

TITLE: Generalized Memory Search Routine

AUTHOR: James Orton, Royal McBee Corporation

ABSTRACT:

DISCLAIMER:

"The authors of this program material, the POOL organization and Royal McBee believe this program to be correct; however they bear no responsibility, financial or otherwise, for errors resulting from its use. This program is distributed only to individual and installation members of POOL. Further distribution of this manual and accompanying tapes for use by non-members is prohibited."



GENERALIZED MEMORY SEARCH ROUTINE

Royal McBee Corporation
James Orton

FUNCTION:

To examine a sequence of memory locations from L_0 to L_f and print out the contents of every location whose address is in the range A_0 to A_f . The search does not alter the contents of the locations searched. Output is in the standard form of the Decimal Memory Printout (Program 21.0), a modified version of which is used for this purpose.

OPERATION:

1. With the routine located starting at location M_0 , transfer to $M_0 + 0400$ (locations M_0 through $M_0 + 0363$ are used for the output subroutine).
2. The computer will print "adr" and stop on input. Type in A_0A_f in eight digits (for example, "34173523" if $A_0 = 3417$ and $A_f = 3523$). Depress START COMP lever. Both A_0 and A_f , with any leading zeros, must be input.
3. The computer will print "loc" and stop on input. Type L_0L_f as above. Depress START COMP lever. Both L_0 and L_f , with any leading zeros, must be input.
4. The computer will examine the locations $L_0, L_0 + 1, \dots, L_f$ and print out the contents of every location whose address A is such that

$$A_0 \leq A \leq A_f,$$

one location to a line. If the word is in hexadecimal form, a tab and the binarized address portion of the word are also printed (see sample output below). Non-instruction words are printed in hexadecimal or decimal at $q = 0$, according to whether the transfer control button is depressed or raised.

5. The last instruction of the routine is a Breakpoint 4 stop, followed by a return to the starting location ($M_0 + 0400$).

SAMPLE OUTPUT:

Suppose that the following words are stored in locations 5000 through 5008 and that the search routine is located starting at 2000:

5000	Z0000
5001	h6363
5002	t5012
5003	Z0001
5004	,0000006
5005	Z0000
5006	,8wwj002
5007	s3718
5008	,80000000

The result of searching for addresses 0000 through 0001 over locations 5000 to 5008 would be as follows:

.0002400		
adr	00000001	
loc	50005008	
5000	Z0000	
5003	Z0001	
5004	,0000006	0001
5005	Z0000	
5006	,8wwj0002	0000
5008	,80000000	0000

STORAGE:

The program occupies 5 1/2 tracks. No external temporary storage is required.

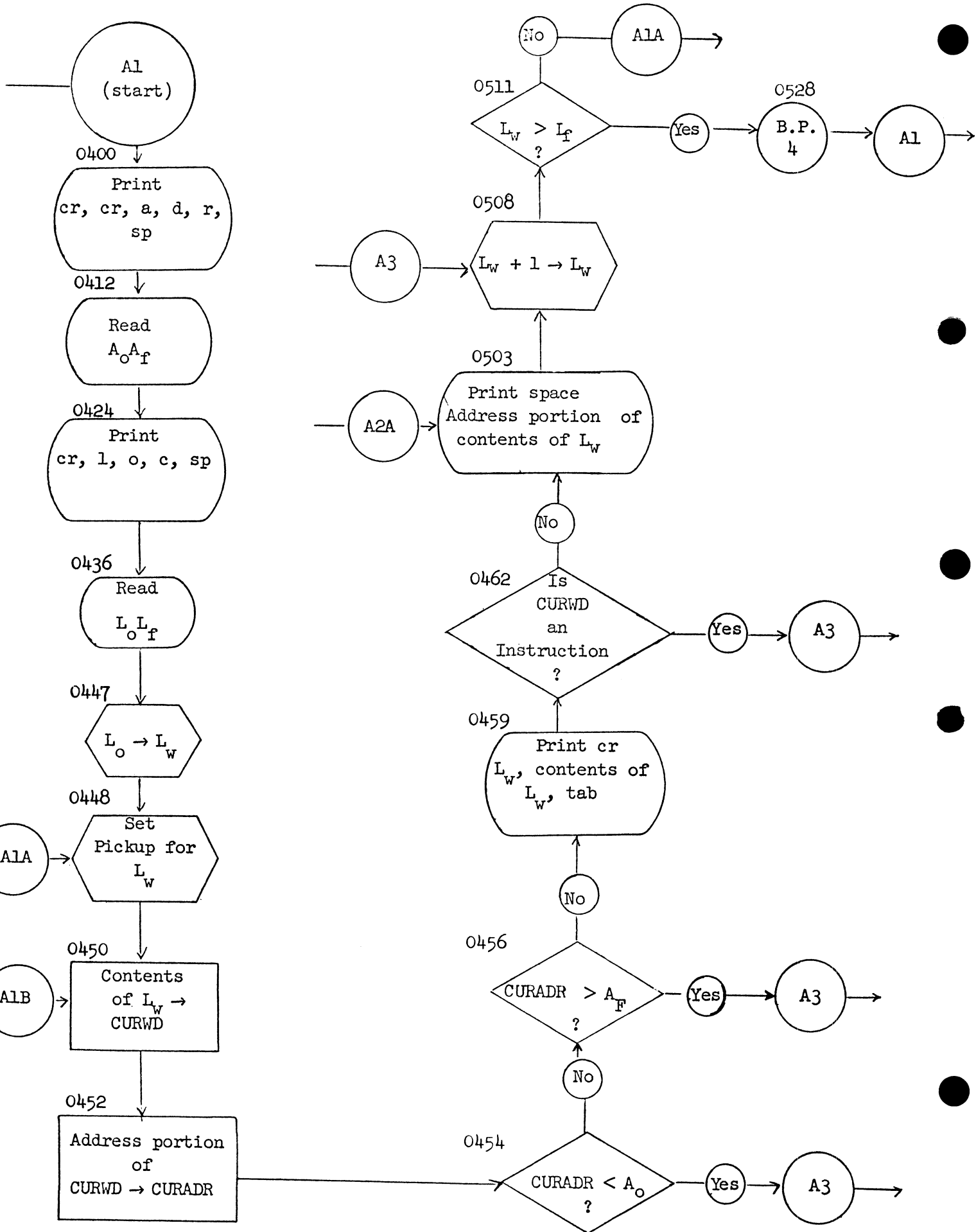
TIME:

To search the entire drum for any given sequence of addresses, exclusive of printing, requires about 8.0 min. Printing speed is the same as for Program 21.0.

SUBROUTINE D2A:

The following changes in the Decimal Memory Printout routine (21.0) were made to obtain D2A, the output for the Generalized Memory Search:

<u>Location</u>	<u>Was</u>	<u>Change To</u>	<u>Comments</u>
0002	xp0000	u0019	To set modifier =0
0003	xi0000	y0108	For address input
0004	h0254	xc0143	Clear ACC
0005	r0341	y0225	
0006	u0325	h0204	0 into Lf
0007	y0225	u0000	
0019	xp1648	xp0348	Space to replace C. R.
0020	t0359	u0359	To set modifier = 0
0149	xp1600	xp2400	Tab to replace C. R.
0300	xp1600	xu6363	Subroutine exit



GENERALIZED MEMORY SEARCH ROUTINE

K3-156

PREPARED FOR: L&P-30 RPC-4000 USERS ORGANIZATION - POOL			PAGE OF 1 / 3	
JOB NO.	PROGRAM NO. K3-156	PROGRAM PREPARED BY: J. Orton	PROGRAM CHECKED BY: POOL Review	DATE 12/17/59
PROBLEM: Generalized Memory Search Routine				TRACK 04

PROGRAM INPUT CODES	STOP	LOCATION	INSTRUCTION		STOP	CONTENTS OF ADDRESS	NOTES
			OPERATION	ADDRESS			
; 0 0 0 [Lo+0400]	/						
/ 0 0 0 [, Lo ,]	/	⊗					
		0 4 0 0	X P	1 6 0 0	/	c.r.	A1 (Start)
		0 1	X Z	0 0 0 1	/	1 @ 29	(0415)
		0 2	X P	1 6 0 0	/	c.r.	
		0 3	X Z	0 0 0 0	/	⊗	
		0 4	X P	5 7 0 0	/	a	
		0 5	X Z	0 0 0 0	/		
		0 6	X p	2 1 0 0	/	d	
		0 7	X Z	0 0 0 0	/	⊗	
		0 8	X P	1 3 0 0	/	r	
		0 9	X Z	0 0 0 0	/		
		1 0	X P	0 3 0 0	/	space	
		1 1	X Z	0 0 0 0	/	⊗	
		1 2	X P	0 0 0 0	/	} Input: Type in	
		1 3	X I	0 0 0 0	/		} A ₀ A _f ; start compute
		1 4	H	0 5 1 9	/	Temp 1	
		1 5	N	0 4 0 1	/	⊗ 1 @ 29	A _f @ 29
		1 6	R	0 3 4 1	/	} D2A	Binarize
		1 7	U	0 3 2 6	/		
		1 8	H	0 5 1 5	/	AF	
		1 9	B	0 5 1 9	/	⊗ Temp 1	A ₀ @ 15
		2 0	M	0 5 2 3	/	1 @ 14	A ₀ @ 29
		2 1	R	0 3 4 1	/	} D2A	Binarize
		2 2	U	0 3 2 6	/		
		2 3	H	0 5 2 6	/	⊗ AZ	
		2 4	X P	1 6 0 0	/	c.r.	
		2 5	X Z	0 0 0 0	/		
		2 6	X P	0 6 0 0	/	1	
		2 7	X Z	0 0 0 0	/	⊗	
		2 8	X P	3 5 0 0	/	o	
		2 9	X Z	0 0 0 0	/		
		3 0	X P	5 3 0 0	/	c	
		3 1	X Z	0 0 0 0	/	⊗	

Royal McBee Corporation
 DATA PROCESSING DIV.
 PORT CHESTER, NEW YORK

PREPARED FOR: LGP - 30 RPC-4000 USERS ORGANIZATION - POOL				PAGE 2	OF 3
JOB NO.	PROGRAM NO. K3-156	PROGRAM PREPARED BY: J. Orton	PROGRAM CHECKED BY: POOL Review	DATE 12/17/59	
PROBLEM: Generalized Memory Search Routine				TRACK 04	

PROGRAM INPUT CODES	STOP	LOCATION	INSTRUCTION		STOP	CONTENTS OF ADDRESS	NOTES
			OPERATION	ADDRESS			
	/						
	/	0 4 3 2	X P	0 3 0 0	/	Space	
		3 3	X Z	0 0 0 0	/		
		3 4	U	0 4 3 6	/		
, 0 0 0 0 0 0 1	'	3 5	W W W 0	J 0 0 2	/	X	(0463) NONINS
		3 6	X P	0 0 0 0	/	} Input: Type in } Lo Lf; start compute	
		3 7	X I	0 0 0 0	/		
		3 8	H	0 5 1 7	/	Temp 2	
		3 9	N	0 5 1 8	/	X 1 @ 29	Lf @ 29
		4 0	R	0 3 4 1	/	} D2A	Binarize
		4 1	U	0 3 2 6	/		
		4 2	H	0 5 2 1	/	Lf	
		4 3	B	0 5 1 7	/	X Temp 2	Lo @ 15
		4 4	M	0 5 2 3	/	1 @ 14	Lo @ 29
		4 5	R	0 3 4 1	/	} D24	Binarize
		4 6	U	0 3 2 6	/		
		4 7	H	0 5 2 7	/	X LW	
		4 8	B	0 5 2 7	/	LW	ALA
		4 9	Y	0 4 5 0	/	ALB	
		5 0	X B	[]	/	[LW]	ALB
		5 1	H	0 5 3 0	/	X CURWD	
		5 2	E	0 5 2 4	/	B/18-29	
		5 3	H	0 5 2 5	/	CURADR	
		5 4	S	0 5 2 6	/	AZ	
		5 5	T	0 5 0 8	/	X A3	If addr (Lw) < A ₀
		5 6	B	0 5 1 5	/	AF	
		5 7	S	0 5 2 5	/	CURADR	
		5 8	T	0 5 0 8	/	A3	If Addr (Lw) > A ₂
		5 9	B	0 5 2 7	/	X Lw	
		6 0	R	0 3 0 0	/	} D2A	Print Cr, L _w
		6 1	U	0 0 0 3	/		
		6 2	B	0 5 3 0	/	CURWD	
		6 3	E	0 4 3 5	/	X NONINS	Bits S, HI, 16-17,30

PREPARED FOR: LGP-30 RPC-4000 USERS ORGANIZATION - POOL			PAGE 3	OF 3
JOB NO.	PROGRAM NO. K3-156	PROGRAM PREPARED BY: J. Orton	PROGRAM CHECKED BY: POOL Review	DATE 12/17/59
PROBLEM: Generalized Memory Search Routine			TRACK 05	

PROGRAM INPUT CODES	STOP	LOCATION	INSTRUCTION		STOP	CONTENTS OF ADDRESS	NOTES
			OPERATION	ADDRESS			
	/						
	/	05 0 0	T	05 0 3	/		
		0 1	S	05 1 6	/	1 @ 30	
		0 2	T	05 0 8	/	A3	
		0 3	X P	03 0 0	/	space	A2A
		0 4	B	05 2 5	/	CURADR	
		0 5	R	03 5 3	/	D2A } Print Addr(Lw)	
		0 6	U	03 4 4	/		
		0 7	X Z	00 0 0	/		
		0 8	B	05 2 7	/	Lw	A3
		0 9	A	05 3 1	/	1 @ 29	
		1 0	H	05 2 7	/	Lw	
		1 1	B	05 2 1	/	Lf	
		1 2	S	05 2 7	/	Lw	
		1 3	T	05 2 8	/	Stop	If Lw > Lf next inpt
		1 4	U	04 4 8	/	ALA	To next loc.
		1 5	[/	Af	(0418)(0456)
, 0 0 0 0 0 0 1	/	1 6		2	/	1 @ 30	(0501)
		1 7	[/	Temp 2	(0438)(0443)
		1 8	X Z	00 0 1	/	1 @ 29	(0439)
		1 9	[/	Temp 1	(0414)(0419)
		2 0	[/		
		2 1	[/	LF	(0442)(0511)
		2 2	[/		
		2 3	X Y	00 0 0	/	1 @ 14	(0444)(0420)
		2 4	X Z	63 6 3	/	B/18-29	(0452)
		2 5	[/	CURADR	(0453)(0457)(0504)
		2 6	[/	AZ	(0423)(0454)
		2 7	[/	LW (0447)	(0448)(0459)(0508) (0510)(0512)
		2 8	X Z	04 0 0	/	B.P. 4 stop	
		2 9	U	04 0 0	/	A1	Next input
		3 0	[/	CURWD	(0451)(0462)
		3 1	X Z	00 0 1	/	1 @ 29	(0509)

Royal McBee Corporation
 DATA PROCESSING DIV.
 PORT CHESTER, NEW YORK

