DEAR AUTHOR:

Your program has been accepted into the Users' Library; but our engineer
has suggested that I forward these comments to you for your future reference.

Carol Barnes
Users' Library

The program works very well, is a very good application, good
documentation. We will list the program under B206-Data Base
Management & also P600-Subroutine Packages.

022840

2 # PROGRAM SUBMITTAL

7

☒ **New Program**   ☐ **Revision to Program**

Model No.   ☐ 67   ☐ 97   ☒ 41C

Program Title   Sort/Merge for Extended Memory ASCII Files

No. of Steps/Lines   1 3 9 287 bytes   **Category No.** B206,9600 ~~4-6-0~~

Category Name   DATA BASE MANAGMENT ~~Computer Science~~

Abstract — 50 Word Maximum This is a general utility sort program for Extended Memory ASCII files. Sort fields may contain any character except the null byte. Each character is ordered according to its equivalent numeric value. The user must specify the sort field's starting location and length (up to 24 characters long), and whether the sort is to be in ascending or descending order. New records may easily be merged into an already sorted file. A subroutine entry point is also provided.

Necessary Accessories: Extended Functions Module

Name   Gary Goodman

Company System Development Corporation

Address 601 Caroline Street

City   Fredericksburg   **State/Country** Virginia   **Zip Code** 22401

Phone Number (703 **)** 371-8316

If my program is accepted, my bonus choice is: (Please select two programs if your program is a revision.)

Acceptance Choice: ☐ FOUR PROGRAMS, ☒ CREDIT FOR FOUR PROGRAMS*, OR TWO PROGRAMS AND 10 BLANK CARDS.

⌊⌊⌊⌊⌊⌋   ⌊⌊⌊⌊⌊⌋   ⌊⌊⌊⌊⌊⌋   ⌊⌊⌊⌊⌊⌋

* No partial credit will be given. Select all four programs at the same time.   12/6/82

Submittal Checklist: Please use the checklist below to insure submittal of all proper program documentation.

☒ **Program Submittal**   ☒ **Program Description II**   ☒ **Program Listing(s)**   ☒ **Registers, Status ...**

☒ **Program Description I**   ☒ **User Instructions**   ☒ **Magnetic Card(s)**   ☐ **Keyboard, Card Labeling** (optional)

PROGRAM SUBMITTAL

☒ **New Program**   ☐ **Revision to Program**

Model No.   ☐ 67   ☐ 97   ☒ 41C

Program Title   Sort/Merge for Extended Memory ASCII Files

No. of Steps/Lines 1 3 9   **Category No.** 4 6 0

Category Name   Computer Science

Abstract — 50 Word Maximum   This is a general utility sort program for Extended Memory ASCII files. Sort fields may contain any character except the null byte. Each character is ordered according to its equivalent numeric value. The user must specify the sort field's starting location and length (up to 24 characters long), and whether the sort is to be in ascending or descending order. New records may easily be merged into an already sorted file. A subroutine entry point is also provided.

Necessary Accessories: Extended Functions Module

Name   Gary Goodman

Company System Development Corporation
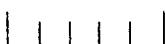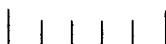
Address 601 Caroline Street

City   Fredericksburg   **State/Country** Virginia   **Zip Code** 22401

Phone Number (703 **)** 371-8316

If my program is accepted, my bonus choice is: (Please select two programs if your program is a revision.)

Acceptance Choice: ☐ FOUR PROGRAMS, ☒ CREDIT FOR FOUR PROGRAMS*, OR TWO PROGRAMS AND 10 BLANK CARDS.

⌊_⌊_⌊_⌊_⌊_⌋   ⌊_⌊_⌊_⌊_⌊_⌋   ⌊_⌊_⌊_⌊_⌊_⌋   ⌊_⌊_⌊_⌊_⌊_⌋

***** No partial credit will be given. Select all four programs at the same time.**

Submittal Checklist: Please use the checklist below to insure submittal of all proper program documentation.

☒ Program Submittal   ☒ Program Description II   ☒ Program Listing(s)   ☒ Registers, Status ...

☒ Program Description I   ☒ User Instructions   ☒ Magnetic Card(s)   ☐ Keyboard, Card Labeling (optional)

Program Title Sort/Merge for Extended Memory ASCII Files

Contributor's Name Gary Goodman

Address System Development Corporation, 601 Caroline Street

City Fredericksburg **State/Country** Virginia **Zip Code** 22401

Program Description This is a general utility sort program for ASCII files contained in the HP-41C's extended memory (available with the Extended Functions/Memory modules). Sort ∧fields may contain any character except the null byte. Each character is ordered according to the equivalent numeric value of its byte code. The user must specify the sort field's starting location and length (up to 24 characters long), and whether the sort is to be in ascending or descending order. New records added to the end of a previously sorted file may easily be merged into their proper location by starting the sort from the position of the new records near the end of the file. A full sort takes a considerable amount of time; merging records into a presorted file is much quicker.

When used as a program, the user will be clearly prompted for all required parameters. However, the sort may also be called as a subroutine, in which case the prompting sequence is skipped.

Necessary Accessories Extended Functions module

Operating Limits and Warnings The file must contain some free space at the end for the sort to execute. The minimum space is the lesser of (25 or the record length + 1) bytes. If not enough free space remains, the program will terminate with the END OF FL error message. The file will not be changed.

Note: All references to record numbers and column positions follow the standard terminology given in the Extended Functions/Memory Module Owner's Manual, i.e. the first record is record 0, and the first column is column 0.

Sample Problem (Sketch if Desired)   A "MEMO" File containing prioity messages, date & time messages, date messages, and long-term messages are to be sorted in the above order and by date and time if they are those message types.

1. Create an ASCII file named "MEMO" containing 40 registers. Put the following records in the file.

```
                1         2         3      column
01234567890123456789012345678901234567890   position
09/29 14:30 MEETING
10/01 20:00 MOVIES
  CALL JACK   ASAP
09/28 GROCERIES $42.38
LENT DRILL TO SCOTT
  PICK UP BREAD AND MILK
FINISH PROGRAM FOR HP USERS LIB
09/29 GAS $13.60
10/01 12:00 LUNCH WITH RALPH
```

Sort the file as specified above, i.e. sort columns 0 through 10 in ascending order. View the sorted file.

SOLUTION:

Input	Function	Display	Comments
40			
"MEMO"	CRFLAS	40.00	Create ASCII file named MEMO.
"09/29 14:30 ..."	APPREC	40.00	Put first record into file.
"10/01 20:00 ..."	APPREC	40.00	Append next record into file.
" CALL JACK ..."	APPREC	40.00	"
"09/28 GROCER..."	APPREC	40.00	"
"LENT DEILL T..."	APPREC	40.00	"
" PICK UP BRE..."	APPREC	40.00	"
"FINISH ...HP US"	APPREC	40.00	Append first part of record.
"ERS LIB"	APPCHR (note)	40.00	Append remainder of record.
"09/29 GAS $1..."	APPREC	40.00	Append next record into file.
"10/01 1...ITH R"	APPREC	40.00	Append first part of record.
"ALPH"	APPCHR (note)	40.00	Append remainder of record.
	XEQ 'SORT	FILE NAME?	Name of file to be sorted
"MEMO"	R/S	START COL.?	First column pos. of sort field
0	R/S	LENGTH?	Length of sort field
11	R/S	SORT FROM 1?	Sort from 1st record? R/S for Yes
	R/S	ASC. OR DESC.	Ascending or Descending, A or D
	A	1.	working on first record
		2.	working on second record
		3. etc.	Number shows record being sorted.
		9.00	Number of records in the file
0	SEEKPT	0.00	Position record pointer to rec. 0.
	GETREC	CALL JACK ASAP	priority record
	GETREC	PICK UP BREAD AND MILK	next priority record
	GETREC	09/28 GROCERIES $42.38	1st date rec. No time rec for 9/28
	GETREC	09/29 14:30 MEETING	date and time record for 9/29
	GETREC	09/29 GAS $13.60	followed by date record for 9/29
	GETREC	10/01 12:00 LUNCH WITH R	first part of record for 10/01
	GETREC	ALPH	remainder of preceeding record
	GETREC	10/01 20:00 MOVIES	next date and time record for 10/01
	GETREC	FINISH PROGRAM FOR HP US	first long-term message
	GETREC	ERS LIB	remainder of preceeding message
	GETREC	LENT DRILL TO SCOTT	last long-term message

Sample Problem (Sketch if Desired)

2. Add the following two records to the file which was created and sorted in step 1 on the previous page.

```
              1         2         3
0123456789012345678901234567890
12/18 ANDYS WEDDING
 MAIL LETTERS
```

Merge these records with the sorted records, using the same sort order.
View the sorted file.

SOLUTION:

Input	Function	Display	Comments
"MEMO"	FLSIZE	40.00	Make MEMO the working file.
"12/18 ANDYS ..."	APPREC	40.00	Append first record to file.
" MAIL LETTERS"	APPREC	40.00	Append next record.
	XEQ "SORT	FILE NAME?	Name of file to be sorted
"MEMO"	R/S	START COL.?	First column pos. of sort field
0	R/S	LENGTH?	Length of sort field
11	R/S	SORT FROM 1?	No! Start with the new rec., #9
9	R/S	ASC. OR DESC.	Key A for ascending order
		9.	merging ninth record
		10.	merging tenth record
		11.00	Number of records in the file
0	SEEKPT	0.00	Position record pointer to rec. 0.
	GETREC	CALL JACK ASAP	
	GETREC	MAIL LETTERS	priority record merged here
	GETREC	PICK UP BREAD AND MILK	
	GETREC	09/28 GROCERIES $42.38	
	GETREC	09/29 14:30 MEETING	
	GETREC	09/29 GAS $13.60	
	GETREC	10/01 12:00 LUNCH WITH R	
	GETREC	ALPH	
	GETREC	10/01 20:00 MOVIES	
	GETREC	12/18 ANDYS WEDDING	date record merged here
	GETREC	FINISH PROGRAM FOR HP US	
	GETREC	ERS LIB	
	GETREC	LENT DRILL TO SCOTT	

		SIZE: (HP-41C) 014	

STEP	INSTRUCTIONS	INPUT	FUNCTION	DISPLAY
1.	Enter the program & check that SIZE ≥ 14			
2.	Execute the Sort/Merge program.		XEQ ˥SORT	FILE NAME?
3.	Enter name of ASCII file to be sorted.	name	R/S	START COL?
4.	Enter the column position* of the start of the sort field. ·	col. #	R/S	LENGTH?
5.	Enter the length of the sort field. (length ≤ 24)	length	R/S	SORT FROM 1?
6a.	To sort the entire file:		R/S	ASC. OR DESC.
6b.	Else to merge new records at the end of a file into their sorted positions Enter the record number* of the first record to be merged.	rec. #	R/S	ASC. OR DESC.
7.	Enter sort order: ascending or descending		A or D	see ** below

	Comments			
*	Column positions and record numbers start at 0, same as X-functions manual.			
**	While the program is running, the display shows the record number of the record currently being sorted. At the end of the sort, the display shows the number of records in the file.			

	Subroutine Useage			
	To execute Sort/Merge as a subroutine, the calling program must:			
a.	put the name of the file to be sorted into the ALPHA register,			
b.	store the column position of the start of the sort field into reg. 07,			
c.	store the length of the sort field into reg. 08,			
d1.	for a full sort, store 1 in reg. 09,			
d2.	for merging records, store the record number of the first record to be merged in reg. 09,			
e.	for an ascending sort, clear flag 01, for a descending sort, set flag 01,			
f.	call Sort/Merge with XEQ ˥*SORT			

⊠ 41C

STEP/LINE	KEY ENTRY	COMMENTS	STEP/LINE	KEY ENTRY	COMMENTS
01	◆LBL "SORT"	Prompts for the sort parameters	42	RCL 09	0.001(R9 - 1) used for comparison rec. loop control
02	CF 01		43	1 E-3	
03	"FILE NAME?"		44	ST* Y	
04	AON		45	-	
05	PROMPT		46	STO 11	
06	AOFF		47	SF 00	
07	FLSIZE		48	SF 25	
08	"START COL.?"		49	RCL 09	
09	PROMPT		50	XEQ 80	– Eval. sort rec.
10	STO 07		51	CF 00	
11	"LENGTH?"		52	FS?C 25	Check for EOF
12	PROMPT		53	GTO 02	
13	STO 08		54	BEEP	
14	1		55	RTN	– sort return point
15	"SORT FROM 1?"		56	◆LBL 02	
16	PROMPT		57	RCL 11	
17	STO 09		58	INT	
18	"ASC. OR DESC."		59	XEQ 80	– Eval 1st part of comparison rec.
19	SF 27		60	◆LBL 03	
20	◆LBL 00		61	RCL IND 12	
21	PROMPT		62	FS? 01	Compare sort fields
22	GTO 00		63	X<>Y	
23	◆LBL D		64	X<Y?	
24	SF 01		65	GTO 04	– Insert record
25	◆LBL A		66	X>Y?	
26	◆LBL "*SORT"	Subroutine entry point and start of sort process	67	GTO 06	
27	1 E3		68	RCL 13	
28	ST/ 07		69	X<=0?	
29	RCL 08	Calc. number of registers needed to hold numerical equivalent of the characters in the sort field.	70	GTO 06	– Sort fields match
30	3		71	XEQ 81	– Eval. next part of compar. rec.
31	+		72	DSE 12	
32	4		73	GTO 03	
33	/		74	◆LBL 04	Insert Record Process
34	INT		75	RCL 09	
35	STO 10		76	SEEKPT	
36	◆LBL 01	– Start main loop	77	GETREC	
37	RCLFLAG		78	1	
38	FIX 0	Display rec. # of record being sorted.	79	ST+ 09	
39	CF 21		80	RCL 11	
40	VIEW 09		81	INT	
41	STOFLAG		82	SEEKPT	Insert first 24 characters of rec.
			83	INSREC	
			84	RCL 09	
			85	SEEKPT	
			86	24	
			87	DELCHR	
			00		

☒ 41C

STEP/LINE	KEY ENTRY	COMMENTS	STEP/LINE	KEY ENTRY	COMMENTS
88	◆LBL 05		107	◆LBL 80	Subroutine to
89	FC? 17	Check for	108	RCL 07	return the ASCII
90	GTO 01	end-of-record	109	+	numerical equiv.
91	RCL 09		110	SEEKPT	of the characters
92	SEEKPT	Insert Rest of	111	FS? 00	in the sort field.
93	GETREC	Record Process	112	FS? 25	
94	SEEKPT		113	X<0?	
95	24		114	RTN	
96	DELCHR		115	GETREC	
97	RCL 11		116	RCL 10	
98	SEEKPT		117	STO 12	
99	APPCHR		118	RCL 08	
100	GTO 05		119	STO 13	
101	◆LBL 06	Increment loop	120	◆LBL 81	
102	ISG 11	counters	121	RCL 13	
103	GTO 02	- next compar. rec.	122	4	Characters are
104	1		123	X>Y?	taken in groups
105	ST+ 09		124	X<>Y	of 4.
106	GTO 01	- next sort rec.	125	ST- 13	
			126	0	
			127	◆LBL 82	
			128	256	
			129	*	
			130	ATOX	- numerical equiv.
			131	+	
			132	DSE Y	
			133	GTO 82	- next letter
			134	FC? 00	
			135	RTN	
			136	STO IND 12	
			137	DSE 12	
			138	GTO 81	- next group
			139	END	

REGISTERS, STATUS, FLAGS, ASSIGNMENTS

DATA REGISTERS	STATUS

DATA REGISTERS

00	unused
1	
2	These registers are used to
3	hold the numerical equivalent
4	of the characters in the sort
5	field, taken in groups of 4.
6	
7	Starting column position of sort field
8	Length of sort field
9	Record pointer for sort record
10	used -- Ceiling(length/4)
11	used -- loop controller & rec. ptr.
12	used -- indirect addressing
13	used -- characters remaining in field

STATUS

SIZE 014	TOT. REG. 055	USER MODE
ENG	FIX SCI	ON OFF
DEG	RAD GRAD	

FLAGS

#	INIT S/C	SET INDICATES	CLEAR INDICATES
0		eval. sort rec.	eval. compar. rec.
1		descending sort	ascending sort
17		not end-of-record	end-of-record
21			no print
25		End-of-file clears flag 25.	
27		USER mode	

Note: Program restores display format and the status of flag 21 to their initial settings.

ASSIGNMENTS

FUNCTION	KEY	FUNCTION	KEY