

Program J (*Inverse in place*)

N	IS	6	Number of elements in the permutation
t	IS	\$255	Temporary storage
i	IS	\$2	Variables of the algorithm
ii	IS	i	Two characters indicate that the
jj	IS	\$3	variable is multiplied by 8
k	IS	jj	
mm	IS	\$4	
	LOC	Data_Segment	
X	GREG	@	
	OCTA	0	X[0] is not used
	OCTA	6,2,1,5,4,3	The data of Table 1.3.3-4
	LOC	#100	
* Inverse a permutation in place			
01	Invert	SET t,N	1 <u>J1. Negate all.</u> $t \leftarrow n$.
02		SL mm,t,3	1 $m \leftarrow n$.
03	1H	LDO k,X,mm	N
04		NEG k,k	N
05		STO k,X,mm	N $X[k] \leftarrow -X[k]$.
06		SUB mm,mm,8	N $m \leftarrow m - 1$.
07		PBP mm,1B	N Branch if $m > 0$.
08		SL mm,t,3	1 $m \leftarrow n$.
09	2H	SR i,mm,3	N <u>J2. Initialize.</u> $i \leftarrow m$.
10	3H	SL jj,i,3	A <u>J3. Find neagtive entry.</u> $j \leftarrow i$.
11		LDO i,X,jj	A $i \leftarrow X[j]$.
12		PBP i,3B	A To J3 if $i > 0$.
13	4H	NEG i,i	N <u>J4. Invert.</u>
14		SL ii,i,3	N
15		LDO t,X,ii	N $t \leftarrow X[-i]$
16		STO t,X,jj	N and $X[j] \leftarrow X[-i]$.
17		SR t,mm,3	N $t \leftarrow m$
18		STO t,X,ii	N and $X[-i] \leftarrow m$.
19	5H	SUB mm,mm,8	N <u>J5. Loop on m.</u> $m \leftarrow m - 1$.
20		PBP mm,2B	N Branch if $m > 0$.
* inspect memory locations of array X for the result			
	TRAP	0,Halt,0	
Main	IS	Invert	█

Analysis

In this program some registers have two names. For example, the labels *i* and *ii* stand for one register. The program uses *i* to represent a value of the array *X* and *ii* for an index to this array.

The program needs $(5N + A)\mu + (16N + 3A + 7)v$. For the test data the procedure has the statistics: 117 instructions, 40 mems, 133 oops; 14 good guesses, 8 bad. As in this case $N = 6$ and $A = 10$ the formula gives $30 + 10 = 40\mu$ and $96 + 30 + 7 = 133v$.