

LABEL 000000000PRINTER00175100CC EX OBJECT/READ,FILE SOURCEFILE=SYMBOL/COOL+00+0000000

OBJECT /READ

SYMBOL/COOL

Data Documents/Inc.

33492

Data Documents, Inc.

STARTER DECKS
MAY 1969 BY P.E.G.

\$ SET OMIT = COOL

B=5500 COLD START PROGRAM

00010000
00020000
00030000

\$ POP OMIT

\$ SET OMIT = NOT(COOL)

B=5500 COOL START PROGRAM

00040000
00040050
00050000

\$ POP OMIT

COMMENT: * TITLE: B5500/B5700 MARK XIV SYSTEM RELEASE *

* FILE ID: SYMBOL/COOL TAPE ID: SYMBOL2/FILE000 *
* THIS MATERIAL IS PROPRIETARY TO BURROUGHS CORPORATION *
* AND IS NOT TO BE REPRODUCED, USED, OR DISCLOSED *
* EXCEPT IN ACCORDANCE WITH PROGRAM LICENSE OR UPON *
* WRITTEN AUTHORIZATION OF THE PATENT DIVISION OF *
* BURROUGHS CORPORATION, DETROIT, MICHIGAN 48232 *

00060000
00060050
00060100

* COPYRIGHT (C) 1971, 1972 BURROUGHS CORPORATION *
* AA320206 AA386657 *;

00060101
00060102
00060103

BEGIN DEFINE P=POLISH#,M=MEMORY#;

DEFINE DEFAULTDIRECT=4500#;

DEFINE KERNELADR=1#;

DEFINE MEMDISKADR=20#;

DEFINE ESPBOTTOM=50#;

DEFINE SCN(SCN1)=SCAN(CARD,SCN1)#;

NAAME=1#, NUMBER=2#, DELIM=0#, PERCENT=4#;

REAL CORE,READBACK,LOADER,J,I,DISKTOP,CARD,TYPE,MN,YR;

REAL ESUTEMP;

00060104
00060105
00060106

\$ SET OMIT = COOL

REAL LOWADR;

\$ POP OMIT

ARRAY INFO[*],SHAR[*],BUFFER[*],TEMPP[*],FILEX[*];

ARRAY FILEH[*],LABELS[*],ZERO[*],R[*];

INTEGER DY,LOADED,QX,XY,N,T,K,SYSNO,NUMWDS;

REAL TREAL=T;

\$ SET OMIT = NOT(COOL)

BOOLEAN RD,SKEDTQG,HARD,HARC,RC,FNDERR;

00060107
00060108
00060109

\$ POP OMIT

LABEL READCARD,FENCE,ESU,DATE,FILE,USE,DKA,DKB,STOP,EXIT,TIMER

,ERROR,START,TART,DIRECT,DIRECTORYTOP

,MRCLEAN,XLOADED,HANDLE,SYSTEMS;

00070000
00071000
00072000

\$ SET OMIT = NOT(COOL)

DEFINE REMOVE(REMOVE1)=BEGIN REMOVER(REMOVE1); GO REMOVED END#;

LABEL QUIT,REMOVED,PUNT,ARND,LP,UN;

REAL HDRO,RTI;

DEFINE RECSIZE=HDRO,[1:14]#,BLOCKSIZE=HDRO,[15:15]#;

RECSPERBLOCK=HDRO,[30:12]#,SEGSPERBLOCK=HDRO,[42:6]#;

BOOLEAN FIRSTCARD;

00073000
00074000
00075000

\$ POP OMIT

LABEL REMCD,MEMDUMP,BOMBOUT,REM,RECONSTR,GLOB;

SWITCH TYPESWITCH:=ERROR,FILE,FENCE,STOP,ESU,USE,DATE,USE,USE

,DIRECT,DIRECTORYTOP,SYSTEMS,REMCD,MEMDUMP

,REM,RECONSTR,XLOADED; % SAVE FOR LAST

ARRAY MONTHS[12]=0,31,59,90,120,151,181,212,243,273,304,334;

BOOLEAN SPOID,P1ID,P2ID,PRINTWAIT;

00081000
00082000
00083000

DEFINE KERNEL=16#; % SHOULD BE NUMBER OF FIRST WURDS = 1

ARRAY FIRST[17]=0,

"FILE0000",

"FENCE00",

"STOP0000",

"ESU00000",

00084000
00090000
00100000

00120000
00120010
00120099

00120100
00120101
00130000

00140000
00150000
00160000

00169000
00170000
00170010

00170020
00170030
00170049

00170050
00170100
00180000

00190000
00191000
00200000

00201000
00201100
00210000

00220000
00230000
00240000
00250000

	"USE00000",		00260000
	"DATE0000",		00270000
	"TYPE0000",		00280000
1	"OPTN0000",		00290000
2	"DIRECT00",		00300000
3	"DRCTRYTP",		00310000
4	"SYSTEMS0",		00320000
5	"REMOVECD",		00321000
6	"MEMDUMPO",		00321100
7	"REMOVEFO",		00321200
8	"RECONSTR",		00321300
9	"*0000000",		00330000
10	ARRAY OPTIONS[54] ← 0,		00340000
11	"DRA00000",	%47, USE DRUM A	00350000
12	"DRB00000",	%46, USE DRUM B	00360000
13	"BOJ00000",	%45	00370000
14	"EOJ00000",	%44	00380000
15	"OPEN0000",	%43, TYPE FILE OPEN	00390000
16	"TERMNATE",	%42, CALL TERMINATE PROCEDURE	00400000
17	"DATE0000",	%41, INITIALIZE DATE @ H/L	00410000
18	"TIME0000",	%40, INITIALIZE TIME @ H/L	00420000
19	"ONEBREAK",	%39, USE ONLY ONE BREAKOUT TAPE	00430000
20	"AUTOPRNT",	%38, AUTOMATICALLY PRINT PBT	00440000
21	"CLEARWRS",	%37, CLEAR WRITE READY STATUS @ TERMINAL	00450000
22	"DISCOND0",	%36, WRITE DISC. CODE ON TERMINAL	00460000
23	"CMLFILE",	%35, TYPE WHEN COMPILER FILES OPEN & CLOSE	00470000
24	"CLOSE000",	%34, TYPE FILE CLOSE	00480000
25	"ERRORMSG",	%33, ERROR MSGS WHEN PROGR RECOVERY USED	00490000
26	"RETO0000",	%32, TYPE MT RETENTION MESSAGES	00500000
27	"LIBMSG00",	%31, TYPE LIBRARY MESSAGES	00510000
28	"SCHEDMSG",	%30, TYPE SCHEDULE MESSAGES	00520000
29	"SECMSG00",	%29, TYPE FILE SECURITY MESSAGES	00530000
30	"DSKTOG00",	%28, PREVENT I/O BELOW USER DISK AREA	00540000
31	"RELTOG00",	%27, PREVENT DISK RELEASE STATEMENT	00550000
32	"PBDRELO0",	%26, PRINTER BACKUP DISK RELEASE	00560000
33	"CHECK000",	%25, CHECK MEMORY LINKS	00570000
34	"DISKMSG0",	%24, TYPE DISK ERROR MESSAGES	00580000
35	"DISKLOG0",	%23, DISK LOGGING	00590000
36	"LIBERR00",	%22, SUPPRESS LIBRARY ERROR MESSAGES	00600000
37	"PBDONLY0",	%21, GO TO PRINTER BACK-UP ONLY	00610000
38	"SAVEPBT0",	%20, DONT STACK FILES ON PB TAPES	00620000
39	"RSMMSG00",	%19, PRINT SET OR RESET MESSAGES	00630000
40	"AUTOUNLD",	%18, NO USER DISK WILL UNLOAD EXPIRED	00640000
41	"RNALLO0",	%17, RUN ALL DECKS(SHAREDISK)	00650000
42	"CODEOLAY",	%16, OLAY CODE TO ECM(AUXMEM)	00660000
43	"COREST00",	%15, JOB CORE ESTIMATES(STATISTICS)	00670000
44	"DATAOLAY",	%14, OLAY DATA TO ECM(AUXMEM)	00680000
45	"HALT0000",	%13, MAKES SYSTM HANG ON-SHOULD HL MSG	00690000
46	"REMBTE00",	%12, ENABLES DATACGM(TSS, IF NOT DCP)	00700000
47	"CEMESS00",	%11, LIBRARY MESSAGES FOR CANDE	00710000
48	"BATCHZIP",	%10, ZIP DECKS TO RUN ON BATCH(SHAREDISK)	00720000
49	"NOBATCH0",	%9, CONTROLS RUNNING OF BATCH JOBS ON TSS	00730000
50	"STOPTST",	%8, INHIBIT TAPE WRITE FAILURE TEST	00740000
51	"PNCHLOCK",	%7, LOCK PUNCH AFTER RELEASE	00750000
52	"CDONLY00",	%6, RUN WITH LDCNTRL ONLY	00760000
53	"PKTONLY0",	%5, RUN WITH PACKETS ONLY	00770000
54	"SEPARATE",	%4, SEPARATES PRINTOUT	00780000
55	"AUTOGEO0",	%3, AUTOMATIC INITIATION OF CANDE	00790000
56	"MOD3IO0",	%2, MODEL III I/O CHANNELS	00800000
57	"AUTOMESS",	%1, SMALL MESSAGES IF NOT SET	00810000

Data Documents/Inc.

```

"OPTN0000", %0
"+0000000";
ARRAY MESSAGE[75]:=
1 "DIRECTRY",
2 $ SET OMIT = COOL
3 " BUILT+",
4 $ POP OMIT
5 $ SET OMIT = NOT(COOL)
6 " CHECKD+",
7 $ POP OMIT
8 "INCORREC",
9 "T CARD+0",
10 " ERRO",
11 "R + ",
12 "DIRECTOR",
13 "YTOP NOT",
14 " SET+",
15 "LOADER I",
16 "CADED AL",
17 "READY+ ",
18 $ SET OMIT = NOT(COOL)
19 "COOL STA",
20 $ POP OMIT
21 $ SET OMIT = NOT(NOT COOL)
22 "COLD STA",
23 $ POP OMIT
24 "RTING.,.", "ENTER OK", " TO CONT", "INUE.,.",
25 "NEED VAL",
26 "ID ESU C",
27 "ARD+", "ENTER OK", " TO REMO", "VE DECKS", ".,.",
28 "DRCTRYTP",
29 " CARD NO",
30 "T FIRST+",
31 "DUPLICAT",
32 "E DRCTRY",
33 "TYP CARD",
34 "-MUST RE",
35 "START+ ", % 31
36 "MEMDUMP ", "TOD LARG", "E=NOT LO", "ADED+ ", %32
37 "MEMDUMP ", "OVERWRIT", "TTEN BY ", "KERNEL+ ", %36
38 "ENTER OK", " TO RECO", "NSTRUCT ", "BAD HEAD", "ERS.+ ",
39 " /", " ", "BLKSZ= ", " ", "RE", "CSZ= ",
40 " ,RPB= ", " ,SPB", "= +", % 45
41 "SEGMENT ", "ZERO MAY", " BE BAD+", % 53
42 "DIRECTOR", "YTOP MAY", " BE BAD+", % 56
43 " /", " ", "REMOVED+", % 59
44 "BAD DIRE", "CT VALUE", " ", % 62
45 "CHECK DE", "CLARED F", "ILE ADDR", "ESSES+ ", % 65
46 "NO. OF E", "SUS TO B", "E CHANGE", "D FROM ", " ", %69
47 "++++++";
ARRAY LOADBUTTON[15]+
48
49 OCT0441000401570421, % 1 20 01050000
50 OCT0157000000104411, % 2 21 01060000
51 OCT0211001441310055, % 3 22 01070000
52 OCT4155124500004425, % 4 23 01080000
53 OCT0060013101600064, % 5 24 01090000
54 OCT4441010402530305, % 6 25 01100000
55 OCT0100023441310055, % 7 26 01110000
56 OCT0062011441310055, % 8 27 01120000
57 OCT0066013441310055, % 9 30 01130000

```

```

00820000
00870000
00880000
00890000
00900000
00910000
00910050
00920000
00930000
00930050
00940000
00950000
00960000
00970000
00980000
00990000
01000000
01010000
01020000
01030000
01031999
01032000
01032001
01032999
01033000
01033001
01034000
01034100
01034200
01034300
01034400
01034500
01034600
01034700
01034800
01034900
01034920
01034940
01034950
01034960
01034970
01034990
01035000
01035010
01035020
01035030
01035040
01035050
01035060
01039000
01040000
01050000
01060000
01070000
01080000
01090000
01100000
01110000
01120000
01130000

```

	OCT0072015441310055	% 10	31	01140000
	OCT0076017441310055	% 11	32	01150000
	OCT151400000477C0200	% 12	33	01160000
1	OCT0441100401004441	% 13	34	01170000
2	OCT0253010420527405	% 14	35	01180000
3	OCT7405005101002411	% 15	36	01190000
4	BOOLEAN STREAM PROCEDURE OK(B); VALUE B;			01195000
5	BEGIN SI:=B; L: IF SC=" " THEN BEGIN SI:=SI+1; GO L END;			01195100
6	IF SC="0" THEN SI:=SI+1 ELSE GO AWAY;			01195200
7	IF SC="K" THEN TALLY:=1;			01195300
8	AWAY:OK:=TALLY;			01195400
9	END OK;			01195500
10	REAL STREAM PROCEDURE DECI(V); VALUE V;			01195700
11	BEGIN SI:=LOC V; DI:=LOC DECI; DS:=8 DEC;			01195800
12	END DECI;			01195900
13	REAL STREAM PROCEDURE SEARCH(CARD,FIRST,N); VALUE CARD,N;			01200000
14	BEGIN LABEL L,EXIT;			01210000
15	SI:=CARD; SII:=SI-N; DII:=LOC CARD; DS:=8LIT"0";			01220000
16	DI:=LOC CARD; N(DS:=CHR);			01225000
17	L: SI + FIRST; SI + SI+8; TALLY + TALLY+1;			01230000
18	IF SC = " " THEN BEGIN TALLY + C; GO TO EXIT END;			01240000
19	DI + FIRST; DI + DI+8; FIRST + DI;			01250000
20	SI:=LOC CARD;			01260000
21	IF B SC NEQ DC THEN GO L;			01270000
22	EXIT:SEARCH + TALLY;			01280000
23	END;			01290000
24	REAL STREAM PROCEDURE SCAN(LSTADR,TOKEN);			01290100
25	BEGIN LABEL L,XXIT; LOCAL SV,SVR;			01290200
26	DI:=TOKEN; DS:=8LIT"00000000"; DI:=DI-8;			01290250
27	SI:=LSTADR; SII:=SI+5; SIII:=SC;			01290300
28	TALLY:=3; SCAN:=TALLY;			01290400
29	L: IF SC=" " THEN BEGIN SI:=SI+1; GO L; END;			01290500
30	IF SC=ALPHA THEN			01290600
31	IF SC LSS "0" THEN BEGIN % ALPHA			01290700
32	TALLY:=7; DI:=DI+1;			01290800
33	7(DS:=CHR; TALLY:=TALLY+63; IF SC=ALPHA THEN ELSE JUMP OUT);			01290900
34	SV:=TALLY; SV(DS:=LIT" ");			01291000
35	TALLY:=1; SCAN:=TALLY;			01291100
36	END			01291200
37	ELSE BEGIN % NUMBER			01291300
38	TALLY:=0;			01291400
39	7(SI:=SI+1; TALLY:=TALLY+1);			01291500
40	IF SC=ALPHA THEN			01291600
41	IF SC GEQ "0" THEN ELSE JUMP OUT TO XXIT			01291700
42	ELSE JUMP OUT);			01291800
43	SV:=TALLY; SII:=SI-SV;			01291900
44	DS:=SV OCT;			01292000
45	TALLY:=2; SCAN:=TALLY;			01292100
46	END			01292200
47	ELSE			01292300
48	IF SC="'" THEN BEGIN			01292400
49	SI:=SI+1; TALLY:=7; DI:=DI+1;			01292500
50	8(IF SC NEQ "'" THEN BEGIN DS:=CHR; TALLY:=TALLY+63; END			01292600
51	ELSE BEGIN SI:=SI+1; JUMP OUT; END);			01292700
52	IF TOGGLE THEN GO XXIT;			01292800
53	SV:=TALLY; SV(DS:=LIT" "); SVR:=SI;			01292900
54	SI:=LOC SV; SI:=SI+7; IF SC="7" THEN GO XXIT			01293000
55	ELSE BEGIN TALLY:=1; SCAN:=TALLY;			01293100
56	SI:=SVR; END;			01293200
57	END ELSE BEGIN			01293300

Data Documents/Inc

	IF SC="%" THEN BEGIN TALLY:=4; SCAN:=TALLY;	01293400
	GO XXIT; END;	01293500
1	DI:=DI+7;	01293600
2	DS:=CHR;	01293700
3	TALLY:=0; SCAN:=TALLY;	01293800
4	END;	01293900
5	XXIT: SV:=SI; SI:=LOC SV; DI:=LSTADR; DS:=WDS;	01294000
6	END;	01294100
7	REAL STREAM PROCEDURE ISOLATE(CARD);	01300000
8	BEGIN LABEL L,A,X,E;	01310000
9	LOCAL T;	01320000
10	SI ← CARD; SI ← SI+5; SI ← SC;	01330000
11	L: IF SC = " " THEN BEGIN SI ← SI+1; GO TO L END;	01340000
12	IF SC = "" THEN	01350000
13	BEGIN SI ← SI+1;	01360000
14	17(SI ← SI+1; TALLY ← TALLY+1;	01370000
15	IF SC = "" THEN	01380000
16	BEGIN T ← SI; DI ← T; DS ← LIT " ";	01390000
17	JUMP OUT TO X	01400000
18	END;);	01410000
19	E: TALLY ← 0; GO TO X;	01420000
20	END;	01430000
21	IF SC = ALPHA THEN BEGIN	01440000
22	17(SI ← SI+1; TALLY ← TALLY+1;	01450000
23	IF SC = ALPHA THEN GO TO A;	01460000
24	JUMP OUT TO X; A:);	01470000
25	GO TO E;	01480000
26	END ELSE BEGIN SI ← SI+1; TALLY ← 1 END;	01490000
27	X: ISOLATE ← TALLY;	01500000
28	T ← SI;	01510000
29	DI ← CARD; SI ← LOC T; DS ← WDS;	01520000
30	END;	01530000
31	SAVE PROCEDURE SPECIAL(C); VALUE C; REAL C;	01540000
32	BEGIN IF ISOLATE(CARD) ≠ 1 THEN GO TO ERROR;	01550000
33	STREAM(C;CARD);	01560000
34	BEGIN SI ← CARD; SI ← SI-1;	01570000
35	DI ← LOC C; DI ← DI+7;	01580000
36	IF SC ≠ DC THEN TALLY ← 1;	01590000
37	C ← TALLY;	01600000
38	END;	01610000
39	IF P THEN GO TO ERROR;	01620000
40	END;	01630000
41	SAVE PROCEDURE IDERR(IOD); VALUE IOD; REAL IOD; FORWARD;	01640000
42	SAVE REAL PROCEDURE EQM; FORWARD;	01650000
43	SAVE PROCEDURE WAITFORPRINTERFINISH(U); VALUE U; REAL U;	01651000
44	BEGIN PRINTWAIT:=TRUE;	01652000
45	DO UNTIL EQM=U;	01653000
46	PRINTWAIT:=FALSE;	01654000
47	END;	01655000
48	SAVE PROCEDURE IO(ADDRESS,IOD); VALUE ADDRESS,IOD; REAL ADDRESS,IOD;	01656000
49	BEGIN REAL T;	01660000
50	LABEL L;	01670000
51	IOD ← ADDRESS INX IOD;	01680000
52	L: P([IOD],IIO);	01690000
53	T ← EQM;	01700000
54	IF T.[30:3] ≠ 0 THEN	01710000
55	IF QX THEN GO TO L ELSE QX := 2;	01720000
56	IF ADDRESS ≥ 0 THEN	01730000
57	IF T.[26:4] ≠ 0 THEN	01735000
		01740000

```

1      IF T.[29:1] AND T.[3:5]=6 AND M[ADDRESS]=040      01751000
2      THEN BEGIN      01752000
3          M[ADDRESS]:=0;      01753000
4          M[ADDRESS+1]:=0;      01754000
5          GO L;      01755000
6          LND      01756000
7      ELSE IF (T.[3:5]=22 OR T.[3:5]=26)      01757000
8          THEN IF T.[27:1]      01758000
9              THEN BEGIN      01759000
10                 WAITFORPRINTERFINISH(T.[3:5]);      01760000
11                 IOU:=0&T[3:3:5]&1[18:47:1]      01761000
12                 &1[29:44:4];      01762000
13                 GO L;      01763000
14                 END      01764000
15             ELSE WAITFORPRINTERFINISH(T.[3:5])      01765000
16             ELSE IF T.[29:1] AND T.[3:5]=10 AND      01767000
17                 M[ADDRESS]=0      01768000
18                 THEN      01769000
19                     ELSE BEGIN IOERR(IOU); GO L; END;      01770000
20     END;      01850000
21     SAVE PROCEDURE SPOINOUT(WHICH); VALUE WHICH; INTEGER WHICH;      01851000
22     BEGIN      01852000
23         IF SPOIC THEN      01853000
24             IO(MESSAGE INX ABS(WHICH),@7400000000000000&WHICH[24:1:1]);      01854000
25         END SPOINOUT;      01855000
26     DEFINE KEYIN(KEYIN1)=SPOINOUT(NABS(KEYIN1));#;      01856000
27     DEFINE SPOUT=SPOINOUT#;      01857000
28     SAVE PROCEDURE IOERR(IOU); VALUE IOU; REAL IOU;      01860000
29     BEGIN REAL T;      01870000
30         IOU + IOU.[3:5];      01890000
31         T + IF IOU=30 THEN "SPU" ELSE      01900000
32             IF IOU=10 THEN "CRA" ELSE      01910000
33             IF IOU= 6 THEN "DKA" ELSE      01920000
34             IF IOU=22 THEN "LPA" ELSE      01921000
35             IF IOU=26 THEN "LPB" ELSE      01922000
36             IF IOU= 4 THEN "DRA" ELSE "XXX";      01930000
37         STREAM(T,A+[MESSAGE[4]]);      01940000
38         BEGIN SI + LOC T; SI + SI+5; DS + 3CHR; END;      01950000
39         SPOUT(4);      01960000
40         IF IOU =10 THEN      01970000
41             BEGIN DO UNTIL P(RRR).[24:1] = 0;      01980000
42                 DO UNTIL P(RRR).[24:1];      01990000
43             END ELSE      02000000
44             IF IOU LSS 22 THEN GO BOMBOUT;      02000100
45     END IOERR;      02020000
46     SAVE PROCEDURE DISKIO(A,C); VALUE A;      02030000
47     ARRAY C[*];      02040000
48     REAL A;      02050000
49     BEGIN BOOLEAN K;      02060000
50         C := C INX NOT 0;      02070000
51         IF R:=A.[1:1] THEN A := ABS (A);      02080000
52         STREAM(A,X+FLAG(C));      02090000
53         BEGIN SI + LOC A; DS + 8 DEC END;      02100000
54         IF R THEN IO(C,@140360140100000&NUMWDS[8:38:10]      02110000
55             &((NUMWDS+29)DIV 30)[27:42:6]) ELSE      02120000
56         BEGIN      02130000
57             IO(C,@140360100100000);      02140000
58             M[READBACK] + M[C];      02150000
59             IO(READBACK,@140360140100000);      02160000
60             STREAM(READBACK;X+FLAG(C));      02170000

```


Data Documents/Inc.

1	BEGIN SI ← READBACK; SI ← SI+8; DI ← DI+8;	02180000
2	5(IF 48 SC ≠ DC THEN TALLY ← 1);	02190000
3	READBACK ← TALLY;	02200000
4	END;	02210000
5	IF P THEN IOERR(⊖1400000000000000);	02220000
6	END;	02230000
7	END;	02240000
8	SAVE REAL PROCEDURE SCANWORD(TABLE); ARRAY TABLE[*];	02250000
9	BEGIN REAL T;	02260000
10	T:=IF(M[CARD] EQV 0)≠NOT 0 THEN KERNEL ELSE ISCLATE(CARD);	02270000
11	SCANWORD:=IF T=KERNEL THEN T ELSE SEARCH(CARD, TABLE, T);	02280000
12	END;	02290000
13	SAVE REAL PROCEDURE SCANUMBER;	02300000
14	BEGIN REAL T;	02310000
15	T ← ISCLATE(CARD);	02320000
16	STREAM(R←0; T, CARD);	02330000
17	BEGIN DI←DI-T; CARD ← DI;	02340000
18	SI ← CARD;	02350000
19	DI ← LOC R;	02360000
20	DS ← T OCT;	02370000
21	END;	02380000
22	SCANUMBER ← P;	02390000
23	END;	02400000
24	%	02410000
25	\$ SET OMIT = NOT(COCL)	02410500
26	SAVE BOOLEAN PROCEDURE LOOKFOR(L1, L2); VALUE L1, L2; REAL L1, L2;	02411000
27	BEGIN REAL I; LABEL XXIT;	02411100
28	FOR I:=0 STEP 2 UNTIL RTI DO	02411200
29	IF (L1 EQV RT[I])≠NOT 0 OR RT[I] LSS 0 THEN	02411300
30	IF (L2 EQV RT[I+1])≠NOT 0 OR RT[I+1] LSS 0 THEN	02411400
31	BEGIN LOOKFOR:=TRUE; I:=RTI; GO XXIT; END;	02411500
32	XXIT:	02411600
33	END;	02411700
34	\$ POP OMIT	02411750
35	%	02411800
36	SAVE INTEGER PROCEDURE PACKER;	02420000
37	BEGIN;	02430000
38	STREAM (A:=[BUFFER[9]], B:=[PACKER]);	02440000
39	BEGIN SI := A; SI := SI + 4;	02450000
40	DI := B; DI := DI + 6;	02460000
41	4(SKIP 3 SB; 3(IF SB THEN DS:=SET ELSE SKIP DB; SKIP SB));	02470000
42	END;	02480000
43	END;	02490000
44	%	02500000
45	SAVE PROCEDURE LOADERS;	02510000
46	BEGIN LABEL TOPPER, NXTCRD, EXIT;	02520000
47	INTEGER N, SIZE;	02521000
48	IF (TYPE=KERNEL AND LOADED)	02530000
49	OR (TYPE≠KERNEL AND (LOADED.[46:1])) THEN	02531000
50	BEGIN SPOUT(9);	02540000
51	GO BOMBOUT;	02550000
52	END;	02560000
53	IF TYPE NEQ KERNEL THEN GO NXTCRD;	02561000
54	TOPPER: STREAM(B:=[BUFFER[1]], A:=N; B:=BUFFER[9].[6:9],	02570000
55	D←PACKER INX BUFFER);	02580000
56	BEGIN SI ←B; DS ← A WDS; END;	02590000
57	SIZE:=SIZE+N;	02591000
58	NXTCRD: IO(BUFFER INX 0, ⊖2401200400000000);	02600000
59	IF BUFFER[0].[4:44]=0 THEN GO TO TOPPER;	02610000
60	SIZE:=(SIZE+29) DIV 30;	02611000

Data Documents/Inc.

33788

```

IF (TYPE=KERNEL AND (1+SIZE) GTR MEMDISKADR) THEN 02612000
BEGIN
  SPCUT(35); 02612100
  LOADED:=LOADED OR 2; 02612300
END ELSE 02612400
IF (TYPE≠KERNEL AND (MEMDISKADR+SIZE) GTR ESPBOTTOM) THEN 02612500
BEGIN 02612600
  SPCUT(32); 02612700
  GO EXIT; 02612800
  02612900
END; 02613000
TEMPP := BUFFER; 02620000
BUFFER := 111 INX BUFFER; 02630000
BUFFER[C] := IF TYPE=KERNEL THEN 1 ELSE DECI(MEMDISKADR); 02640000
LOADED := LOADED OR 02650000
  (IF TYPE=KERNEL THEN 1 ELSE 2); 02651000
IO(BUFFER INX 0,@1400000000000000&SIZE[27:42:6]); 02660000
BUFFER := TEMPP; 02670000
EXIT; 02671000
END; 02680000
$ SET OMIT = NOT COCL 02689999
SAVE PROCEDURE REMOVER(L); VALUE L; INTEGER I; 02690000
BEGIN 02700000
  STREAM(A:=[LABELS[I]],X:=(L=0),L,ADR:=J*(L NEQ 0)* 02710000
    ((80-I) DIV 2),MONTHS); 02711000
  BEGIN 55(DS:=LIT" "); DS:=LIT"+"; DI:=MONTHS; 02712000
    SI:=A; X(DS:=8CHR;DS:=LIT"/"; DS:=8CHR; JUMP OUT TO RMVE); 02720000
    SI:=SI+1;DS:=7 CHR; DS:=LIT"/";SI:=SI+1;DS:=ZCHR; 02720200
  RMVE:DS:=9LIT" REMOVED("; 02730000
    CI:=CI+L; 02730100
    GO NAM; 02730120
    GO HDR; 02730140
    GO DECK; 02730160
  NAM :DS:=8LIT"BAD NAME";GO FIN; 02730200
  HDR :DS:=7LIT"BAD HDR"; GO FIN; 02730300
  DECK:DS:=4LIT"DECK"; 02730400
  FIN :DS:=9LIT") AT ADR="; 02730500
    SI:=LOC ADR; X:=DI; DS:=9DEC; DI:=X; DS:=7FILL; 02730600
  END; 02740000
  LABELS[I]:=014; 02750000
  IF NOT(P1IO OR P2IO) THEN IF NOT SPOIO THEN ELSE 02751000
    IO(MONTHS INX 0,@7400000000000000) ELSE 02755000
    IO(MONTHS INX 0,IF P1IO THEN @540070004000000 ELSE 02757000
      @640070004000000); 02760000
  END; 02770000
  $ POP OMIT 02770001
  SAVE REAL PROCEDURE EOM; 02780000
  BEGIN GO TO TIMER END; 02790000
  *****START HERE***** 02800000
  TART :@20: GX + 1; GO TO START; 02810000
  TIMER :@22:P(INI); GO TO TIMER; % TIME INTERVAL 02820000
    :@24:GO TO TIMER; % KEYBOARD REQUEST 02830000
    :@25:P(22); GO GLOB; % LPA FINISHED 02840000
    :@26:P(26); GO GLOB; % LPB FINISHED 02850000
    :@27:P(@14,LOD,RTN); % I/O #1 FINISHED 02860000
    :@30:P(@15,LOD,RTN); % I/O #2 FINISHED 02870000
    :@31:P(@16,LOD,RTN); % I/O #3 FINISHED 02880000
    :@32:P(@17,LOD,RTN); % I/O #4 FINISHED 02890000
    :@34:GO TO TIMER; % INQUIRY REQUEST 02900000
  *****CODE STARTS HERE***** 02910000
  START:; P(64,STS); 02920000

```

Data Documents, Inc.

```

DISKTUP ← -1;                                02930000
NUMWDS := 30;                                02940000
CORE ← P(CORE,LOU).[33:15]+3;                02950000
1  STREAM(CORE);                              02960000
2     BEGIN DS ← 8 LIT "0"; SI ← CORE;        02970000
3         32(DS ← 32 WDS);                    02980000
4     END;                                     02990000
5     SHAR := [M(CORE)]&31[8:38:10];          03000000
6     CORE := CORE + 31;                      03010000
7     LOADER ← CORE;                         03020000
8     CORE ← CORE + 30;                      03030000
9     INFO ← [M(CORE)]&30[8:38:10];          03040000
10    CORE ← CORE + 40;                      03050000
11    FILEH ← [M(CORE)]&30[8:38:10];         03060000
12    CORE ← CORE + 40;                      03070000
13    FILEX := [M(CORE)]&450[8:38:10];       03080000
14    CORE := CORE + 450;                    03090000
15    LABELS ← [M(CORE)]&30[8:38:10];       03100000
16    CORE ← CORE + 40;                      03110000
17    READBACK ← CORE;                      03120000
18    CORE ← CORE + 40;                      03130000
19    ZERO := [M(CORE)]&30[8:38:10];        03130100
20    CORE := CORE + 40;                    03130200
21    RTI := [M(CORE)]&300[8:38:10];        03130300
22    CORE := CORE + 310;                   03130400
23    SPIO := P(KRR).[22:1];                03131000
24    P1IO := P(RRR).[27:1];                03131050
25    P2IO := P(RRR).[26:1];                03131100
26    $ SET OMIT = COOL                     03131299
27    LOWADR := 99999999; % INITIALIZE       03131300
28    $ POP OMIT                             03131401
29    $ SET OMIT = NOT(COOL)                 03140000
30    DISKIU(=0,SHAR);                      03150000
31    FOR I := 13 STEP 5 UNTIL 28            03160000
32        DO SHAR[I] := SHAR[I+1] := 0; %CLEAR ALL INT 03170000
33    INFO[0] := @40;                       03180000
34    INFO[1] ← 0;                          03185000
35    IO(=(INFO INX 0),@1400040000000000); 03190000
36    SYSNO := INFO[1].[4:2]; %GET SYSTEM NUMBER 03200000
37    FIRSTCARD := TRUE;                    03200025
38    RTI := -1;                             03200030
39    $ POP OMIT                             03200050
40    I ← 0;                                 03210000
41    BUFFER := [M(CORE)]&10[8:38:10];       03230000
42    %                                       03240000
43    READCARD: CARD ← CORE; %READ CARD      03250000
44    IO(CARD,@2400000400000000);           03260000
45    HANDLE: STREAM(C:=CARD+10) DS := 2 LIT "%-"; 03270000
46    TYPE := SCANWORD(FIRST);               03280000
47    $ SET OMIT = NOT(COOL)                 03280099
48    IF FIRSTCARD THEN                     03280100
49        IF TYPE NEQ 10 AND TYPE NEQ KERNEL THEN 03280200
50        BEGIN                             03280300
51            SPOUT(24);                     03280400
52            GO TO ERROR;                   03280500
53        END;                               03280600
54    $ POP OMIT                             03280601
55    GO TYPESWITCH(TYPE);                   03280700
56    GLOB: IF PRINTWAIT THEN P(RTN) ELSE P(DEL); 03280800
57    GO TIMER;                              03280900

```

```

%
ERROR:      SPOUT(2);                                03290000
            DO UNTIL P(HRR),[24:1] = 0;              03300000
            DO UNTIL P(RRR),[24:1];                 03310000
            P(64,STS);                                03320000
            GO READCARD;                              03330000
            GO READCARD;                              03340000
%
MEMDUMP:    03350000
XLOADED:    LOADERS; XLOAD LOADER DECK              03351000
            CARD := CURE;                            03360000
            GO TO HANDLE;                            03370000
%
REM:        03380000
$ SET OMIT = COOL                                  03390000
GO ERROR;                                          03391000
$ POP OMIT                                         03391100
$ SET OMIT = NOT(COOL)                             03391200
CARD:=CORE;                                        03391300
ID(CARD,@2400000400000000);                       03391400
STREAM(C:=CARD+10); DS:=2L1["%-"];                03391500
LP:         IF P(SCN(HDRO)) NEQ NAAME THEN          03391600
            IF HDRO NEQ "=" THEN GO ERROR          03391700
            ELSE RT[RTI:=RTI+1]:=-1              03391800
            ELSE RT[RTI:=RTI+1]:=HDRO;            03391900
            IF P(SCN(HDRO)) NEQ DELIM THEN          03392000
            IF RT[RTI]="STOP " THEN BEGIN RTI:=RTI-1; GO READCARD; END; 03392100
            ELSE GO ERROR                          03392200
            ELSE                                     03392300
            IF HDRO NEQ "/" THEN GO ERROR;         03392400
            IF P(SCN(HDRO)) NEQ NAAME THEN          03392500
            IF HDRO NEQ "=" THEN GO ERROR          03392600
            ELSE RT[RTI:=RTI+1]:=-1              03392700
            ELSE RT[RTI:=RTI+1]:=HDRO;            03392800
            IF P(SCN(HDRO),DUP) NEQ PERCENT THEN    03392900
            IF P NEQ DELIM THEN GO ERROR           03393000
            ELSE GO LP                             03393100
            ELSE BEGIN P(DEL); GO REM; END;        03393200
$ POP OMIT                                         03393300
%
SYSTEMS:    SPECIAL ("=");                          03393400
            SHAR[0] := SCANUMBER ;                 03393500
            IF SHAR[0] GTR 4 OR SHAR[0] LSS 1 THEN 03393600
            BEGIN SHAR[0] := 1;                    03393700
            GO TO ERROR;                            03400000
            END;                                    03410000
            GO TO READCARD;                         03420000
%
DIRECTORYTOP:
$ SET OMIT = NOT(COOL)                             03430000
            IF NOT FIRSTCARD THEN                  03440000
            BEGIN                                    03450000
                SPOUT(27);                          03460000
                GO BOMBOUT;                          03470000
            END;                                    03480000
$ POP OMIT                                         03480050
            DISKTOP:=SCANUMBER;                     03480100
            J ← DISKTOP+19;                          03480200
$ SET OMIT = NOT(COOL)                             03480300
            DISKIO(-(DISKTOP-SYSNO),INFO);         03480400
            FIRSTCARD:=FALSE;                      03480500
            FIRSTCARD:=FALSE;                      03490000
            FIRSTCARD:=FALSE;                      03500000
            FIRSTCARD:=FALSE;                      03510000
            FIRSTCARD:=FALSE;                      03510100

```

Data Documents/Inc.

	\$ POP OMIT		03510101
	GO TO READCARD;		03520000
1	%		03530000
2	DIRECT:	INFO[4] ← SHARE[4]+SCANUMBER;	03540000
3		GO TO READCARD;	03550000
4	%		03560000
5	FENCE:	SPECIAL("=");	03570000
6		INFO[19]+@10000-(P(DUP)-SCANUMBER) DIV 1024×1024;	03580000
7		IF INFO[19]<@20000 THEN INFO[19]+@20000 ELSE	03590000
8		IF INFO[19]>@70000 THEN INFO[19]+@70000;	03600000
9		GO TO READCARD;	03610000
10	%		03620000
11	ESU:	ESUTEMP:=SCANUMBER;	03630000
12		SPECIAL("%");	03635000
13		GO READCARD;	03640000
14	%		03650000
15	DATE:	MN ← SCANUMBER;	03660000
16		IF MN < 1 OR MN > 12 THEN GO TO ERROR;	03670000
17		SPECIAL("/");	03680000
18		DY ← SCANUMBER;	03690000
19		SPECIAL("/");	03700000
20		YR ← SCANUMBER;	03710000
21		DY ← MONTHS[MN-1]+DY;	03720000
22		IF YR MOD 4 = 0 AND (YR MOD 100 ≠ 0 OR YR MOD 400 = 0)	03730000
23		AND MN ≥ 3 THEN DY ← DY+1;	03740000
24		DY ← YR MOD 100 × 1000+DY,	03750000
25		STREAM(D+[DY],I+[INFO[1]]);	03760000
26		BEGIN SI ← D; DS ← 8 DEC END;	03770000
27		GO READCARD;	03780000
28	%		03810000
29	USE:	IF TYPE NEQ 8 THEN XY:=SCANWORD(OPTIONS)	03820000
30		ELSE XY:=48-SCANUMBER;	03820100
31		IF XY=0 THEN GO ERROR;	03830000
32		TREAL:=1;	03880000
33		FOR K:=2 STEP 1 UNTIL XY DO TREAL:=O&TREAL[1:2:46];	03890000
34		INFO[0]:=INFO[0] OR TREAL;	03900000
35		SPECIAL("%");	03905000
36		GO READCARD;	03910000
37	%		03920000
38	FILE:		03930000
39	\$ SET OMIT = NOT(COOL)		03940000
40	GO TO ERROR;		03950000
41	\$ POP OMIT		03950050
42	\$ SET OMIT = COOL		03960000
43	IF P(SCN(LABELS[28-I])) NEQ NAAME THEN GO ERROR;		03971000
44	IF P(SCN(N)) NEQ DELIM THEN GO ERROR		03972000
45	ELSE IF N NEQ "/" THEN GO ERROR;		03973000
46	IF P(SCN(LABELS[29-I])) NEQ NAAME THEN GO ERROR;		03974000
47	IF DISKTOP LSS 0 THEN BEGIN SPOUT(6); GO ERROR; END;		03975000
48	FILEH[0]:=(*P(DUP))&30[1:34:14]&30[15:33:15]&		03975100
49	1[30:36:12]&1[42:42:6];		03975200
50	IF P(SCN(N)) NEQ DELIM THEN GO ERROR;		03976000
51	IF P(SCN(FILEH[9])) NEQ NUMBER THEN GO ERROR;		03977000
52	IF FILEH[9] GTR 20 THEN GO ERROR;		03978000
53	IF P(SCN(N)) NEQ DELIM THEN GO ERROR;		03979000
54	IF P(SCN(FILEH[8])) NEQ NUMBER THEN GO ERROR;		03980000
55	IF P(SCN(N)) NEQ DELIM THEN GO ERROR;		03981000
56	IF P(SCN(N)) NEQ NUMBER THEN GO ERROR		03982000
57	ELSE IF N GTR 999 THEN GO ERROR		03983000
	ELSE FILEH[3]I:=DY&DY[12:30:18]&N[2:38:10];		03984000

```

IF (K:=P(SCN(N))) = PERCENT THEN 03985000
ELSE 03986000
BEGIN IF K NEQ DELIM THEN GO ERROR; 03987000
IF P(SCN(N)) NEQ NUMBER THEN GO ERROR 03988000
ELSE FILEH[0]:=(+P(DUP))& 03989000
N[1:34:14]; 03989100
K:=N; 03989200
IF P(SCN(N)) NEQ DELIM THEN GO ERROR; 03990000
IF P(SCN(N)) NEQ NUMBER THEN GO ERROR 03991000
ELSE FILEH[0]:=(+P(DUP))& 03992000
N[15:33:15]; 03992100
FILEH[0]:=(+P(DUP))&((N+29) DIV 30)[42:42:6] 03992200
&(N DIV K)[30:36:12]; 03992300
IF P(SCN(N)) NEQ PERCENT THEN GO ERROR; 03993000
END; 03994000
N:=FILEH[9]; 03996000
FOR K + 1 STEP 1 UNTIL N DO 04180000
BEGIN CARD + CORE; 04190000
IO(CARD,@240000040000000); 04200000
IF SCANWRD(FIRST) ≠ 0 THEN GO TO ERROR; 04210000
CARD + CORE; 04220000
FILEH[9+K] + SCANUMBER; 04230000
SPECIAL("%"); 04240000
IF FILEH[9+K] LSS LOWADR THEN 04241000
IF FILEH[9+K] NEQ DISKTOP+4 THEN LOWADR:=FILEH[9+K]; 04245000
END; 04250000
FOR K + K STEP 1 UNTIL 20 DO 04260000
FILEH[9+K] + 0; 04270000
FILEH[4]:=#44; 04280200
FILEH[7] + FILEH[8]XN-1; 04290000
I + I+2; 04300000
DISKIO(J=(I DIV 2),FILEH); 04310000
IF I = 30 THEN 04320000
BEGIN DISKIO(J,LABELS); 04330000
J + J+16; 04340000
I + 0 04350000
END; 04360000
GO TO READCARD; 04370000
$ POP OMIT 04370050
RECONSTR; 04371000
$ SET OMIT = NOT COOL 04371100
RC:=HARC:=TRUE; GO READCARD; 04371200
$ SET OMIT = COOL 04371300
GO ERRCR; 04371400
$ RESET OMIT 04371500
REMC0; 04372000
$ SET OMIT = NOT COOL 04372009
RD:=HARD:=TRUE; GO READCARD; 04372010
$ SET OMIT = COOL 04372015
GO TO ERRUR; 04372020
$ RESET OMIT 04372021
X!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!ND MORE CARDS!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! 04380000
STOP; SHAR[4]:=INFO[4]; 04390500
INFO[16]:=15; 04395000
IF SPOIC THEN 04395100
BEGIN 04395200
SPOUT(12); 04395300
KEYIN(12); 04395400
IF NOT OK([MESSAGE[12]] INX 0) THEN 04395500
BOMBOUT; 04395550

```

Data Documents/Inc.

```
BEGIN STREAM(B:=[MESSAGE[12]]); 04395600
  DS:=24 LIT "COOL/COLD START ABORTED+"; 04395700
SPOUT(12); 04395800
DO UNTIL FALSE; 04395900
END; 04396000
STREAM(A:=INFO[4],SV:=0,N:=[MESSAGE[27]]); 04396050
  BEGIN DS:=20LIT"USER DISK BEGINS AT "; SI:=LOC A; 04396100
    DS:=8DEC; SV:=DI; DI:=DI-8; DS:=7FILL; DI:=SV; 04396150
    DS:=LIT"+"; 04396200
  END; 04396250
  SPOUT(27); 04396300
  END; 04396400
$ SET OMIT = COOL 04397000
  IF (INFO[4] LEQ DISKTOP OR (N:=LOWADR LSS INFO[4])) THEN 04397100
    BEGIN IF SPOID THEN 04397200
      BEGIN SPOUT(62); IF N=1 THEN SPOUT(65); END; 04397300
      GO BUMBOUT; 04397400
    END; 04397500
  IF DISKTOP < 0 THEN 04400000
    BEGIN 04410000
      SPOUT(6); 04420000
      GO TO ERROR; 04430000
    END; 04440000
$ POP OMIT 04440001
  IF ESUTEMP LEQ 0 THEN 04441000
    BEGIN 04441100
      SPOUT(17); 04441200
      GO TO ERROR; 04441300
    END; 04441400
    SHAR[1] + DISKTOP; 04450000
    IF SHAR[0] GTR 4 OR SHAR[0] LSS 1 THEN SHAR[0] := 1; 04460000
$ SET OMIT = NOT COOL 04470004
***** LOOK AT SEGMENT ZERO ***** 04470005
  IF SHAR[1] GTR 10000 OR SHAR[1] LSS 50 OR 04470010
    SHAR[4] GTR 100000 OR SHAR[4] LSS 50 04470015
    THEN BEGIN SPOUT(53); GO BUMBOUT; END; 04470020
  FOR N:=C STEP 1 UNTIL 29 DO 04470030
    IF P(SHAR[N],TOP,XCH,DEL) THEN ELSE 04470035
      BEGIN FNDERR:=TRUE; SHAR[N]:=0; END; 04470040
  FOR N:=10 STEP 5 UNTIL SHAR[0]*5+5 DO 04470045
    BEGIN 04470050
      IF ((SHAR[N] OR SHAR[N+1]) AND @3700000000000000) NEQ 0 04470055
        THEN BEGIN FNDERR:=TRUE; SHAR[N]:=SHAR[N+1]:=0; END; 04470060
      IF ((SHAR[N+3] OR SHAR[N+4]) AND @3700000000000000) NEQ 0 04470065
        THEN BEGIN FNDERR:=TRUE; SHAR[N+3]:=SHAR[N+4]:=0; END; 04470070
    END; 04470075
    IF FNDERR THEN SPOUT(53); 04470080
***** END OF SEGMENT ZERO EXAMINATION ***** 04470085
$ POP OMIT 04470090
  DISKIO(0,SHAR); 04470095
$ SET OMIT = NOT COOL 04470099
  IF ESUTEMP NEQ INFO[2] THEN 04470100
    IF SPOID THEN 04470105
      BEGIN 04470110
        STREAM(A:=INFO[2],D:=0,R:=ESUTEMP,C:=MESSAGE INX 69); 04470120
        BEGIN DI:=DI+31; SI:=LOC A; C:=DI; DI:=LOC D; DS:=8DEC; 04470130
          DI:=LOC D; DS:=7FILL; DI:=C; 8(IF SC=" " THEN SI:=SI+1 04470140
          ELSE DS:=CHR); DS:=4LIT" TO "; C:=DI; DI:=LOC D; DS:=8DEC; 04470150
          DI:=LOC D; DS:=7FILL; DI:=C; SI:=LOC D; 8(IF SC=" " THEN 04470160
          SI:=SI+1 ELSE DS:=CHR); DS:=LIT"+"; 04470170
```

```

END; 04470180
SPOUT(69); 04470200
SPOUT(14); 04470210
1 KEYIN(69); 04470220
2 IF NOT OK(MESSAGE INX 69) THEN GO BOMBOUT; 04470230
3 END; 04470235
4 $ POP OMIT 04470236
5 INFO[2]:=ESUTEMP; 04470240
6 $ SET OMIT = COOL 04480000
7 LABELS[28-I] + @114; 04490000
8 LABELS[29-I] + 0; 04500000
9 FOR I:=0 STEP 1 UNTIL(SHAR[0]-1) DO DISKIO(DISKTOP-I,INFO); 04510000
10 $ POP OMIT 04510050
11 $ SET OMIT = NOT(COOL) 04520000
12 FOR I:=-7 STEP 1 UNTIL -4 DO DISKIO(DISKTOP+I,ZERO); 04520200
13 FOR I:=1 STEP 1 UNTIL 3 DO DISKIO(DISKTOP+I,ZERO); 04520300
14 %***** LOOK AT DIRECTORYTOP ***** 04520400
15 FNDERR:=FALSE; 04520450
16 FOR I:=0 STEP 1 UNTIL 29 DO 04520500
17 IF P(INFO[I],TOP,XCH,DEL) THEN ELSE 04520600
18 BEGIN FNDERR:=TRUE; INFO[I]:=0; END; 04520700
19 IF (INFO[2] DIV 100) GTR 10 OR (INFO[2] MOD 100) GTR 10 04520800
20 THEN BEGIN SPOUT(56); GO BOMBOUT; END; 04520900
21 IF INFO[8] GTR @7777 THEN 04521000
22 BEGIN FNDERR:=TRUE; INFO[8]:=1; END; 04521050
23 IF INFO[9] LSS 0 THEN 04521100
24 BEGIN FNDERR:=TRUE; INFO[9]:=1; END; 04521150
25 IF FNDERR THEN SPOUT(56); 04521175
26 %***** END OF DIRECTORYTOP EXAMINATION ***** 04521200
27 DISKIO(DISKTOP-SYSNO,INFO); 04521300
28 $ POP OMIT 04530050
29 N := ESPBOTTOM ;%START CLEAN ESPDISK 04540000
30 MRCLEAN: IF N LSS (DISKTOP-SHAR[0]) THEN 04550000
31 BEGIN ;STREAM(B:=N,X:=FILEH); 04560000
32 BEGIN SI:=LOC B;DS:=8 DEC;END; 04570000
33 P(N); 04580000
34 IF (N:=(DISKTOP-SHAR[0]-N))GEQ 63 THEN 04590000
35 IO(FILEH INX 0,@140000007700000) 04600000
36 ELSE 04610000
37 IO(FILEH INX 0,@1400000000000000&N[27:42:6]); 04620000
38 N := P + 63; 04630000
39 GO TO MRCLEAN; 04640000
40 END; 04650000
41 STREAM (B:=DISKTOP+1,X:=FILEH); 04660000
42 BEGIN SI:=LOC B;DS:=8 DEC;END; 04670000
43 IO(FILEH INX 0,@1413200000000000 & 04680000
44 $ SET OMIT = NOT(COOL) 04680099
45 (IF RD THEN 3 ELSE 2)[30:45:3]); 04680100
46 $ POP OMIT 04680101
47 $ SET OMIT = COOL 04680199
48 3[30:45:3]); 04680200
49 $ POP OMIT 04680201
50 $ SET OMIT = NOT(COOL) 04685000
51 N+(((INFO[4]-DISKTOP+3) DIV (35/2))x16) + J; 04690000
52 MN:=DISKTOP +4; 04691000
53 YR:=INFO[4]; 04691100
54 TYPE:=INFO[2]x1000000; 04691200
55 IF P1IO OR P2IO THEN BEGIN 04691300
56 STREAM(MI:=MESSAGE[53]); 04691400
57 DS:=25LIT"CHECK PRINTER FOR OUTPUT="); 04691500

```



```
SPOUT(53);                                04691600
                                           04691700
                                           04700000
FOR J+J STEP 16 UNTIL N DO
BEGIN                                       04710000
  NUMWDS := 480;                            04720000
  DISKIO(=(J-15),FILEX);                    04730000
  FOR I:=0 STEP 2 UNTIL 28 DO               04740000
  BEGIN                                       04750000
    STREAM(A:=0:D:=LABELS[I]);              04760000
    BEGIN SI:=0;IF SC#0 THEN TALLY:=1;      04770000
      SI:=SI+8;IF SC#0 THEN TALLY:=1;      04780000
      AI=TALLY;                              04790000
    END;                                       04800000
    IF P THEN REMOVE(0);                     04810000
  END;                                       04820000
  FOR I+28 STEP -2 UNTIL 0 DO               04830000
  IF LABELS[I]=@114 THEN GO QUIT ELSE      04840000
  IF LABELS[I]#@14 THEN                    04850000
    IF RTI LEQ 0 THEN GO ON ELSE           04851000
    IF P(LOOKFOR(LABELS[I],LABELS[I+1]))   04852000
    THEN BEGIN                                04853000
      STREAM(A:=LABELS[I],B:=MESSAGE[59]); 04854000
      BEGIN SI:=A; SI:=SI+1; DS:=7CHR;      04855000
        DI:=DI+1; SI:=SI+1; DS:=7CHR;      04856000
      END;                                       04856500
      LABELS[I]:=@14;                          04856600
      IF NOT(P110 OR P210) THEN              04856650
      IF NOT SPOID THEN ELSE                04856670
      IO(MESSAGE INX 59,@7400000000000000) 04856700
      ELSE                                       04856750
      IO(MESSAGE INX 59,IF P110 THEN         04856760
        @540030004000000 ELSE              04856770
        @640030004000000);                  04856780
    END;                                       04857000
  ELSE                                       04858000
  ON: BEGIN                                    04860000
    IF (LABELS[I] EQV "DECK ")=NOT 0 THEN   04869000
    BEGIN                                       04869200
      IF NOT HARD THEN BEGIN                04869400
        IF SPOID THEN                       04869500
        BEGIN                                  04869550
          SPOUT(20); KEYIN(20);             04869600
          RD:=OK(MESSAGE[20] INX 0);        04869800
          HARD:=TRUE;                        04870000
          END;                               04870100
        END;                                  04870200
      IF RD THEN                             04870400
      IF ((LABELS[I+1] AND @77000000007777) EQV 04880000
        @12000000003714)=NOT 0 THEN        04890000
      BEGIN                                       04900000
        LABELS[I+1]+(P(DUP,LOD) AND        04910000
          @77777777770000) OR @6060;      04920000
        REMOVE(2);                           04930000
      END;                                       04935000
    END;                                       04940000
    SKEDTOG:=LABELS[I].[6:24]="FILE" AND    04941000
    (LABELS[I+1] EQV "SCHEDUL")=NOT 0;     04942000
    FILEH := ((NUMWDSI:=I*15)INX FILEX)&30[8:38:10];04950000
    K:=                                       04959000
    FILEH[9]:=(+P(DUP)) AND 31;             04960000
```

```

STREAM(T+0:FILEH); 04970000
BEGIN SI+FILEH; 30(IF SB THEN BEGIN TALLY+1; 04980000
      JUMP OUT END ELSE SI+SI+8); T+TALLY; 04990000
END; 05000000
IF P OR(K GTR 20) THEN REMOVE(1); 05010000
FOR T:=K+9 STEP -1 UNTIL 10 DO 05020000
  IF (DY:=FILEH(T)) NEG 0 THEN 05030000
  IF DY GEQ TYPE OR 05040000
  (DY LEQ YR AND DY GTR MN) THEN REMOVE(1); 05050000
DY:=0; 05059000
FOR T:=10 STEP 1 UNTIL 29 DO 05060000
  BEGIN 05061000
    DY:=DY+(XY:=(FILEH(T) NEG 0)); 05062000
    IF T GEQ K+10 THEN IF XY THEN T:=31; 05063000
  END; 05064000
  IF ((T=31) OR (DY GTR K) AND NOT SKEDTOG) OR 05065000
  (DY NEG 0 AND FILEH(8)=0) THEN REMOVE(1); 05070000
***** RECONSTRUCT BAD HEADERS ***** 05070005
IF (HDRO:=FILEH(0)).L1:14]=0 OR BLOCKSIZE=0 OR 05070010
  RECSPERBLOCK=C OR SEGSPERBLOCK=0 THEN 05070020
  BEGIN 05070030
    IF NOT HARC THEN BEGIN 05070040
      IF SPOIO THEN 05070045
      BEGIN 05070047
        SPOUT(40); KEYIN(40); 05070050
        RC:=OK([MESSAGE[40]] INX 0); 05070060
        HARC:=TRUE; 05070070
      END; 05070075
    END; 05070080
  IF RC THEN 05070090
  BEGIN IF SEGSPERBLOCK=0 THEN 05070100
    BEGIN IF RECSPERBLOCK=0 THEN 05070110
      BEGIN 05070120
        IF RECSIZE=0 THEN GO PUNT; 05070130
        IF BLOCKSIZE=0 THEN GO PUNT; 05070140
        RECSPERBLOCK:=BLOCKSIZE DIV RECSIZE; 05070150
        SEGSPERBLOCK:=(BLOCKSIZE+29) DIV 30; 05070160
      END 05070170
      ELSE 05070180
      BEGIN 05070190
        IF BLOCKSIZE=0 THEN 05070200
        IF RECSIZE=0 THEN GO PUNT 05070210
        ELSE 05070220
        BEGIN BLOCKSIZE:=RECSPERBLOCK*RECSIZE; 05070230
        SEGSPERBLOCK:=(BLOCKSIZE+29) DIV 30; 05070240
        END 05070250
      ELSE 05070260
      BEGIN 05070270
        RECSIZE:=BLOCKSIZE DIV RECSPERBLOCK; 05070280
        SEGSPERBLOCK:=(BLOCKSIZE+29) DIV 30; 05070290
      END 05070300
    END 05070310
  END 05070320
  ELSE 05070330
  BEGIN 05070340
    BLOCKSIZE:=SEGSPERBLOCK * 30; 05070350
    IF RECSIZE=0 THEN 05070360
    IF RECSPERBLOCK=0 THEN GO PUNT 05070370
    ELSE 05070380
    RECSIZE:=BLOCKSIZE DIV RECSPERBLOCK 05070385

```

1
2
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

0
1
2
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

ELSE RECSPERBLOCK:=BLOCKSIZE DIV RECSIZE; 05070390

END; 05070400

FILEH[0]:=HDR0; 05070405

GO ARND; 05070410

PUNT; BLOCKSIZE:=30; 05070420

RECSIZE:=30; 05070425

RECSPERBLOCK:=1; 05070430

SEGSERBLOCK:=1; 05070435

FILEH[0]:=HDR0; 05070440

ARND; 05070450

IF NUMWDS NEQ 0 THEN ESUTEMP:=FILEX[NUMWDS-1]; 05070452

DISKIO(J:=15+(I DIV 2),FILEX[NUMWDS]); 05070454

IF NUMWDS NEQ 0 THEN FILEX[NUMWDS-1]:=ESUTEMP; 05070456

END; 05070460

STREAM(A:=[LABELS[I]],BS:=BLOCKSIZE,RS:=RECSIZE,RPB:=RECSPERBLOCK, 05070470

SPB:=SEGSERBLOCK,SV:=0,B:=[MESSAGE[45]]); 05070480

BEGIN SI:=A; SI:=SI+1; DI:=B; DS:=7CHR; DI:=DI+1; SI:=SI+1; 05070490

DS:=7CHR; DI:=DI+8; SI:=LOC BS; DS:=5DEC; SV:=DI; DI:=DI-5; 05070500

DS:=4FILL; DI:=SV; DI:=DI+8; DS:=5DEC; SV:=DI; DI:=DI-5; 05070510

DS:=4FILL; DI:=SV; DI:=DI+6; DS:=4DEC; SV:=DI; DI:=DI-4; 05070520

DS:=3FILL; DI:=SV; DI:=DI+6; DS:=2DEC; DI:=DI-2; DS:=FILL; 05070530

END; 05070540

IF NOT(P1IO OR P2IO) THEN IF NOT SPOID THEN ELSE 05070550

IO(MESSAGE INX 45,@7400000000000000) ELSE 05070560

IO(MESSAGE INX 45,IF P1IO THEN @540100004000000 05070570

ELSE @640100004000000); 05070580

END; 05070590

***** END RECONSTRUCTION CODE ***** 05070600

REMOVED: 05071000

END; 05080000

DISKIO(J,LABELS); 05090000

END; 05100000

LABELS[0]+@114; 05110000

QUIT: 05120000

IF P1IO OR P2IO THEN IO(0,@440004000100000&(IF P1IO THEN 05120010

@1 ELSE @2)[4:46:2]); 05120020

\$ POP UNIT 05120050

DISKIO(J,LABELS); 05130000

FOR I ← 0 STEP 1 UNTIL 29 DO INFO[I] ← 0; 05160000

IF SPOID THEN 05160100

SPOUT(0); 05170000

QX ← 0; 05180000

IO(LOADER,@240000540000000); 05190000

IF QX ≠ 2 THEN 05200000

STREAM(L←LOADER,A+16); BEGIN SI←L; DS←20 WDS END 05210000

ELSE 05220000

STREAM(LOADBUTTON,A+@20); BEGIN SI←LOADBUTTON;DS←15 WDS END; 05230000

P(0,STS,0,STF); 05240000

GO TO TART; 05250000

DO UNTIL FALSE; 05260000

END OF WARM LOADER. 05270000

Data Documents/Inc.

LABEL 000000000PRINTER00175100CC EX OBJECT/READ;FILE SOURCEFILE=SYMBOL/CUOL+00+0000000

OBJECT /READ

1
2
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

Data Documents/Inc.

33483

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