserved (23-39)	Comments: Field Name (40-72)
07	F	F	C			2					FORCE 2 FOR POS. REAL NUMBERS
08	F	N	C	1	4						CONTROL FIELD
09	F	D	C	1	64						DATA FIELD
10	F										
11	F										
12	F										
13	F										
14	F										



IBM

International Business Machines Corporation

SEQUENCE SPECIFICATIONS

GX21-9089-3
Printed in U.S.A.

Page 02

Program Identification (75 76 77 78 79 80)

Header

Line Number (3 4 5)	Type (6)	Job: Card: MATCH, SORT, SELECT, MERGE; Disk: SORTR, SORTA, SORTRS; Tape: SORTT (7-12)	Largest sum (in bytes) of Control Field Lengths for Any Record Type (13-17)	Sequence (A/D/S) (18)	Card Match: Stacker Select (19-24)	Number (1/N) (25)	Alternate Coll. Seq.(S) (26)	Print Option (27)	Output Option (X) (28)	Output Record Length (29-32)	Checkpoint Option(C) (33)	Non-Verify Option(N) (34)	Reserved (35)	Null Output (N) (36)	CCP/Disk Sort Only: Storage Size to Execute (37-38)	Length Code(R/C) (39)	Record Length (40-43)	Job Description (44-72)
000	H																	

Record Type

Line Number (3 4 5)	Type (I/O) (6)	Continuation (A/O/*) (7)	C/Z/D/P/U (8)	Stacker Select (9)	Factor 1 Location From - To (10-16)	Rel. EQ NE LT GT LE GE (17-18)	F/C/K (19)	Factor 2 (Field, Constant, or Keyword) (20-39)	Comments: Record Name (40-72)
01	I								CATCH-ALL FOR NEG. REAL NUMBERS
02									
03									
04									
05									
06									

Field

Line Number (3 4 5)	Type (6)	Type (N/O/F/D/S/*) (7)	(C/Z/D/P/U/V) (8)	Location From (9-12)	Location To (13-16)	Forced: Record Character (17)	Forced: Substitute Character (18)	Continuation (19)	Overflow Field Length (20-22)	Reserved (23-39)	Comments: Field Name (40-72)
07	F	F	C				1				FORCE 1 FOR NEG. REAL NUMBERS
08	F	D	C	1	4						** OPPOSITE SEQUENCE CONTROL FLD
09	F	D	C	1	64						DATA FIELD
10	F										
11	F										
12	F										
13	F										
14	F										

Header Specification

Column 6

H identifies the line as a header line.

Columns 7-12

SORTR identifies the job as a tagalong sort.

Columns 13-17

The total length of the control field (for either a positive or negative real number) is 5 bytes (the entry in column 17).

Column 18

The A stands for ascending order. The smallest negative real number comes first; the largest positive real number comes last in the sort sequence (for example: –2.0, –1.0, 0.0, 1.0, 2.0).

Column 28

X indicates that the control field is dropped when writing to the output file.

Columns 29-32

When the control field is dropped, only the data portion remains. The data portion in this sample job is the entire input record. Therefore, 64 is entered in columns 29 through 32.

Record Type Specifications

Column 6

Two record types are specified (positive real numbers and negative real numbers). Both types are to be included (I in column 6 of line 01, pages 01 and 02).

Columns 8, 9-16, 17-18, 19, and 20-39

On line 01 (page 01), these columns specify that if position 1 (the leftmost byte of the real number, 4-byte control field from the input record) is less than or equal to the character constant " (see note), the work record will have a character 2 forced into the first byte of its 5-byte control field (line 07 on page 01). This include line selects only positive real numbers. Line 01 (page 02), which is an include-all line, indicates that all records not described by the first include line (page 01) are also included in the sort. The first include line (line 01 on page 01) selects all positive real numbers; however, this page 02 include-all line selects all negative real numbers, which will have a character 1 forced into the first byte of their work record control field (line 07 on page 02). All negative real numbers are forced in front of all positive real numbers in the sort sequence.

Note: The special character " has a bit configuration of 01111111, which is equivalent to the high order byte of the maximum positive real number.

Field Specifications

Column 6

F identifies the lines on pages 01 and 02 as field specification lines.

Columns 7, 8, 9-16, and 18

On line 07 (pages 01 and 02), these columns specify that for positive real numbers (line 07 on page 01), a character 2 will be forced into the first byte of the work record control field; for negative real numbers (line 07 on page 02), a character 1 will be forced into the first byte of the work record control field.

On line 08 (pages 01 and 02), columns 7, 8, and 9 through 16 specify that the control field is in positions 1 through 4 of the input records. Note that real numbers are always carried in true form (never in two's complement form). Therefore, sorting negative real numbers into ascending sequence requires them (when treated as character fields) to be sorted into descending sequence. The O in column 7, line 08 on page 02 indicates an opposite control field, which means that the negative real numbers will be sorted in descending sequence.

On line 09 (pages 01 and 02), columns 7, 8, and 9 through 16 specify that 64 input record characters are included as data in the sorted output file.

SAMPLE SORT JOB 4. RECORD SELECTION BASED ON BINARY VALUES

Records can be selected based on a binary value by using one or both of the following types of include sequences.

Type 1 Include Sequence

For binary values whose hexadecimal equivalents can be represented by one or two characters in the 64-character set (Appendix B, Standard EBCDIC Collating Sequence table), use either an *include character* (I and C in columns 6 and 8 of the record type specifications) or an *include zone* (I and Z in columns 6 and 8 of the record type specifications) ANDed with an *include digit* (IAD in columns 6 through 8 of the record type specifications).

Type 2 Include Sequence

For binary values whose hexadecimal zone portions do not appear in the 64-character set, use two include specifications for the zone portion ANDed with an *include digit* for the digit portion.

Example: Select records that contain a binary value of 44 in position 2. The binary representation of this number is 0010 1100, which is a hex 2C. For the digit portion, select a character from the table in Appendix B with a digit portion of C (for example, <, *, %, and @). Use this character in an include line, comparing to the digit portion only. Since there is no character in the 64-character set whose zone portion is equivalent to 2, use two include lines to describe constants with a zone portion greater than 1 and less than or equal to 2. Describe these constants as packed, which reverses the zone and digit portions from hex F1 and F2 to hex 1F and 2F.

Record Type

Line Number	Type (I/O)	Continuation (A/O/*)	C/Z/D/P/U	Stacker Select	Factor 1 Location From	Factor 1 Location To	Rel. EQ NE LT GT LE GE	F/C/K	Factor 2 (Field, Constant, or Keyword)	Comments (Record Name)
01	I		D			2	EQ	C	@	DIGIT = C
02	I	A	P			2	LE	C	2	AND ≤ HEX 2F
03	I	A	P			2	GT	C	1	AND > HEX 1F
04										
05										
06										

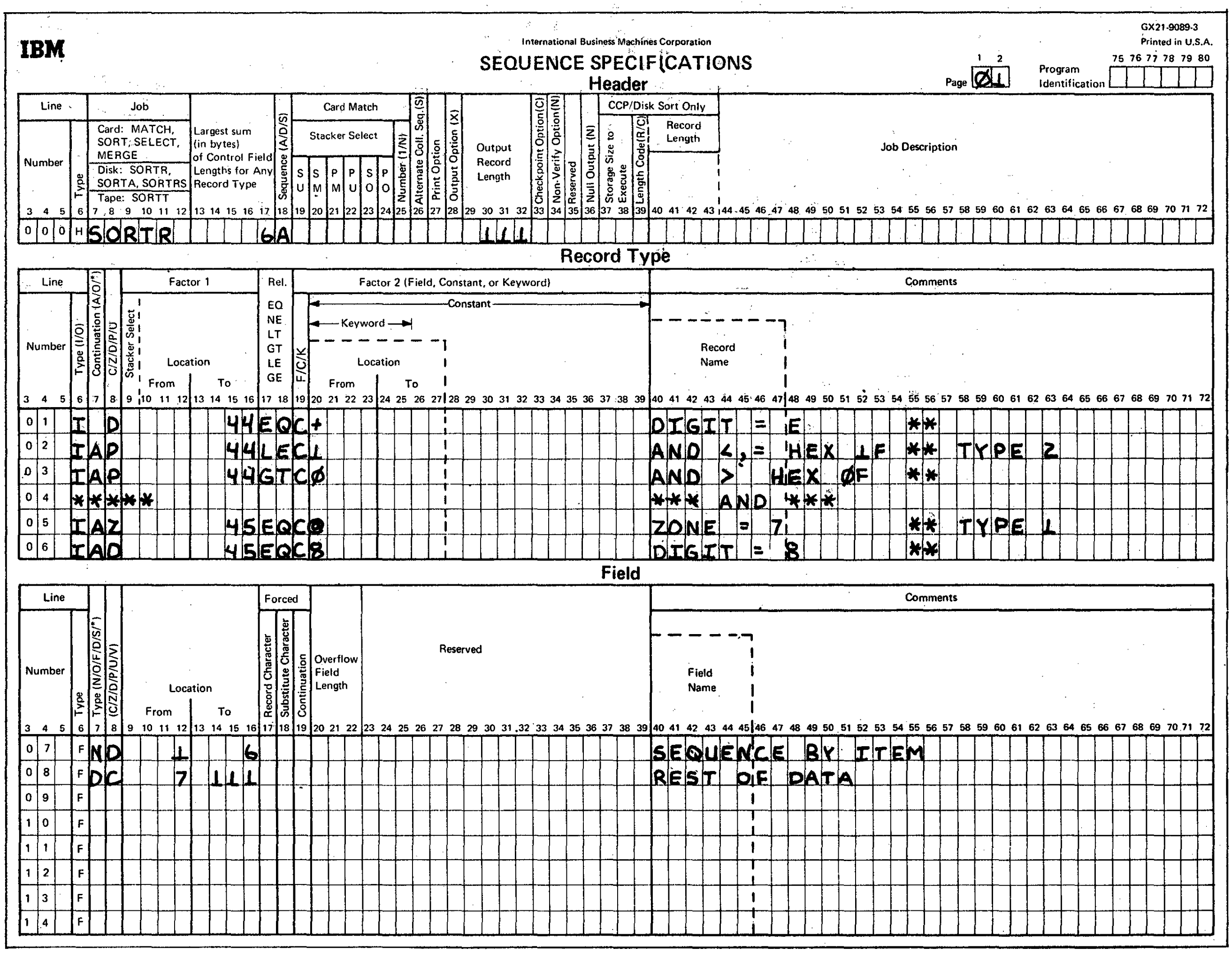
	Hexadecimal Value		Binary Value		Decimal Value
Field to be selected	2C	=	0010 1100	=	44
Character @	7C	=	0111 1100	=	124
Packed character 2	2F	=	0010 1111	=	47
Packed character 1	1F	=	0001 1111	=	31

The record type specifications shown will cause selection of all records containing (in column 2) a value between binary 31 (hex 1F) and binary 47 (hex 2F) whose digit portion is equal to C.

Note: The 32 hexadecimal values whose zone portions are either A or B cannot be accessed with these types of entries. Therefore, there can be no record selection based on these values.

Sample Sort Job 4 Description

The purpose of this sample job is to select records from an inventory file whose reorder point is 7800. The reorder point is a 2-byte quantity in columns 44 and 45 of the inventory records. The quantity is stored in binary (7800 = 0001 1110 0111 1000), and the hexadecimal representation of this number is hex 1E78. (Use the *IBM System/32 and System/34 Basic Assembler Reference Card*, GX21-7674, to help with binary and hexadecimal conversions.)



IBM

International Business Machines Corporation

GX21-9089-3
Printed in U.S.A.

SEQUENCE SPECIFICATIONS

Page Ø1 Program Identification

Header

Line Number (3-5)	Type (6)	Job (7-12) Card: MATCH, SORT, SELECT, MERGE / Disk: SORTR, SORTA, SORTRS / Tape: SORTT	Largest sum (in bytes) of Control Field Lengths for Any Record Type (13-17)	Sequence (A/D/S) (18)	Card Match Stacker Select SU SM PM PU SO PO (19-24)	Number (1/N) (25)	Alternate Coll. Seq.(S) (26)	Print Option (27)	Output Option (X) (28)	Output Record Length (29-32)	Checkpoint Option(C) (33)	Non-Verify Option(N) (34)	Reserved (35)	Null Output (N) (36)	CCP/Disk Sort Only: Storage Size to Execute (37-38)	Length Code(R/C) (39)	Record Length (40-43)	Job Description (44-72)
000	H	SORTR	6	A						111								

Record Type

Line Number (3-5)	Type (I/O) (6)	Continuation (A/O/*) (7)	C/Z/D/P/U (8)	Stacker Select (9)	Factor 1 Location From (10-12)	Factor 1 Location To (13-16)	Rel. EQ NE LT GT LE GE (17-18)	F/C/K (19)	Factor 2 (Field, Constant, or Keyword) (20-39)	Comments: Record Name (40-72)
01	I		D			44	EQ	C	+	DIGIT = E **
02	I	A	P			44	LE	C	1	AND <,= HEX 1F ** TYPE 2
03	I	A	P			44	GT	C	Ø	AND > HEX ØF **
04	*	*	*	*	*					*** AND ***
05	I	A	Z			45	EQ	C	@	ZONE = 7 ** TYPE 1
06	I	A	D			45	EQ	C	8	DIGIT = 8 **

Field

Line Number (3-5)	Type (6)	Type (N/O/F/D/S/*) (7)	(C/Z/D/P/U/V) (8)	Location From (9-12)	Location To (13-16)	Forced Record Character (17)	Forced Substitute Character (18)	Continuation (19)	Overflow Field Length (20-22)	Reserved (23-39)	Comments: Field Name (40-72)
07	F	N	D	1	6						SEQUENCE BY ITEM
08	F	D	C	7	111						REST OF DATA
09	F										
10	F										
11	F										
12	F										
13	F										
14	F										

	Hexadecimal Value		Binary Value		Decimal Value
Field to be selected, column 44	1E	=	0001 1110	=	30
Character +	4E	=	0100 1110	=	78
Packed character 1	1F	=	0001 1111	=	31
Packed character 0	0F	=	0000 1111	=	15
Field to be selected, column 45	78	=	0111 1000		
Character @	7C	=	0111 1100		
Character 8	F8	=	1111 1000		

These specifications will cause selection of all records containing (in column 44) a value between 15 (hex 0F) and 31 (hex 1F) whose digit portion is equal to E. They will further limit selection of records to those which have a binary value in column 45 whose zone portion equals 7 and whose digit portion equals 8. Therefore, all records selected will have hex 1E78 in columns 44 and 45.

Header Specification

Column 6

H identifies the line as a header line.

Columns 7-12

SORTR identifies the job as a tagalong sort.

Columns 13-17

The total length of the control field is 6 bytes.

Column 18

The A stands for ascending order. The file will be sorted according to the standard collating sequence shown in Appendix B.

Columns 29-32

The control field is not dropped when writing to the output file (column 28 is blank). Therefore, the length of the output record equals the length of the control field (6 bytes) plus the length of the data field (105 characters) specified by lines 07 and 08. This output record length totals 111, which is the entry for columns 29 through 32.

Record Type Specifications

Line 01 (I D) selects records whose digit portion in column 44 is E. Lines 02 and 03 select records whose packed value in column 44 is less than or equal to hex 1F, and at the same time is greater than a packed value of hex 0F.

The only hexadecimal values that satisfy both specifications for lines 02 and 03 are hex 11 through hex 1F. Line 04 is a comment line. Lines 05 and 06 select records whose zone portion in column 45 is 7 and whose digit portion is 8.

Field Specifications

Column 6

F identifies the lines as field specification lines.

Columns 7, 8, and 9-16

Line 07 indicates that the control field is in positions 1 through 6 of the input records. Only the digit portions of the characters will be used. This control field will be included in the output file.

Line 08 indicates that the other 105 input record characters are included as data in the output file.

Appendix E. Sort Programmer Messages

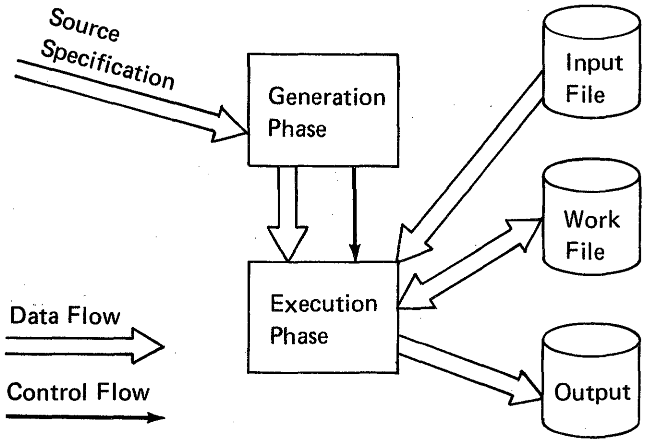
Sort programmer messages are activated (or suppressed) by the print option in the header statement. These messages, if activated, are displayed (only a portion of the message actually appears on the display screen), and they are printed if SYSLOG is assigned to the printer. Generally, SYSLOG should be assigned to the printer if sort programmer messages are activated.

Sort programmer messages are issued mainly during the generation phase of the sort program. No operator action is required. Therefore, each message description in this section consists only of a simple explanation for the programmer.

To understand these messages, you must have a basic understanding of how the sort program works. Specifically, it is helpful to know what happens during the generation phase and during the execution phase.

The generation phase translates your sequence specifications into machine language.

The execution phase gets the records you want sorted, sorts the records, and writes the sorted records onto the output file in the order that you want them.



Following is a typical sort diagnostic message as it appears on a listing:

```
SORT-7264   W   SPECIFICATION IN WRONG
                NUMERICAL ORDER
```

Note that a one-character alphabetic code, which indicates the significance of the message, follows the message identification code.

Code	Meaning	Significance
T	Terminal	Processing cannot continue. The program ends immediately and no more processing is done. Sort operator message SORT-7725 follows and the operator must cancel the job. For an explanation of the sort operator messages see *Sort Operator Messages* in the *IBM System/32 Displayed Messages Guide,* GC21-7704.
S	Severe	A sort sequence specification statement contains an error. When the generation phase is complete, sort operator message SORT-7725 is issued and the operator must cancel the job. For an explanation of the sort operator messages see *Sort Operator Messages* in the *IBM System/32 Displayed Messages Guide,* GC21-7704.
W	Warning	An error condition might exist. Corrective action is required if the condition is unintentional. When the generation phase is complete, sort operator message SORT-7722 is issued. For an explanation of the sort operator messages see *Sort Operator Messages* in the *IBM System/32 Displayed Messages Guide,* GC21-7704.
A	Action	A condition exists that requires an operator response. A sort operator message follows. For an explanation of the sort operator messages see *Sort Operator Messages* in the *IBM System/32 Displayed Messages Guide,* GC21-7704.
I	Information	For information only. Messages with an I code indicate the current status of the job.

SORT-7101 I PRINT OPTION ASSUMED TO BE 0

The print option in the header line is not blank, 0, 1, 2, or 3. Therefore, it is assumed to be 0.

SORT-7102 W SORT JOB ASSUMED TO BE SORTR

Neither SORTA, SORTR, or SORTRS job type is specified on the header line. Becuase an output record length is specified, SORTR is assumed. To avoid this error, correct columns 7 through 12 of the header line.

SORT-7103 T SORT JOB TYPE NOT SPECIFIED

Neither a job type nor an output record length is specified in the header line. Place the proper job type in columns 7 through 12 of the header line. Resubmit the job.

SORT-7105 W INVALID ENTRY, COLUMN 36

Column 36 of the header line does not contain an N or a blank. A blank is assumed. To avoid this error next time the job is run, specify either N or blank in column 36 of the header line.

SORT-7108 T INPUT/OUTPUT FILE STATEMENT MISSING

The // FILE statements for the input file, the output file, or both are missing for this job. Include the missing // FILE statement; then resubmit the job.

SORT-7150 S CONTROL FIELD DROPPED—NO DATA SPECIFICATIONS

No data specifications (including summary specifications) were entered for this job, but the control field is being dropped. If the job were allowed to continue, the output file would contain all blanks. Add data field specifications to the job and/or retain the control field; then resubmit the job.

SORT-7151 T CONTROL FIELD LENGTH NOT GIVEN

No control field length is specified on the header line for this job. Place the length of your control fields in columns 13 through 17 of the header line; then resubmit the job.

SORT-7152 S CONTROL FIELD LENGTH TOO LARGE

The length specified for the control field is too large. A length of 256 is assumed. Specify a control field length of 256 or less in columns 13 through 17 of the header line; then resubmit the job.

SORT-7153 T INPUT RECORD LENGTH NOT GIVEN

The record length for the input file cannot be determined from the information stored on the disk. If this error occurs often, contact IBM for program support services.

SORT-7154 W ASCENDING SEQUENCE ASSUMED A, COLUMN 18

Column 18 of the header line contains neither D, for descending sequence, nor A, for ascending sequence. Ascending sequence is assumed. To avoid this error next time the job is run, specify either A or D in column 18 of the header line.

SORT-7155 W OUTPUT OPTION ASSUMED X, COLUMN 28

Column 28 of the header line contains neither X nor blank. Output option X is assumed. To avoid this error next time the job is run, specify either X or blank in column 28 of the header line.

SORT-7156 S OUTPUT RECORD LENGTH NOT SPECIFIED

This is a SORTR or SORTRS job, but no output record length is specified on the header line. The maximum length of 4096 is assumed. Specify the length of the output records in columns 29 through 32 of the header line; then resubmit the job.

SORT-7157 S OUTPUT RECORD LENGTH TOO LARGE

This is a SORTR or SORTRS job. The output record length specified in columns 29 through 32 of the header line is larger than 4096. The maximum length of 4096 is assumed. Be sure the header line is not shifted by one or more columns. Specify an output record length of 4096 or less; then resubmit the job.

SORT-7158 S OUTPUT RECORD LENGTH INCONSISTENT

This is a SORTR or SORTRS job and you are not dropping the control field. The output record length (columns 29 through 32) that includes the control field should not be less than the length of the control field (columns 13 through 17). Be sure you do not want to drop the control field. Recompute the output record length; correct the entry on the header line; then resubmit the job.

SORT-7159 S WORK RECORD LENGTH TOO LARGE

The work record length for this job is too large. For information on how to determine the maximum work record length, see Appendix A, *Calculating the Sizes of Files for Sort.*

SORT-7160 W ALTERNATE COLLATING SEQUENCE ASSUMED

Column 26 of the header line does not contain an S or a blank. Alternate collating sequence is assumed. If alternate collating sequence is not to be used, specify a blank in column 26 of the header line; then resubmit the job. If alternate collating sequence is correct, place an S in column 26 to avoid this occurring again.

SORT-7161 T INVALID CONTROL FIELD LENGTH

A nonnumeric character or an embedded blank was specified in columns 13 through 17 of the header line. Correct the entry, making sure that no nonnumeric characters or embedded blanks are specified; then resubmit the job.

SORT-7162 T INVALID OUTPUT RECORD LENGTH

A nonnumeric character or an embedded blank was specified in columns 29 through 32 of the header line. Correct the entry, making sure that no nonnumeric characters or embedded blanks are specified; then resubmit the job.

SORT-7176 S INVALID ALTERNATE SEQUENCE STATEMENT

Column 26 of the header line specifies that an alternate sequence is to be used, but the statement just read is not a comment statement or a valid sort specification statement, nor does it contain ALTSEQƀƀ in columns 1 through 8. Correct the incorrect statement or delete it if it does not belong in the job; or add a valid alternate sequence statement.

SORT-7177 W MISSING THE DOUBLE ASTERISK (**) STATEMENT

No statement containing asterisks (*) in columns 1 and 2 was between the alternate sequence statement and the include, omit, or field specification statement. The sort program assumes an ** statement following the alternate sequence statement. To avoid this error next time the job is run, place an ** statement between the alternate sequence statement and the include, omit, or field specification statement.

SORT-7178 S ALTSEQ STATEMENT HAS INVALID DATA

An alternate sequence statement entry that is used to alter the normal sequence is in error. One of the four-position groups (columns 9 through 12, 13 through 16, and so on) in the ALTSEQ statement does not contain a valid hex character (0 through 9 or A through F). An asterisk (*) flags the columns in error on the source listing. Processing continues with the next four-position group. Correct the incorrect alternate sequence statement; then resubmit the job.

SORT-7179 T UNEXPECTED END OF FILE FOUND

An end-of-file statement followed the alternate sequence statement. No comment statement or ** statement exists. Check that the last part of the sequence specification is not missing from the job or that an end-of-file statement is not placed incorrectly in the job.

SORT-7181 W NO VALID ALTSEQ STATEMENTS FOUND

ALTSEQ is specified on the header line, but no valid ALTSEQ statements were entered. The sort program ignores the ALTSEQ specification on the header line. If an alternate collating sequence is to be used, correct the invalid ALTSEQ statements, or add them if none were specified; then resubmit the job. If alternate collating sequence is not to be used, leave column 26 of the header line blank to avoid this message.

SORT-7202 S INVALID SPECIFICATION TYPE, COLUMN 6

A specification is not identified in column 6 as a valid type of specification. Valid specification types are comments, include sets, omit sets, or field specifications. The specification is bypassed and the rest of the specifications are processed. The specification was probably specified incorrectly or misplaced in the job. Correct the error; then resubmit the job.

SORT-7204 W BLANK CONTINUATION ASSUMED, COLUMN 7

A nonblank character is specified in column 7 of a specification that is the first in an omit or include set; a blank is expected. The sort program assumes a blank in column 7. To avoid this message, leave column 7 blank.

SORT-7206 S INVALID CONTINUATION, COLUMN 7

A character other than A, O, or blank is specified in column 7 of a specification that is part of an omit or include set. The specification is ignored and the next specification is processed. Correct the specification by placing an A or O in column 7; then resubmit the job.

SORT-7208 W OR CONTINUATION ASSUMED, COLUMN 7

A blank is specified in column 7 of a statement that is part of an omit or include set. Column 7 should be A or O. O is assumed. If O is incorrect, enter an A in column 7; then resubmit the job. If O is correct, place an O in column 7 to avoid this message.

SORT-7210 W INVALID C/Z/D/P/U/V SPECIFICATION, COLUMN 8

An invalid entry is specified in column 8 of a specification statement. The valid entries are:

Specification Type	Column 8 Entry
Include or omit record type	C, Z, D, P, or U
Control field	C, Z, D, P, or U
Data field or summary field	C, Z, D, P, U, or V

The sort program assumes a C in column 8. If C is incorrect, place the correct entry in column 8; then resubmit the job. If C is correct, place a C in column 8 to avoid this message.

Note: When the specification type is an include or omit record type and factor 2 is a keyword, column 8 must contain a C. When the specification type is a control field and you want an unconditional force or a force-all line, column 8 must contain a C.

SORT-7212 S SPECIFICATION IN WRONG LOGICAL ORDER

The specification in error is either:

– A control field specification (F in column 6) that incorrectly follows an omit specification (O in column 6).

– An omit specification (O in column 6) that incorrectly follows an include specification (I in column 6).

The specification is bypassed and the next specification is processed. Put the specifications in the correct order, then resubmit the job.

SORT-7214 S INCLUDE OR OMIT STATEMENT AFTER INCLUDE-ALL

An INCLUDE-ALL specification was read for this job, but an include or omit specification (I or O in column 6) was found. Only control field specifications should follow the INCLUDE-ALL specification. The specifications are processed as though the INCLUDE-ALL specification was not read. Remove either the INCLUDE-ALL specification or the include or omit specifications; then resubmit the job.

SORT-7216 S LAST SET NOT A VALID INCLUDE SET

The last set of sequence specifications should be an include set, but one of the following occurred:

– No specifications were specified after the header line.

– The last set was an omit set.

– The last set was an include set that contained no field specifications.

Make one of the following corrections, depending on the condition that caused the message; then resubmit the job:

– Supply the sequence specifications for the job.

– Eliminate the omit set; the records are omitted by default.

– Supply the missing field specifications.

SORT-7218 S TOO MANY SEQUENCE SPECIFICATIONS

All the main storage allocated by the sort program for the select/build routine and for error information was used, but there are still specifications to be processed. The remaining specifications are checked for errors. Remove all causes for error messages from the specifications; then resubmit the job. If this error occurs again and no other error messages are issued, you must decrease the number of sequence specifications.

SORT-7220 T TOO MANY ERRORS

The main storage allocated by the sort program for the select/build routine and for error information was filled with error information. No more specifications can be processed. Correct all diagnosed errors; then resubmit the job.

SORT-7222 W ZONE OR V FIELD LENGTH EXCEEDS 1

The referenced specification is a field specification (F in column 6). The type specified is a zone field (Z in column 8) or a V field (V in column 8) that should have a length of one byte.

The specified number of bytes (calculated using columns 9 through 12 and 13 through 16) is used in building the work record field for this record type. The byte zone or overflow information generated is placed in the rightmost byte (low-order byte) of this field. The other positions to the left contain binary zeros (control fields) or blanks (data fields). If a one-byte zone or V field is incorrect, correct the field specification; then resubmit the job.

SORT-7224 S COLUMNS 9-16 OR COLUMNS 20-27 ARE INVALID

The location information (columns 9 through 16 and/or columns 20 through 27) in a specification is invalid because:

- The From location is greater than the To location.
- The To location is 0 or blank.
- The lengths of factor 1 and factor 2 are not equal.

Correct the error; then resubmit the job.

SORT-7225 S INVALID KEYWORD BEGINNING IN COLUMN 20

The specification causing this error is an include or omit record type specification. A K was found in column 19, but neither UDATE, UMONTH, UDAY nor UYEAR was found beginning in column 20. UDATE is assumed. Correct the specification; then resubmit the job.

SORT-7226 W DIGIT FIELD LENGTH EXCEEDS 16

A digit field length of more than 16 was specified in the specifications. A length of 16 is assumed. If the digit field length is actually 16 or less, correct the specification. If the digit field length is greater than 16, divide it into two different fields and create new specifications for these fields; then resubmit the job.

SORT-7228 W CHARACTER CONSTANT LENGTH EXCEEDS 20

The specification causing this error is an include or omit specification (I or O in column 6). Since a C is specified in columns 19, factor 2 is a character constant. The length implied by factor 1 (columns 9 through 12 and 13 through 16) is greater than 20, the maximum allowed. The 20 rightmost characters in factor 1 are compared against the constant in columns 20 through 39 for this job. Correct the specification statement if a 20-character field is not correct or to avoid this message.

SORT-7230 W SAME SPECIFICATION TYPE ASSUMED, COLUMN 6

The specification causing this message has a blank in column 6, so the specification type is not known. However, column 7 contains an A or O and the previous specification was an include or omit specification (I or O in column 6). Therefore, the sort program assumes that this specification is a record type specification and that the specification type is to be identical to that in the previous specification. If the assumption is incorrect, correct the specification; then resubmit the job.

SORT-7232 S ZONE SPECIFIED-FACTOR 2 IS NOT A CONSTANT

A zone is specified in column 8 or an include or omit record type specification, but column 19 does not have a C specified. The specification is bypassed. Correct the specification; then resubmit the job.

SORT-7234 W ZONE SPECIFIED-RELATIONSHIP NOT EQ OR NE

A zone is specified in column 8 of the include or omit record type specification, but columns 17 and 18 do not contain EQ or NE. EQ is assumed and processing of the specification continues. If EQ is not the correct entry for columns 17 and 18, correct the specification, then resubmit the job. If EQ is correct, correct the specification to avoid this message.

SORT-7236 S INVALID RELATIONSHIP, COLUMNS 17-18

The specification causing this error is an include or omit record type specification. The relationship between factor 1 and factor 2, specified in columns 17 and 18, is not EQ, NE, LT, GT, LE, or GE. EQ is assumed. Correct the specification; then resubmit the job.

SORT-7238 W INVALID FACTOR 2 TYPE, COLUMN 19

Column 19 of an include or omit record type specification does not contain an F (field), a C (constant), or a K (keyword). C is assumed; factor 2 is processed as a constant. Check column 19 for a missing or invalid entry. Correct the specification before the job is run again.

SORT-7240 S UNPACKED DECIMAL-LENGTH EXCEEDS 16

The specification causing this error contains a U in column 8 indicating unpacked decimal. Columns 8 through 12 and 13 through 16 define a factor 1 field that is longer than the allowable 16 bytes. A length of 16 is assumed. Check the To and From columns to determine the error. Correct the error before the job is run again.

SORT-7242 W ZONE SPECIFIED-LENGTH EXCEEDS 1

The specification causing this error is an include or omit record type specification that has a Z in column 8. In this case, factor 1 must be one byte long, but it is not. A length of one is assumed and the constant in column 20 of the specification is processed. Correct the specification before the job is run again.

SORT-7244 W PACKED DECIMAL-LENGTH EXCEEDS 8

A packed field is specified (P in column 8), but the field length is more than 8. A packed field can be only 8 columns long. A field length of 8 is assumed and the rightmost eight bytes of the field are used for the include or omit. If column 19 contains a C, the constant in columns 20 through 35 of the specification is used as factor 2. Correct the specification before the job is run again.

SORT-7246 W INVALID FIELD TYPE SPECIFICATION, COLUMN 7

The specification causing this error is a field specification, but column 7 does not contain an F, N, O, D, or S. An entry of N (normal control field) is assumed if there are no previous data or summary data field specifications for this include set. Otherwise, an entry of D (data field) is assumed. Correct the specification before the job is run again.

SORT-7248 S ILLEGAL FORCE-ALL CONTINUATION

The specification causing this error is a field specification for a force-all specification (F in column 7). This specification should follow a conditional force specification, but it does not. The specification is ignored. Correct the specification; then resubmit the job.

SORT-7250 S FIELD STATEMENTS IN WRONG LOGICAL ORDER

The specification causing this error is a control field specification that follows a data (D) or summary (S) specification in this include set. All control field specifications (N, O, or F in column 7) must precede any data or summary data field specifications. This specification is bypassed and the next specification is processed. The specification might be misplaced or specified incorrectly. Correct the specification; then resubmit the job.

SORT-7252 S CONTROL FIELD LENGTH EXCEEDS HEADER VALUE

The accumulated total of the control field lengths in this include set exceeds the value specified in the header line. The remaining specifications in the include set are processed. Check the length specified in the header line against the specified control fields. Correct the errors; then resubmit the job.

SORT-7254 W CONTROL FIELD LENGTH LESS THAN HEADER VALUE

All control field specifications for this include set were processed, and the accumulated length of control fields is less than the value specified in the header line. The remaining positions are filled with hex zeros. Check the length specified in the header line against the specified control fields. Correct any errors before the job is run again.

SORT-7256 S CHARACTER FIELD SPECIFICATION LENGTH EXCEEDS 256

A length greater than 256 is specified in a character field specification (C in column 8). The rest of the sequence specifications are processed. Check the character field specifications. If you require more than 256 positions of data, divide the field into two or more field specification statements. Correct the error; then resubmit the job.

SORT-7258 S TO OR FROM FIELD NOT WITHIN INPUT RECORD

Either one or both of the specifications in the To or From columns are zero or contain a value larger than the input record length. Make sure the correct input file was specified and the To and From columns contain the correct values. Correct the error; then resubmit the job.

SORT-7262 S DATA LENGTH EXCEEDS HEADER VALUE

A SORTR or SORTRS job is being processed and the data field specifications (FD or FS in columns 6 and 7) for this include set exceed the value in the header line. An error exists in either the output record length in the header line or in the To and From fields in the field description lines. If you are dropping the control field (X in column 28 of the header line), the data length is the output record length (columns 29 through 32 of the header line). If you are not dropping the control field, the control field length (columns 13 through 17 of the header line) must be subtracted from the output record length to determine the data field length. Correct the error; then resubmit the job.

SORT-7264 W SPECIFICATION IN WRONG NUMERICAL ORDER

Columns 1 through 5 of each sequence specification statement are used to order the specifications. One or more of the specification statements in this job has a value in columns 1 through 5 that is less than the value in the preceding specification statement. On the source listing, an S is printed to the left of the specification statements that are out of order. To avoid this error, correct the entries in columns 1 through 5 of the sequence specifications so that no entry is less than that on the previous specification statement.

SORT-7266 S FACTOR 1 LENGTH EXCEEDS 256

The specification causing this error is an include or omit specification for character data (C in column 8). Factor 2 is a field (F in column 19) and the length of factor 1 is greater than 256, the maximum length. The sort program assumes a length of 256 for factor 1. If the length of factor 1 must be more than 256, divide this specification into two specifications with an AND continuation line (A in column 7); then resubmit the job.

SORT-7268 W P OR U FACTOR 1 USED WITH ALTSEQ

An include or omit record type specification specifies a packed or unpacked factor 1, but the header line specifies an alternate collating sequence. The sort program changes factor 1 and factor 2 as the ALTSEQ specifications indicate. This change can affect the units position (and the sign) of an unpacked decimal number or any one position of a packed decimal number. As a result, the desired records might not be included or omitted. Do not specify P or U in column 8 of the record type entries when you specify an alternate collating sequence.

SORT-7276 W MORE THAN ONE SUMMARY V FIELD IN INCLUDE SET

This is a SORTRS job. The specification causing the error specifies a summary V field (FSV in columns 6 through 8), but at least one other summary V field was already specified for this include set. The statement in error is assumed to specify a data V field (FDV in columns 6 through 8). To avoid this error, change the extra summary V fields to data V fields.

SORT-7278 W SUMMARY V OR DATA FIELD INCONSISTENT

This is a SORTRS job. The specification causing the error is a summary data field specification (S in column 7). To determine which of the following conditions is true, compare this specification with the first include set having S specifications:

- The number of S specifications prior to this specification differs.
- The length of this S specification differs.
- The substitution character of the V specification differs.
- The relative location of the specification within the output record differs.

The S specifications of the first include set containing S specifications define the summary field positions. Correct the specifications; then resubmit the job.

SORT-7280 W NUMBER OF SUMMARY FIELDS INCONSISTENT

This is a SORTRS job. The number of S specifications in this include set is different from the number of S specifications in the first include set. The S specifications of the first include set with S specifications form the summary output record. To avoid this error, correct the include sets so they all have the same number of S specifications.

SORT-7282 W NO SUMMARY SPECIFICATIONS—SUMMARY SORT

The header line specifies that this is a SORTRS job, but there are no summary data field specifications. All records with duplicate control fields are eliminated from the output file. Check the job type to be sure that a SORTRS job was intended.

SORT-7284 W SUMMARY SPECIFICATIONS IGNORED

This is a SORTR job; consequently, no summary data field specifications (FS in columns 6 and 7) were expected. However, at least one was found. All S specifications are treated as data specifications (D in column 7). Make sure this should be a SORTR job. If not, correct the specifications; then resubmit the job.

SORT-7286 S TOO MANY SUMMARY DATA FIELDS

This is a SORTRS job. More than 24 summary data fields were specified (FS in columns 6 and 7 and C, D, P, or U in column 8). Limit the number of summary data fields to 24 or less.

SORT-7288 W INVALID SUMMARY SPECIFICATION–COLUMNS 20, 21, 22

The overflow field length in columns 20 through 22 is incorrect because it is:

- An invalid entry
- Not right-justified
- Less than the field length given by the To and From entries

The entry is assumed blank. If a valid entry was intended in columns 20 through 22, correct the entry; then resubmit the job.

SORT-7391 T MAIN STORAGE TOO SMALL

The amount of main storage assigned to the sort program is not large enough to process this job. The job cannot be run. Make sure:

- The correct input file was specified.
- If this is a SORTR or SORTRS job, the output length specified in the header line is correct.
- The total number of bytes reserved for the sequence specifications does not exceed the recommended amount (see Chapter 6, *Performance Considerations*).
- There are no entries in columns 35 through 39 of the header line.
- Correct the errors; then resubmit the job.

SORT-7395 T NOT ENOUGH SPACE FOR WORK FILE

The sort program was attempting to automatically allocate space for the work file, but not enough space is available on the disk. Make space available for the work file by reorganizing the disk or by deleting any unnecessary files and then reorganizing the disk. Use the COMPRESS procedure to reorganize the disk. This places all the unused space in one contiguous area on the disk. You can use the SAVE procedure to save copies of files being deleted. Delete files by using the DELETE procedure. You can use the CATALOG procedure to display the present disk VTOC. For further information about these procedures, see the *IBM System/32 System Control Programming Reference Manual,* GC21-7593.

SORT-7398 T WORK FILE SIZE NOT USABLE

The work file size specified on the // FILE NAME-WORK statement is not large enough. Increase the specified work file size; then resubmit the job.

SORT-7401 I JOB COMPLETED GENERATION PHASE

The generation phase is complete. All statements were checked for errors. If severe or terminal errors were found during generation, sort operator message SORT-7725 is displayed; if only warning errors were found, sort operator message SORT-7722 is displayed. If no errors were found, the job proceeds to the execution phase. For an explanation of the sort operator messages, see *Sort Operator Messages* in the *IBM System/32 Displayed Messages Guide,* GC21-7704.

SORT-7402 I NO ERRORS FOUND

No errors were found during the generation phase of the job. Processing continues with the execution phase.

SORT-7403 I OPERATOR CHOSE TO CANCEL JOB

At the end of the generation phase, sort operator message SORT-7722 was displayed and the operator canceled the job. For an explanation of the sort operator messages, see *Sort Operator Messages* in the *IBM System/32 Displayed Messages Guide,* GC21-7704.

SORT-7404 I OPERATOR CHOSE TO CONTINUE JOB

At the end of the generation phase, sort operator message SORT-7722 was displayed and the operator continued the job. For an explanation of the sort operator messages, see *Sort Operator Messages* in the *IBM System/32 Displayed Messages Guide,* GC21-7704.

SORT-7422 A XXX WARNING ERRORS HAVE BEEN FOUND

During the generation of this job, no severe or terminal errors were found. However, a number of warning errors were found that could affect the execution of the job. Statements entered or specified incorrectly probably caused the errors. Sort operator message SORT-7722 is issued after this message. For an explanation of the sort operator messages, see *Sort Operator Messages* in the *IBM System/32 Displayed Messages Guide,* GC21-7704.

SORT-7423 A PERMANENT ERROR READING SPECIFICATIONS

A disk error occurred while the sequence specifications were being read. Sort operator message SORT-7723 is issued after this message. For an explanation of the sort operator messages, see *Sort Operator Messages* in the *IBM System/32 Displayed Messages Guide,* GC21-7704.

SORT-7425 A SEVERE AND/OR TERMINAL ERRORS FOUND

Severe and/or terminal errors occurred during the generation of this job. Sort operator message SORT-7725 is issued after this message. For an explanation of the sort operator messages, see *Sort Operator Messages* in the *IBM System/32 Displayed Messages Guide,* GC21-7704.

SORT-7450 I XX, XXX MAIN STORAGE BYTES ASSIGNED

This message indicates the amount of main storage assigned for the sort program.

SORT-7451 I XXXX BYTES—INPUT RECORD LENGTH

This message indicates the input record length.

SORT-7452 I XXXX BYTES—WORK FILE RECORD LENGTH

This message indicates the record length used for the work file. The record length is computed from the information in the header line.

SORT-7453 I XXXX BYTES—OUTPUT RECORD LENGTH

This message indicates the record length of the output file. The record length is computed from the information in the header line.

SORT-7461 I XXXX BYTES—SELECT/BUILD ROUTINE

This message indicates the number of bytes in main storage occupied by the sequence specifications (including any alternate collating statements).

SORT-7462 I XXXX SEQUENCE SPECIFICATION STATEMENTS

This message indicates the number of sequence specifications that the generation phase processed for this job.

SORT-7600 I INITIAL PHASE STARTED

The execution phase of the job began. Records are being read from the input file and written to the work file in intermediate strings.

SORT-7601 I #XX PASS COMPLETED

Pass XX was completed successfully. The first pass is 00, the second pass 01, and so on.

SORT-7602 I XXXXXX INPUT RECORDS WERE READ IN

This message indicates the number of input records read for your job.

SORT-7603 I XXXXXX RECORDS WERE SELECTED TO BE SORTED

This message indicates how many input records will be sorted.

SORT-7620 A NO INPUT RECORDS WERE SELECTED TO BE SORTED

None of the input records met the include requirements of the sort sequence specifications. Check the specifications or input records that were specified. Sort operator message SORT-7724 is issued after this message. For an explanation of the sort operator messages, see *Sort Operator Messages* in the *IBM System/32 Displayed Messages Guide,* GC21-7704.

SORT-7681 T WORK RECORD COUNT IN ERROR

This message indicates an error in the sort program. Contact IBM for program support services.

SORT-7690 I SORT STARTING FINAL PASS

The final pass of the job began. Records are being written on the output file.

SORT-7691 I FINAL PASS SUCCESSFULLY COMPLETED

The final pass of the job was completed with no errors.

SORT-7692 I XXXXXX SORTED RECORDS PLACED ON OUTPUT FILE

The last pass of the sort is complete. This message indicates how many output records were sorted and placed on the output file.

SORT-7693 A NO INPUT RECORDS FOUND

No records are in the input file. If the job is continued, a null output file is created. Make sure the correct input file was specified and that the file contains the desired records. Sort operator message SORT-7724 is issued after this message. For an explanation of the sort operator messages, see *Sort Operator Messages* in the *IBM System/32 Displayed Messages Guide,* GC21-7704.

SORT-7694 I XX PASSES REMAINING

This message indicates the number of passes left for this job.

SORT-7695 I XXXX BLOCKS USED FOR WORK FILE

This message indicates the number of blocks that were used for the work file.

SORT-7901 I ****** NORMAL EOJ ******

The sort job was completed.

SORT-7902 I ******ABNORMAL EOJ******

Errors prevented the sort job from being run to normal completion.

SORT-7928 A INPUT FILE READ ERROR

A disk error occurred while reading records from an input file. Sort operator message SORT-7728 follows. For an explanation of the sort operator messages, see *Sort Operator Messages* in the *IBM System/32 Displayed Messages Guide,* GC21-7704.

SORT-7929 A WORK FILE READ/WRITE ERROR

A disk error occurred while reading or writing records on the work file. Sort operator message SORT-7729 is issued after this message. For an explanation of the sort operator messages, see *Sort Operator Messages* in the *IBM System/32 Displayed Messages Guide,* GC21-7704.

SORT-7930 A OUTPUT FILE WRITE ERROR

A disk error occurred while the sorted records were being written on the output file. Sort operator message SORT-7730 is issued after this message. For an explanation of the sort operator messages, see *Sort Operator Messages* in the *IBM System/32 Displayed Messages Guide,* GC21-7704.

SORT-7931 A WORK FILE TOO SMALL

The sort work file is full. See the description of sort operator message SORT-7731. Sort operator message SORT-7731 is issued after this message. For an explanation of the sort operator messages, see *Sort Operator Messages* in the *IBM System/32 Displayed Messages Guide,* GC21-7704.

SORT-7932 A OUTPUT FILE TOO SMALL

The sort output file is full. See the description of sort operator message SORT-7732. Sort operator message SORT-7732 is issued after this message. For an explanation of the sort operator messages, see *Sort Operator Messages* in the *IBM System/32 Displayed Messages Guide,* GC21-7704.

Appendix F. Glossary

ADDROUT sort: The type of sort where the output consists of record addresses in the input file.

alphabetic character: Letters A through Z and special symbols #, $, and @.

alphameric character: The alphabetic characters defined and numeric characters 0-9.

block: A predefined unit of storage space for files on the disk. For example, 1 block = 2560 bytes.

byte: The representation of a character by eight binary digits; the amount of storage required for one EBCDIC character.

conditional force: The control field character is replaced before the record is sorted, only if the control field in the input record contains a particular entry.

constant: Data that does not change during execution of a program.

control field: One or more specified fields that are compared to determine the record sequence in the output file.

data: A collection of facts, numbers, letters, and symbols that can be processed by a computer.

data field: See *field.*

execution phase: The part of the sort program that gets the records you want to sort, sorts the records, and writes the sorted records onto the output file.

field: One or more adjacent record positions that contain related information.

file: An organized collection of related records.

file description: The specification for the arrangement of fields within a record and records within a file.

forced control field: A one-position control field that results from replacing the value or adding a value.

generation phase: The part of the sort program that translates the sequence specifications into machine language.

input: Data transferred to computer storage for processing.

integer: A whole number stored in hexadecimal (binary) form.

keyword: A nonvariable character string in sequence specifications (such as UDATE).

library: An area on disk that contains procedure members, source members, load members, and subroutine members as well as areas required by the system control program. For example, sort is stored in the library.

normal control field: Any control field that is specified by an N in column 7 of the field sequence specifications.

OCL: Operation control language.

OCL statement: A statement that identifies the job and its processing requirements to the control program.

offline multivolume file: A multivolume file that is processed in segments by the system. Each segment is processed before the next segment is copied to disk. A segment consists of the information on one diskette.

opposite control field: Any control field that is specified by an O in column 7 of the field sequence specifications.

output: Data transferred from computer storage to printed form, disk, or to the display screen.

output file: File resulting from the sort.

overflow field: In a summary tagalong sort, the field that allows for anticipated field expansion.

packed data field: Each byte within a field represents two numeric digits except the low-order byte, which contains the sign. For non low-order bytes, bits 0-3 for one digit; bits 4-7 for one digit.

procedure: A named collection of related OCL statements, and possibly control statements that perform a particular task.

procedure member: A procedure stored in the library.

program date: The date associated with a program (job step).

real number: A number stored in IBM System/360 floating-point format, in which the leftmost byte contains the sign and exponent (decimal location) and the remaining bytes contain the fraction or mantissa in left-justified binary form.

record: A collection of related data, treated as a unit. For example, one line of an invoice might form a record.

relative record number: In a direct file, the location of a record in relation to the beginning of the file.

right-justify: The placement of data in a field with the last significant byte in the rightmost position.

sequence specification: Defines the arrangement of a file.

source member: A collection of records used as input to a program such as RPG II specifications or sort sequence specifications. Source members are stored in the library.

summary data field: In a summary tagalong sort, a data field designated for accumulated totals.

summary tagalong sort: A type of sort that arranges records by a specified control field and accumulates totals from specified fields within those records.

tagalong sort: A type of sort that arranges records by a specified control field and includes other specified data fields in the output.

unpacked data field: Each byte within a field represents one numeric digit. Bits 0 through 3 are reserved for the sign; bits 4 through 7 are the numeric portion.

work file: An area reserved on the disk for temporary storage of the data being sorted.

Index

READER'S COMMENT FORM

Please use this form only to identify publication errors or request changes to publications. Technical questions about IBM systems, changes in IBM programming support, requests for additional publications, etc, should be directed to your IBM representative or to the IBM branch office nearest your location.

Error in publication (typographical, illustration, and so on). **No reply.**

Page Number *Error*

Inaccurate or misleading information in this publication. Please tell us about it by using this postage-paid form. We will correct or clarify the publication, or tell you why a change is not being made, provided you include your name and address.

Page Number *Comment*

Note: All comments and suggestions become the property of IBM.

Name ______________________________

Address ____________________________

____

● No postage necessary if mailed in the U.S.A.

IBM System/32
Utilities Program Product
Reference Manual
Sort

SC21-7633-2

Cut Along Line

Fold

Fold

FIRST CLASS
PERMIT NO. 40
ARMONK, N. Y.

BUSINESS REPLY MAIL

NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY . . .

IBM Corporation
General Systems Division
Development Laboratory
Publications, Dept. 245
Rochester, Minnesota 55901

Fold

Fold

IBM

International Business Machines Corporation

General Systems Division
4111 Northside Parkway N.W.
P.O. Box 2150
Atlanta, Georgia 30301
(U.S.A. only)

General Business Group/International
44 South Broadway
White Plains, New York 10601
U.S.A.
(International)

IBM ® Technical Newsletter

This Newsletter No. SN21-8090
Date 30 May 1980

Base Publication No. SC21-7633-2
File No. S32-33

Previous Newsletters None

IBM System/32
Utilities Program Product
Reference Manual
Sort

This technical newsletter, a part of version 9, modification 0, provides replacement pages for the subject publication. These replacement pages remain in effect for subsequent versions and modifications unless specifically altered. Pages to be inserted and/or removed are:

9 through 12
57, 58
125, 126

Changes to text and illustrations are indicated by a vertical line at the left of the change.

Summary of Amendments

Add miscellaneous technical changes

Note: Please file this cover letter at the back of the manual to provide a record of changes.

IBM Corporation, Publications, Department 245, Rochester, Minnesota 55901

Printed in U.S.A

International Business Machines Corporation

General Systems Division
4111 Northside Parkway N.W.
P.O. Box 2150
Atlanta, Georgia 30301
(U.S.A. only)

General Business Group/International
44 South Broadway
White Plains, New York 10601
U.S.A.
(International)

S/32 Utilities Reference Manual—Sort (File No. S32-33) Printed in U.S.A. SC21-7633-2