

MCP PROCEDURE NAMES

Data Documents/Inc.

OBJECT /KEYWORD  
=====

\$SET SINGLE LIST	C	0000	
\$SET KEYWORDSCAN	C	0000	
BEGIN %PROGRAM TO READ AND PRINT A DISK SOURCE FILE	C	0000	
			START OF SEGMENT ***** 2
LABEL ENDOFFILE;	C	0000	
% WONEE ZOWEE --- 9 BUFFERS FOR THE PRINTER AND DISK FILES	C	0000	
FILE IN SOURCEFILE DISK SERIAL "SOURCE"/"FILE" (9, 30);	C	0000	
FILE OUT PRINTER 1 (9,17)	C	0003	
POINTER PCARD, P, P1, P2, P3;	C	0007	
REAL I;	C	0007	
DEFINE I1L = STEP 1 UNTIL#;	C	0007	
DEFINE KEYWORD = "PROCEDU" #;	C	0007	
DEFINE KEYWORDFIRSTLETTER = "P"#;	C	0007	
DEFINE DUMPLINE(DUMPLINE1) =	C	0007	
BEGIN	C	0007	
\$SET OMIT = NOT KEYWORDSCAN	C	0007	
LABEL SCANIT;	C	0007	
P1 ← P + POINTER(DUMPLINE1,6);	C	0007	
I ← 80-7;	C	0007	
SCANIT:	C	0007	
SCAN P1:P1 FOR I:I UNTIL = KEYWORDFIRSTLETTER;	C	0007	
IF I≠0 THEN IF DELTA(P,P1) < 80-7 THEN IF REAL(P1,7) =	C	0007	
KEYWORD THEN	C	0007	
BEGIN	C	0007	
\$POP OMIT	C	0007	
WRITE(PRINTER, 10, DUMPLINE1[*]);	C	0007	
\$SET OMIT = NOT KEYWORDSCAN	C	0007	
P1 ← P1+1;	C	0007	
GO TO SCANIT;	C	0007	
END;	C	0007	
\$POP OMIT	C	0007	
END#;	C	0007	
ARRAY CARDIO : 29];	C	0007	
ARRAY LINE1, LINE2, LINE3 [0 : 9];	C	0008	
WHILE TRUE DO	C	0011	
BEGIN	C	0011	
READ(SOURCEFILE, 30, CARD[*])(ENDOFFILE);	C	0011	
PCARD ← POINTER(CARD,6);	C	0017	
REPLACE POINTER(LINE2,6) BY PCARD+80 FOR 10 WORDS;	C	0018	
REPLACE POINTER(LINE3,6) BY PCARD+160 FOR 10 WORDS;	C	0024	
DUMPLINE(CARD); DUMPLINE(LINE2); DUMPLINE(LINE3);	C	0029	
			START OF SEGMENT ***** 3
			3 IS 25 LONG, NEXT SEG 2
			START OF SEGMENT ***** 4
			4 IS 25 LONG, NEXT SEG 2
			START OF SEGMENT ***** 5
			5 IS 25 LONG, NEXT SEG 2
END;	C	0032	
ENDOFFILE;	C	0032	
END.	C	0033	
			2 IS 36 LONG, NEXT SEG 1

Data Documents, Inc.

1 IS 2 LONG, NEXT SEG 0  
15 IS 69 LONG, NEXT SEG 0

NUMBER OF ERRORS DETECTED = 0. COMPILATION TIME = 19 SECONDS.

PRT SIZE = 46; TOTAL SEGMENT SIZE = 182 WORDS; DISK SIZE = 14 SEGS; NO. PGM. SEGS = 15

ESTIMATED CORE STORAGE REQUIRED = 2875 WORDS.

ESTIMATED AUXILIARY MEMORY REQUIRED = 0 WORDS.

NUMBER OF CARD-IMAGES PROCESSED = 37.

Data Documents/Inc.

LABEL 00000000LINE 00100000?COMPILE OBJECT/KEYWORD WITH XALGOL LIBRARY

XALGOL /OBJECT

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

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.

MCP PROC NAMES

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57

Data Documents/Inc.

	SAVE INTEGER PROCEDURE GETSPACE(SIZE,TYPE,SAVEF);%	00014000
	SAVE REAL PROCEDURE WAITIO(IOD,MASK,U);%	00018000
	SAVE PROCEDURE DISKWAIT(CORE,SIZE,DISK);	00019100
1	PROCEDURE ERRORFIXER(TYPE); VALUE TYPE; REAL TYPE; FORWARD;	00019500
2	SAVE PROCEDURE SNOOZE(PRYR,ADDRESS,MASK); VALUE PRYR,ADDRESS,MASK;	00020000
3	PROCEDURE ENTERSYSMTR(N); VALUE N; REAL N; FORWARD;	00024700
4	SAVE PROCEDURE LOGTIMING(MIX,B); VALUE MIX,B; REAL MIX,B; FORWARD;%	00025000
5	SAVE PROCEDURE ESPBIT; COMMENT PRESENCE BIT ROUTINE FOR ESP SEGMENTS ;%	00025900
6	STREAM PROCEDURE MOVE(N)"WORD, FROM"(HERE)"TO"(THERE);%	00082000
7	PROCEDURE STOPM(B); VALUE B; BOOLEAN B; FORWARD;	00089100
8	SAVE PROCEDURE FORGETSPACE(LOC);%	00090000
9	PROCEDURE WORKSET(N); VALUE N; REAL N; FORWARD;	00134125
10	SAVE INTEGER PROCEDURE AUXILIARYSPACE(SIZE);	00135700
11	SAVE PROCEDURE FORGETAUXILIARYSPACE(SIZE, LOC);	00135800
12	SAVE PROCEDURE LOGTIMING(MIX,B); VALUE MIX,B; REAL MIX,B;	00165050
13	REQUESTED ADDRESS IS UNLOCKED, THE PROCEDURE FINDFREEADDRESS	00296380
14	SAVE REAL PROCEDURE TWO(N); VALUE N; INTEGER N;	00306000
15	PROCEDURE FORGETUSERDISK(A,L);VALUE A,L;REAL A,L;FORWARD;%	00316000
16	REAL PROCEDURE PETUSERDISK(N,T);VALUE N,T;REAL N,T;FORWARD ;	00316100
17	SAVE PROCEDURE DT; FORWARD;%	00317000
18	PROCEDURE USERDISKSPECIALCASE(Q,R,U,J);VALUE Q,J;REAL Q,R,J;	00336100
19	SAVE PROCEDURE DISKRTN(SEGNO, SIZE);	00363000
20	PROCEDURE FORGETESPDISK(SEG);VALUE SEG;REAL SEG;FORWARD;	00364000
21	SAVE INTEGER PROCEDURE DISKSPACE(NWORDS,PIMIX,AUX);%	00365000
22	PROCEDURE STATUS;%	00369000
23	PROCEDURE INTERRUPT(TYPE);VALUE TYPE;REAL TYPE; FORWARD;	00370500
24	REAL PROCEDURE FINDOUTPUT(MID,FID,TYPE,FORMS,REEL,COATE,CYCLE,KIND);%	00371000
25	REAL PROCEDURE FINDINPUT(MID,FID,REEL,COATE,CYCLE,COBOL,UL,OF,MODE,FN);	00374000
26	PROCEDURE STARTIMING(FN,U); VALUE FN,U; REAL FN,U; FORWARD;%	00377000
27	PROCEDURE FILEOPEN(X,A); VALUE X,A; INTEGER X,A; FORWARD;	00379000
28	SAVE PROCEDURE SAVEOPEN(A); VALUE A; REAL A;	00379100
29	PROCEDURE MIXPRINT(Q); VALUE Q; REAL Q; FORWARD;	00379400
30	PROCEDURE JOBMESS(MIX,Q,A,B,C,D);VALUE MIX,Q,A,B,C,D;	00379600
31	PROCEDURE SETNOTINUSE(U,RWL); VALUE U,RWL; REAL U,RWL; FORWARD;	00380000
32	PROCEDURE FILLBUFFERS(CURRENT,FINAL,COBOL,NR);	00385000
33	PROCEDURE REALFILECLOSE(A); VALUE A; REAL A; FORWARD;	00389000
34	SAVE PROCEDURE FILECLOSE(A); VALUE A; REAL A;	00389100
35	REAL PROCEDURE DISKADDRESS(MID,FID,FPB),A,H,IU); % (SHM)	00390000
36	PROCEDURE BLASTQ(U); VALUE U; REAL U; FORWARD;%	00392000
37	REAL PROCEDURE FILEHEADER(MID,FID,NROWS,SIZE,BLEN,RLEN,S);%	00393000
38	PROCEDURE PURGEIT(U); VALUE U; INTEGER U; FORWARD;%	00397000
39	DIRECTORYSEARCH, NSECOND, AND CLEANOUT ARE THE PROCEDURES	00418970
40	PROCEDURE LOCKER(SEGMENT,LOCK);	00422500
41	PROCEDURE SELECTRUN(F); VALUE F; REAL F; FORWARD;	00426700
42	PROCEDURE CONTROLCARD(A);VALUE A;REAL A; FORWARD;%	00427000
43	REAL PROCEDURE DIRECTORYSEARCH(A,B,C);VALUE A,B,C;%	00428000
44	PROCEDURE ARTN(A,N); VALUE A,N; ARRAY A[*]; INTEGER N; FORWARD;%	00431000
45	SAVE PROCEDURE DISKIO(L,C,S,D); VALUE C,S,D; REAL L; INTEGER C,S,D;%	00432000
46	PROCEDURE SPOUT(MESSAGE);	00450300
47	PROCEDURE FILEMESS(I,K,M,F,R,D,C);	00451100
48	PROCEDURE SPOUTER(MESSAGE,UNITNO,TYPE);	00451800
49	PROCEDURE FILEMESSAGE(I,K,M,F,R,D,C,TYPE);	00452600
50	PROCEDURE LBMESS(FN,SN,I1,I2,E,UNITNO,X);	00454000
51	PROCEDURE TERMINATE(MIX); VALUE MIX; REAL MIX; FORWARD;	00463100
52	SAVE PROCEDURE TERMINALMESSAGE(N); VALUE N; REAL N; FORWARD;	00463200
53	BOOLEAN PROCEDURE SYSTEMFILE(A,B);VALUE A,B; REAL A,B; FORWARD;	00463300
54	PROCEDURE ENTERSYSFILE(N); VALUE N; REAL N; FORWARD;	00464000
55	PROCEDURE COM5; #FORWARD;%	00469000
56	PROCEDURE FILLSYSTAT;FORWARD;	00469100
57	PROCEDURE SAVESTATISTICS(FILE); VALUE FILE; REAL FILE; FORWARD;	00469400

1	PROCEDURE PSUEDOLOGENTRY(STATS); VALUE STATS; REAL STATS;	00469500
2	PROCEDURE ASR; FORWARD;%	00474000
3	PROCEDURE COM11; FORWARD;%	00475000
4	PROCEDURE COM13; FORWARD;%	00477000
5	PROCEDURE COMMUNICATE0; FORWARD;	00478000
6	PROCEDURE COMMUNICATE1; FORWARD;	00478500
7	PROCEDURE LIBRARYLOAD; FORWARD;	00479000
8	PROCEDURE LIBRARYZERO; FORWARD;	00479500
9	PROCEDURE LIBRARYDUMP; FORWARD;	00480000
10	PROCEDURE DUMPCORE(B); VALUE B; REAL B; FORWARD;	00480100
11	PROCEDURE COM19; FORWARD;%	00483000
12	PROCEDURE COM23; FORWARD;%	00487000
13	PROCEDURE INTRINSICTABLEBUILDER(FH);	00489000
14	PROCEDURE MESSAGETABLEBUILDER; FORWARD;	00491000
15	SAVE PROCEDURE RESULT;	00646000
16	SAVE PROCEDURE PUNT(I); VALUE I; REAL I;	00650000
17	PROCEDURE DCWRITE(S); VALUE S; REAL S; FORWARD;	00700000
18	PROCEDURE DQREAD(S); VALUE S; REAL S; FORWARD;	00701000
19	PROCEDURE INQPT(T); VALUE T; REAL T; FORWARD;	00701500
20	REAL PROCEDURE WHATSUP(STA,MIX,FUNC,ARAE);	00702000
21	PROCEDURE SEEKACTION(STA,MIX,FUNC,IUD);	00706000
22	PROCEDURE STATIONMESSAGEWRITER(V); VALUE V; REAL V; FORWARD;	00713200
23	SAVE REAL PROCEDURE GET(TUSTA); VALUE TUSTA; REAL TUSTA; FORWARD;	00716000
24	PROCEDURE STABLE(RD,S); VALUE RD,S; REAL RD,S; FORWARD;	00717000
25	PROCEDURE NINETEENREADER(S); VALUE S; REAL S; FORWARD;	00719000
26	PROCEDURE COM36; FORWARD; % TYPE 19 IO INTERFACE.	00720000
27	PROCEDURE MARKSPOSTA(STA,INOROUT); VALUE STA,INOROUT; REAL STA,INOROUT;	00800000
28	PROCEDURE INITIALIZEDCFILES(A,DDD,LR); VALUE A,DDD,LR;	00900000
29	PROCEDURE REMOTELOGGER(TYPE,RMCARD,STA,MIX); VALUE TYPE,RMCARD,STA,MIX;	00902000
30	PROCEDURE UPDATEABORTLN(WHOIDIDIT); VALUE WHOIDIDIT; REAL WHOIDIDIT;	00904000
31	PROCEDURE RUNSEPTIC(BUFF); VALUE BUFF; REAL BUFF; FORWARD;	00920000
32	SAVE PROCEDURE DISPOSAL(I,R); VALUE I,R; REAL I,R; FORWARD;	00921000
33	SAVE PROCEDURE LINKEU; FORWARD;	01029000
34	SAVE PROCEDURE STARTIC(U); VALUE U; REAL U; FORWARD;	01165000
35	SAVE PROCEDURE COMPLEXSNOOZE(PRI,COUE); VALUE PRI; REAL PRI,COUE;	01240000
36	PROCEDURE USASITAPE(AREA,TYPE,FRUM,U,DIR); %RHR	01250100
37	SAVE PROCEDURE SNOOZE(NEWPRI,ADDRESS,MASK);	02000000
38	SAVE PROCEDURE INDEPENDENTRUNNER(ROUTINE,PARAMETER,SSZ);	02012000
39	REAL PROCEDURE KEYIN(B); VALUE B; BOOLEAN B; FORWARD;%	02021000
40	BOOLEAN PROCEDURE SMGAMES(MIX,STA,WHICHWAY);	02021100
41	BOOLEAN PROCEDURE WHYSLEEP(MASK); VALUE MASK; REAL MASK; FORWARD;%	02022000
42	SAVE PROCEDURE RUN(MIX); VALUE MIX; REAL MIX;%	02025000
43	SAVE PROCEDURE SAVEMIX(MIX); VALUE MIX; REAL MIX;%	02032000
44	SAVE PROCEDURE HALT;%	02036000
45	SAVE PROCEDURE KILL(A); VALUE A; ARRAY A[*];%	02047000
46	BOOLEAN PROCEDURE OLAY(LOC); VALUE LOC; REAL LOC; FORWARD;	02052500
47	PROCEDURE SEEKNAM(A,B,C,D,E,N); VALUE A,B; REAL A,B,C,D,E,N; FORWARD;	02052600
48	PROCEDURE UNHOOQUE(MIX);%	02053000
49	REAL PROCEDURE GETESPDISK; FORWARD;%	02111000
50	PROCEDURE CHANGEMCP(KTR); VALUE KTR; REAL KTR; FORWARD;	02111100
51	PROCEDURE CHANGEINTRINSICFILE(KTR); VALUE KTR; REAL KTR; FORWARD;	02111200
52	REAL PROCEDURE EXP; FORWARD;	02111300
53	REAL PROCEDURE ANALYSIS; FORWARD;	02111400
54	PROCEDURE SHORTCOMMUNICATE; FORWARD;	02111500
55	PROCEDURE CONTINUITYBIT; FORWARD;	02111600
56	PROCEDURE MESSAGEWRITER;%	02114000
57	PROCEDURE TWXOUT(A,B,C,D); VALUE A,B,C,D; REAL A,B,C,D; FORWARD;	02131012
	PROCEDURE RJINPUT(STA); VALUE STA; REAL STA; FORWARD;	02131013
	PROCEDURE RJEUNITCHANGE(T); VALUE T; REAL T; FORWARD;	02131014
	PROCEDURE SET(TUSTA); VALUE TUSTA; REAL TUSTA; FORWARD;	02131016

	PROCEDURE TACH(A,B,C); VALUE A,B; REAL A,B,C; FORWARD;	02131017
	PROCEDURE SPOSIN(STA, FUNC, BUFF); VALUE STA, FUNC, BUFF; REAL STA, FUNC, BUFF;	02131020
	PROCEDURE SPOUT(MESSAGE);	02132000
1	PROCEDURE SPOUTER(MESSAGE, UNITNO, TYPE);	02132300
2	PROCEDURE ENDOFDECK(R, TUSTA); VALUE R, TUSTA; REAL R, TUSTA; FORWARD;	02177100
3	PROCEDURE PBIQ(A,B); VALUE A; REAL A,B; FORWARD;	02178500
4	PROCEDURE TERMINATE(MIX); VALUE MIX; REAL MIX; %	02180000
5	REAL PROCEDURE PLACEFINDER(S, A, L);	02187100
6	PROCEDURE TERMINALMESSAGA(N); VALUE N; REAL N;	02188000
7	SAVE PROCEDURE TERMINALMESSAGE(N); VALUE N; REAL N;	02330100
8	SAVE PROCEDURE CHECKLINKS;	02330600
9	BOOLEAN PROCEDURE READFROMDISK(M, I, B);	02347150
10	PROCEDURE DRAINQ(UNIT, BUMP, ERROR);	02347200
11	REAL PROCEDURE UNITIN(TINU, WHAT); VALUE WHAT; REAL WHAT;	02348000
12	PROCEDURE IDLETIME;%	02360000
13	REAL PROCEDURE EUF(A,B,L); VALUE A,B,L; REAL A,B,L; FORWARD;	02379000
14	INTEGER PROCEDURE CALCULATEPURGE(PURGE); %	02380000
15	PROCEDURE CHANGEDATE(BUFF); VALUE BUFF; REAL BUFF; FORWARD;	02393100
16	REAL PROCEDURE TAPELABEL(M,F,R,C,P); VALUE M,F,R,C,P;	%AI02393400
17	PROCEDURE PRINTCORE;%	02394000
18	PROCEDURE DUMPCORE(BUFF);	%AI02434100
19	PROCEDURE LOGCOMMENT(Q); VALUE Q; REAL Q; FORWARD;	02434790
20	SAVE PROCEDURE TRACE(WORD, FORMAT); %	02436000
21	PROCEDURE DISKBUG; FORWARD; %	02468000
22	PROCEDURE PRINTRACE;%	02469000
23	PROCEDURE EGGREMOVE(WHATODO, PARAMETER, SPACE);	02522100
24	PROCEDURE EGGENTER(WHATODO, PARAMETER, SECONDS);	02522400
25	PROCEDURE EGGSELECT(PARAMETER); VALUE PARAMETER; REAL PARAMETER; FORWARD;	02522800
26	PROCEDURE NAMEID(A, KTR); %	02603000
27	REAL PROCEDURE TAPELABEL(MULFID, FID, REELNO, CYCLE, PURGE); %	02635000
28	REAL PROCEDURE LABELASCATCH(LBL); VALUE LBL; REAL LBL; %	02659000
29	PROCEDURE NSECOND; FORWARD; %	02692000
30	REAL PROCEDURE SECURITYCHECK(M,F,U,H);	02696000
31	PROCEDURE MAKEPRESENT(C); VALUE C; REAL C; FORWARD;	02696200
32	PROCEDURE SIGNOFF(V,F,W); VALUE V,F,W; ARRAY V[*], F[*]; REAL W; FORWARD;	02696300
33	SAVE PROCEDURE IQREQUEST(F,I,L); VALUE F,I,L; ARRAY F,L[*]; REAL I;	02696500
34	BOOLEAN PROCEDURE MTXIN(I,U,B); REAL U,B; INTEGER I; FORWARD;	02696700
35	PROCEDURE AUXTODISK(D,L,A,DA); VALUE D,L,DA; REAL D,L,DA; ARRAY A[*];	02697300
36	SAVE INTEGER PROCEDURE ACTUALOVERLAYADDRESS(TYPE, MIX, LOC);	02697750
37	REAL PROCEDURE BREAKOUT; %RETURNS HEAD OF BREAK FILE	02700200
38	BOOLEAN PROCEDURE BREAK(TK); VALUE TK; INTEGER TK;	02770600
39	BOOLEAN PROCEDURE BADLUMP(R,HZ); VALUE R,HZ; REAL R,HZ;	02793800
40	PROCEDURE COM12;	02809000
41	SAVE PROCEDURE FORK(R,S,Q,MK); VALUE R,S,Q,MK;	02819600
42	PROCEDURE RSDSED; %PHONY TERMINALMESSAGA, COM5, ...	02825200
43	PROCEDURE REELER(B); VALUE B; REAL B;	02833000
44	PROCEDURE REOPEN; %RESTART FILES	02843000
45	PROCEDURE RBUFIX;	02911000
46	PROCEDURE KERNAL; FORWARD;	02942600
47	PROCEDURE MOVETHEREST;	02942800
48	PROCEDURE KERNAL; %THIS IS THE RESTART SPACE PIRATE	02973400
49	PROCEDURE RESTART(S,T); VALUE T; ARRAY S[*]; REAL T;	03047200
50	SAVE PROCEDURE QTIMER(CANCELSTA); VALUE CANCELSTA; REAL CANCELSTA;	03500100
51	SAVE PROCEDURE INITIATEIO(ICODESC, MIX, U); %	04000000
52	SAVE PROCEDURE QUEUEUP(U); VALUE U; REAL U; %	04016000
53	SAVE PROCEDURE LINKEU;	04019500
54	SAVE PROCEDURE STARTIO(U); VALUE U; REAL U; %	04020000
55	SAVE PROCEDURE PRINTERFINISH(U); VALUE U; REAL U; %	04035000
56	SAVE PROCEDURE IQREQUEST(FINAL, IODESC, LOCATION); %	04040000
57	SAVE PROCEDURE FINISHOFFIO(U); VALUE U; REAL U; %	04067000



```

PROCEDURE PROGRAMRELEASE;% 04099000
SAVE PROCEDURE NEWIO;% 04115000
PROCEDURE DISKORAUERROR(R); VALUE R; REAL R; FORWARD; 04121410
PROCEDURE ACTUALIOERR(R); VALUE R; REAL R; FORWARD; 04121425
PROCEDURE LINKUP(TYPE,KEY); VALUE TYPE,KEY; REAL TYPE,KEY; FORWARD;% 04121450
PROCEDURE CHECKJOBORFILEMESS(MIX,FIB,U);% 04121500
PROCEDURE LOGOUTMAINT(B); VALUE B; REAL B; FORWARD;% 04121600
PROCEDURE MAINTLOGGER(B); VALUE B; REAL B; FORWARD;% 04121650
SAVE PROCEDURE IOFINISH(R,C); VALUE R,C; REAL R,C; 04122000
SAVE REAL PROCEDURE WAITIO(IOID,MASK,U);% 04240000
REAL PROCEDURE TAPEPARITYRETRY(R,U,KEY);% 04254000
PROCEDURE DISKORAUERROR(B); VALUE R; REAL R; 04256000
END PROCEDURE DISKORAUERROR; 04353000
PROCEDURE ACTUALIOERR(R); VALUE R; REAL R; 04353200
REAL PROCEDURE TAPEPARITYRETRY(R,U,KEY);% 04548000
REAL PROCEDURE PLACEFINDER(S, A, L); 04700000
PROCEDURE STABLE(RD,S);VALUE RD,S;REAL RD,S; 05000000
SAVE REAL PROCEDURE GET(TUSTA); VALUE TUSTA; REAL TUSTA; 05500000
SAVE PROCEDURE SET(TUSTA); VALUE TUSTA; REAL TUSTA; 05600000
PROCEDURE LOGOUT(A); VALUE A; REAL A; FORWARD; 05607000
PROCEDURE LOGSPACE(W,L); % THIS MAY ZIP 05700000
PROCEDURE SCRATCHSORT(A,N); 05806500
PROCEDURE SCRATCHDIRECTORYERROR(A,N); 05809600
BOOLEAN PROCEDURE SCRATCHCHECK(I,H,S); 05811170
PROCEDURE SCRATCHSPECIALCASE(CN,A,N,CORADDR,SEGADDR,I,H,SCRATCHSEG); 05811340
PROCEDURE SCRATCHDIRECTORYENTER(A,N); 05813700
PROCEDURE SCRATCHDIRECTORYDELETE(A,N); 05827300
PROCEDURE SCRATCHCLEAN(ARY,BC,LINK); 05837800
REAL PROCEDURE PETUSERDISK(N,T); VALUE N,T; REAL N,T; 05839400
PROCEDURE FORGETUSERDISK(A,N); VALUE A,N; REAL A,N; 05846600
PROCEDURE DKBUSINESS(BUFF); VALUE BUFF; REAL BUFF; 05950000
SAVE PROCEDURE DISKIO(LOCIOID,CORE,SIZE,DISK);% 06000000
PROCEDURE FORGETESPDISK(SEGMENT); VALUE SEGMENT; REAL SEGMENT; FORWARD; 06020500
REAL PROCEDURE GETESPDISK;% 06021000
PROCEDURE FORGETESPDISK(SEGMENT); VALUE SEGMENT; REAL SEGMENT;% 06036000
PROCEDURE DISKBUG;% 06046000
PROCEDURE CLEANOUT(SYS); 06057340
END; % OF PROCEDURE CLEANOUT 06059820
SAVE PROCEDURE DISKWAIT(CURE,SIZE,DISK); 06061500
PROCEDURE DISKSQUASH(BUFF); 06068000
PROCEDURE CHANGEABORT(X); VALUE X; REAL X; FORWARD;% 06179000
PROCEDURE SIGNOFF(VECTOR,FILEBLOCK,PMT); 06180000
PROCEDURE USERDISKSPECIALCASE(Q,R,UT,J); 06350000
PROCEDURE GETMOREDLAYDISK(MIX);% 06400000
REAL PROCEDURE SECURITYCHECK(MID,FID,USERID,HEADER); 06460000
PROCEDURE TACH(STA,MIX,FUNC);VALUE MIX,STA; REAL STA,MIX,FUNC; 06500000
PROCEDURE REMOTELOGGER(TYPE,RMCARD,STA,MIX); %JS06846000
COMMENT PROCEDURE REMOTE_LOGGER_UPDATES_REMOTE_LOG_ACCRD_TO %JS06846300
REAL PROCEDURE NEXTCDNUM(UPDATE); VALUE UPDATE; BOOLEAN UPDATE; 07001600
PROCEDURE STARTADECK(N); VALUE N; REAL N; FORWARD; 07002000
PROCEDURE ENTERCONTROLDECK(H); VALUE H; ARRAY H[*]; FORWARD; 07002100
PROCEDURE COM23;% 07004000
PROCEDURE STARTLOADN(KTR); VALUE KTR; REAL KTR;% 07243000
PROCEDURE TABLEOFCONTENTS(B,COMNT);% 07268000
PROCEDURE REMOVEDECK(N,TUSTA);VALUE N,TUSTA;REAL N,TUSTA; 07298000
PROCEDURE DECKREMOVER(B); VALUE B; REAL B;% 07354000
BOOLEAN PROCEDURE READFROMDISK(H,IB);% 07376000
BOOLEAN PROCEDURE PRINTORPUNCHWAIT(Q,PNCH);VALUE Q,PNCH;REAL Q,PNCH; 07405100
PROCEDURE ENDOFDECK(R,TUSTA);VALUE R,TUSTA; REAL R,TUSTA; 07406000
PROCEDURE STARTADECK(N); VALUE N; REAL N; 07422000

```

	PROCEDURE RUNTHEDECK(B); VALUE B; REAL B;%	07457000
	PROCEDURE EXTERNALEND(B); VALUE B; REAL B;	07473100
	PROCEDURE CHANGE PRIORITY(BUFF,MIX); VALUE BUFF,MIX; REAL BUFF,MIX;	07485000
1	PROCEDURE ENTERCONTROLDECK(H); VALUE H; ARRAY H[*];	07541000
2	BOOLEAN PROCEDURE MTXIN(I,U,BUFF);%	08000000
3	PROCEDURE TAPEPURGE(BUFF); VALUE BUFF; REAL BUFF;%	08024000
4	PROCEDURE GIMEDATE(B,DT); VALUE B,DT; REAL B,DT; FORWARD; %RM	08070000
5	PROCEDURE REWINDANDLOCK(WHAT); VALUE WHAT; REAL WHAT;%	08079000
6	PROCEDURE PRINTDIRECTORY(BUFF,CODE);	08095000
7	BOOLEAN PROCEDURE RHEOSTAT(TYPE,MASK,MIX,STA);	08136030
8	PROCEDURE PBIO(A,P); VALUE A; REAL A,P; FORWARD; %P	08170100
9	PROCEDURE CONTINUITYBIT;%	08171000
10	BOOLEAN PROCEDURE PRINTRPUNCHWAIT(Q,PNCH); VALUE Q,PNCH; REAL Q,PNCH;	08255000
11	% THIS PROCEDURE IS RESPONSIBLE FOR STARTING PRNPBT/DISK. IT CHECKS	08255055
12	PROCEDURE PRINTBACKUP(BUFF); VALUE BUFF; REAL BUFF; %P	08282000
13	% THIS PROCEDURE HANDLES THE PB MESSAGE, MAKING THE NECESSARY CHECKS	08282110
14	SAVE PROCEDURE INITIALIZE; FORWARD;	08303000
15	SAVE REAL PROCEDURE COREND; FORWARD;	08303200
16	PROCEDURE TIMEOUT (B); VALUE B; REAL B;%	08305000
17	PROCEDURE GIMEDATE(B,DT); VALUE B,DT; REAL B,DT;	08317000
18	PROCEDURE SETDATE(BUFF); VALUE BUFF; REAL BUFF;%	08343000
19	PROCEDURE CHANGEDATE(BUFF); VALUE BUFF; REAL BUFF;%	08376000
20	PROCEDURE SETIME(BUFF); VALUE BUFF; REAL BUFF;%	08390000
21	REAL PROCEDURE FORMESS(BUFF,H1); VALUE BUFF,H1; REAL BUFF,H1;	08418000
22	PROCEDURE SUSTATUS(A,DDD,B); VALUE A,DDD,B; REAL A,B; ARRAY DDD[*];	08438900
23	PROCEDURE OUTPUTLABEL(B); VALUE B; REAL B;%	08439000
24	PROCEDURE TIMEUSED(B, X);	08525000
25	REAL PROCEDURE ANVIL(IL,Z); VALUE IL,Z; REAL IL,Z;%	08546000
26	PROCEDURE LOGOUT(A); VALUE A; REAL A;	08568000
27	PROCEDURE SAVETHEUNIT(B); VALUE B; REAL B;%	08575000
28	BOOLEAN PROCEDURE WHYSLEEP(MASK); VALUE MASK; REAL MASK;	08599000
29	END PROCEDURE WHYSLEEP;	08623000
30	PROCEDURE CHANGE OPTION(BUFF,RS);%	08624000
31	PROCEDURE TYP OP(KTR,RO); VALUE KTR,RO; REAL KTR,PO;	08679000
32	PROCEDURE DISKLOG(MID,FID,H); VALUE MID,FID,H; REAL MID,FID; ARRAY H[*];	08697100
33	PROCEDURE PBIO(ALPHA, POINTER); VALUE ALPHA; REAL ALPHA, POINTER; %P	08700000
34	% THIS PROCEDURE HANDLES IO FOR THE CREATION OF BACK-UP FILES. FOR	08700910
35	PROCEDURE TIMERELAXER(KTR,TYPE,MIX);%	08730000
36	PROCEDURE CHANGE FACTOR(BUFF,TF); VALUE BUFF,TF; REAL BUFF; BOOLEAN TF;	08800000
37	PROCEDURE SHEETIDDLER(BUFF,TYPE,SID); VALUE BUFF,TYPE,SID;	08850000
38	PROCEDURE REMOTEMESSAGE(BUFF,TYPE,MIX); VALUE BUFF,TYPE,MIX;	09000000
39	PROCEDURE LGINOUT(BUFF,TYPE); VALUE BUFF,TYPE; REAL BUFF,TYPE;	09200000
40	PROCEDURE BLACKOUT(BUFF);	09281000
41	PROCEDURE LOGDISK(A); VALUE A; REAL A;	09300000
42	PROCEDURE WHATINTRNSIC(BUFF); VALUE BUFF; REAL BUFF; FORWARD;	09400000
43	REAL PROCEDURE NEXTAUXMEMWORD(HEADER,FILEPARAM,DISKADDRESS);	09400100
44	IF DISKADDRESS = 0 THEN % FIRST CALL ON PROCEDURE	09401300
45	EXIT; END PROCEDURE NEXTAUXMEMWORD;	09404300
46	PROCEDURE TRANSFERMCP TOAUXMEM(HDRADRS,MAXLOC); VALUE HDRADRS,MAXLOC;	09404400
47	END PROCEDURE TRANSFERMCP TOAUXMEM;	09410000
48	PROCEDURE SETMONITORFILE(STOP); VALUE STOP; REAL STOP;	09410200
49	END PROCEDURE SETMONITORFILE;	09423800
50	PROCEDURE GETMONITORROW;	09423900
51	END PROCEDURE GETMONITORROW;	09428800
52	PROCEDURE ENTERSYSMTR(N); VALUE N; REAL N;	09428900
53	EXIT; END PROCEDURE ENTERSYSMTR;	09433500
54	PROCEDURE CHANGEAUXFILES(BUFF,WA);	09433600
55	PROCEDURE INTRINSICTABLEBUILDER(FH); VALUE FH; REAL FH;	09500000
56	PROCEDURE CHANGEINTRINSICFILE(BUFF); VALUE BUFF; REAL BUFF;%	09600000
57	PROCEDURE CHANGE MCP(BUFF); VALUE BUFF; REAL BUFF;	09679100

	BOOLEAN PROCEDURE SYSTEMFILE(A,B); VALUE A,B; REAL A,B;%	09700000
	REAL STREAM PROCEDURE GNC(B);%	10001000
	BOULEAN STREAM PROCEDURE TAN(C); VALUE C;%	10007000
1	REAL PROCEDURE EXP;%	10012000
2	STREAM PROCEDURE FRMT1(Q,T,M,WB); VALUE Q,T,M;%	10047000
3	PROCEDURE DT;	10082000
4	PROCEDURE WKSETVALUES(KTRX); VALUE KTRX; REAL KTRX;	12201000
5	END PROCEDURE WKSETREQUESTS;	12321500
6	PROCEDURE WORKSET(N); VALUE N; REAL N;	12350500
7	% NOTE: WKSETNOSELECT IS A FLAG TO PROCEDURE SELECTRUN TO	12425500
8	REAL PROCEDURE PRNPBTSPCASE1(Z);	12500000
9	% THIS PROCEDURE HANDLES THE FOLLOWING FUNCTIONS FOR COM19, DEPENDING	12500110
10	PROCEDURE PRNPBTSPCASE2(Z);	12800000
11	% THIS PROCEDURE HANDLES ADDITIONAL THINGS FOR COM19, VALUES OF Z ARE:	12800110
12	PROCEDURE COM19;	13000000
13	PROCEDURE DCREAD(STA); VALUE STA; REAL STA;	13200000
14	PROCEDURE DCWRITE(STA); VALUE STA; REAL STA;	13250000
15	PROCEDURE RJEUNITCHANGE(T); VALUE T; REAL T;	13301000
16	PROCEDURE TWXOUT(UNIT,NUMCHRS,AREA,STA);	13379000
17	PROCEDURE RJINPUT(STA); VALUE STA; REAL STA;	13509000
18	BEGIN %THIS IS THE PROCEDURE THAT HANDLES ALL REMOTE JOB INPUT.	13510000
19	REAL PROCEDURE ANALYSIS;%	14000000
20	SAVE INTEGER PROCEDURE ACTUALOVERLAYADDRESS(TYPE, MIX, LOC);	14105000
21	PROCEDURE AUXILIARYMEMORYCASUALTYRECOVERY;	14120000
22	PROCEDURE AUXILIARYTABLEINITIALIZE;	14122000
23	PROCEDURE MAKEPRESENT(C); VALUE C; REAL C;%	14155000
24	PROCEDURE ZIPPER(A,B,C); VALUE A,B,C; REAL A,B,C; FORWARD;	14342100
25	PROCEDURE COM5;%	14343000
26	PROCEDURE ZIPPER(W1,W2,USERSTA); VALUE W1,W2,USERSTA;	14531000
27	REAL PROCEDURE EUF(A,B,L); VALUE A,B,L; REAL A,B,L;	14543000
28	PROCEDURE COM11; COMMENT ALGOL I/O COMMUNICATE;%	14623000
29	PROCEDURE COM36; % COMMENT TYPE 19 IO INTERFACE	14716000
30	PROCEDURE DISPLAY(X); VALUE X; REAL X;%	15019000
31	PROCEDURE COM13;%	15060000
32	BOOLEAN PROCEDURE CONQUER(C,N,L,S,G);	15168000
33	BOOLEAN PROCEDURE PRTGAMES(BUFF,MIX); VALUE BUFF,MIX; REAL BUFF,MIX;	15400000
34	PROCEDURE CHEKREMOLOG(BUFF); VALUE BUFF; REAL BUFF;	15442200
35	PROCEDURE WHATMCP(BUFF); VALUE BUFF; REAL BUFF; %FORMATS WM MESSAGE	15500000
36	PROCEDURE WHATINTRNSIC(BUFF); VALUE BUFF; REAL BUFF;	15534000
37	PROCEDURE COREPRINT(Q); VALUE Q; REAL Q;	15600000
38	COMMENT : THIS PROCEDURE COMPUTES AND TYPES THE AMOUNTS OF SAVE	15600050
39	REAL PROCEDURE AUXPRINT(Q); VALUE Q; REAL Q;	15605000
40	END PROCEDURE AUXPRINT;	15609600
41	PROCEDURE LOGCOMMENT (Q); VALUE Q; REAL Q;	15610000
42	REAL PROCEDURE KEYINSCAN(KTR,MIX); REAL KTR,MIX;	16034900
43	END PROCEDURE KEYINSCAN;	16043800
44	PROCEDURE KEYINO(B,KTRX); VALUE B,KTRX; REAL B,KTRX;	16044000
45	END PROCEDURE KEYINO;	16344000
46	PROCEDURE KEYIN1(B,KTRX); VALUE B,KTRX; REAL B,KTRX;	16345000
47	END PROCEDURE KEYIN1;	16689500
48	PROCEDURE KEYIN2(KTRX); VALUE KTRX; REAL KTRX;	16690000
49	END PROCEDURE KEYIN2;	16903100
50	REAL PROCEDURE KEYIN(B); VALUE B; REAL B;	16904000
51	% THIS PROCEDURE FUNCTIONS AS A DRIVER FOR AUXILIARY PROCEDURES	16904500
52	% THIS PROCEDURE FUNCTIONS AS A DRIVER FOR AUXILIARY PROCEDURES	16904500
53	% "KEYINO","KEYIN1" AND "KEYIN2". PROCEDURES "KEYINO" AND "KEYIN1"	16905000
54	% ARE CALLED DIRECTLY, AND PROCEDURE "KEYIN2" IS CALLED AS AN	16905500
55	END PROCEDURE KEYIN;	16971500
56	PROCEDURE LBMESS(FN,SN,I1,I2,E,UNITNO,X);	17000000
57	PROCEDURE STOPM(NCS); VALUE NCS; REAL NCS;	17900500

Data Documents, Inc.

END PROCEDURE STOPM;	17928000
PROCEDURE TASKTASK; FORWARD;	17928500
PROCEDURE FILEHOLD(A,B,TQG,LOC,HULD);	18000000
REAL PROCEDURE DIRECTORYSEARCH(A,B,OPTN);*	18155000
PROCEDURE PICKTHELOCK; FORWARD;	18468100
PROCEDURE EVENTANDINTERRUPT; FORWARD;	18468200
PROCEDURE COMMUNICATE;	18500000
PROCEDURE COMMUNICATE0;	18700000
PROCEDURE SHORTCOMMUNICATE;	19500000
PROCEDURE TASKTASK;	19687000
PROCEDURE PICKTHELOCK;	19900000
BEGIN COMMENT THIS PROCEDURE HANDLES LOCKING/UNLOCKING OF	19900010
PROCEDURE EVENTANDINTERRUPT;	19900900
* LOWER WORD: PROCEDURE DESCRIPTOR FOR INTERRUPT	19902509
PROCEDURE SELECTRUN1;	20011200
*** THOSE DECLARED IN PROCEDURE SELECTRUN,	20016400
THIS PROCEDURE WILL BE EXECUTED. THIS PROCEDURE CAN ALSO	20018200
THIS PROCEDURE WILL BE EXECUTED. THIS PROCEDURE CAN ALSO	20018200
RETURNING TO THAT PROCEDURE.	20018500
* PROCEDURE ONLY	20023000
END PROCEDURE SELECTRUN1;	20080200
PROCEDURE SELECTRUN2;	20080500
*** THOSE DECLARED IN PROCEDURE SELECTRUN,	20085600
END PROCEDURE SELECTRUN2;	20140300
PROCEDURE SELECTRUN(F); VALUE F; REAL F;	20140800
*** VARIABLES DECLARED IN THIS PROCEDURE,	20146000
PROCEDURES "SELECTRUN1" AND/OR "SELECTRUN2" WILL BE EXECUTED,	20147200
PROCEDURE "SELECTRUN1" AND "SELECTRUN2" MAY, IN TURN, SPECIFY	20147300
*** SEE ALSO "SEGMENT ZERO" SECTION IN PROCEDURE "SELECTRUN2" FOR	20176200
PROCEDURE PRINTTHECOVER(CARD,UNITNO,PS);	20289010
PROCEDURE FETCH(UNITNO,CARDLOC,SOURCE);	20292000
REAL PROCEDURE SCN(UNITNO,CARDLOC,SOURCE,ACCUM,KOUNT,LASTSCAN,	20314000
PROCEDURE SEEKNAM(A,B,C,D,E,N); VALUE A,B; REAL A,B,C,D,E,N;	20382010
REAL PROCEDURE PPC;	20383000
PROCEDURE SECURITYMAINT( TYPE,SMID,SFID,CMM,SFH,CARD);	20511100
*** ALSO SEE PROCEDURE "SELECTRUN1" (SEQ.NO.20055600) FOR	20542800
REAL PROCEDURE LIBCC;	20566000
END; * LIBCC PROCEDURE	20580350
REAL PROCEDURE CCSET; FORWARD;	20580400
PROCEDURE CCFINISH;	20580800
REAL PROCEDURE CCOMPFILE;	20583800
REAL PROCEDURE INITIALIZEIT;	20586700
REAL PROCEDURE CCUNIT;	20589700
REAL PROCEDURE CCSECMAINT;	20590850
REAL PROCEDURE CCLABEL;	20594850
BOOLEAN PROCEDURE CCFIND;	20596750
PROCEDURE CONTROLCARD(CARD); VALUE CARD; REAL CARD;	20597550
REAL PROCEDURE CCSET;	20700000
REAL PROCEDURE WHATSUP(STA,MIX,FUNC,ARAE);	21000000
PROCEDURE SEEKACTION(STA,MIX,FUNC,IUD);	21500000
PROCEDURE UPDATEABORTLN(WHODIDIT); VALUE WHODIDIT; REAL WHODIDIT;	21700000
\$ RESET OMIT                                  * END OF DATACOM PROCEDURES	21752001
PROCEDURE FINDFREEADDRESS(N);VALUE N;REAL N;FORWARD;	22000500
PROCEDURE NSECOND;*	22001000
PROCEDURE STATUS;*	22055000
BOOLEAN PROCEDURE OLAY(LOC);          * MADE SAVE IN INITIALIZE	22228000
PROCEDURE CHANGEABORT(X); VALUE X; REAL X;*	22900000
PROCEDURE STATIONMESSAGEWRITER(PARAMETER);	23000000
THIS PROCEDURE HANDLES ALL B487 OUTPUT, BUFFER SIZE NEED NOT BE	23003000
KNOWN AS THE PROCEDURE LEARNS FROM THE RESULT DESCRIPTOR HOW MUCH WAS	23004000

57

	SAVE PROCEDURE FORGETSPACE(LOC);%	24000000
	SAVE INTEGER PROCEDURE ACTSPACE(SIZE,SAVEF);	24032050
	SAVE INTEGER PROCEDURE GETSPACE(SIZE,TYPE,SAVEF);	24042200
1	SAVE INTEGER PROCEDURE DISKSPACE(WURDS,MIX,AUX);	24101000
2	SAVE PROCEDURE DISKRTN(SEGNO, SIZE);	24200000
3	SAVE PROCEDURE DCQPT(IR);VALUE IR; BOOLEAN IR;	24500000
4	PROCEDURE FINDFREEADDRESS(N); VALUE N; REAL N;	24600000
5	IF (FINDFREECTR:=FINDFREECTR+1) GEQ 32 THEN% THIS PROCEDURE IS	24610000
6	% CALLED FREQUENTLY SO MAKE IT A SAVE PROCEDURE	24611000
7	PROCEDURE MARKSPOSTA(STA,INQROUT);VALUE STA,INQROUT;REAL STA,INQROUT;	25000000
8	PROCEDURE INQURT(PARAMETER);VALUE PARAMETER; REAL PARAMETER;	25050000
9	SAVE PROCEDURE DISPOSAL(I,R);	26000000
10	PROCEDURE RUNSEPTIC(ELFF);	26000000
11	PROCEDURE NINETEENREADER(STACK); VALUE STACK; REAL STACK;	27001000
12	BOOLEAN PROCEDURE B6500FORMATTER(CT,I,X,XX,HCL,H,OPTION);	27990100
13	PROCEDURE LIBRARYLOADSPECIALCASE(Z); VALUE Z; REAL Z;	27997600
14	PROCEDURE DOES INITIAL SET-UP AND OTHER SPECIAL FUNCTIONS	27997645
15	PROCEDURE LIBRARYLOAD;	28000000
16	END; % OF LIBRARY LOAD PROCEDURE	28399000
17	PROCEDURE LIBRARYDUMP;	28400000
18	PROCEDURE LIBRARYZERO;	28800000
19	SAVE PROCEDURE FILLORKILL(A, START, SIZE, TYPE);	29000000
20	SAVE INTEGER PROCEDURE AUXILIARYSPACE(SIZE);	29100000
21	SAVE PROCEDURE FORGETAUXILIARYSPACE(SIZE, LOC);	29300000
22	PROCEDURE AUXILIARYTABLEINITIALIZE;	29400000
23	PROCEDURE AUXILIARYMEMORYCASUALTYRECOVERY;	29500000
24	PROCEDURE ERRORMESSER(TYPE); VALUE TYPE; REAL TYPE;	30903000
25	PROCEDURE ERRORFIXER(TYPE); VALUE TYPE; INTEGER TYPE;	31000000
26	PROCEDURE JOBMESS(MIX,Q,A,B,C,D); VALUE MIX,Q,A,B,C,D;%	32000000
27	COMMENT : THIS PROCEDURE CAN BE USED TO BUILD AND SPOUT A MESSAGE	32000110
28	PROCEDURE MIXPRINT(Q); VALUE Q; REAL Q;	32100000
29	COMMENT THIS PROCEDURE INVOKES JOBMESS TO TYPE THE JOB SPECIFIERS	32100010
30	PROCEDURE TO BE CALLED--THE ADDRESS MUST BE IN "WHATODO",[CF], AFTER	35011000
31	"SECONDS" SECONDS THE PROCEDURE WILL BE CALLED PASSING "PARAMETER" AS	35012000
32	PROCEDURE EGGENTER (WHATODO,PARAMETER,SECONDS);	35200000
33	PROCEDURE EGGSELECT(PARAMETER);	35301000
34	PROCEDURE EGGREMOVÉ(WHATODO,PARAMETER,SPACE);	35401000
35	PROCEDURE DOLITTLE(OK,T,A,B,Z); VALUE T,A,B; REAL OK,T,A,B,Z;	36001000
36	REAL PROCEDURE FINDOUTPUT(MID,FID,REEL,COATE,CYCLE,TYPE,FORMS,KIND);	37000000
37	REAL PROCEDURE FINDINPUT(MID,FID,REEL,COATE,CYCLE,COBOL,UL,OF,MODE,FN);	37177000
38	PROCEDURE STARTIMING(FN,U); VALUE FN,U; REAL FN,U;%	37271000
39	REAL PROCEDURE DISKADDRESS(MID,FID,FPH,A,H,IO); % (SHM)	37286000
40	PROCEDURE SETNOTINUSE(U,RWL); VALUE U,RWL; REAL U,RWL;	37302000
41	PROCEDURE BLASTQ(U);	37320000
42	PROCEDURE BUILDLABEL(LABLE,MID,FID,HEEL,COATE,CYCLE,PFACT,PTN,BLKODE,%	37337000
43	PROCEDURE FILEMESS(I,K,M,F,R,D,C);	37357000
44	PROCEDURE FILEMESSAGE(I,K,M,F,R,D,C,TYPE);	37357300
45	PROCEDURE FILLBUFFERS(CURRENT,FINAL,COBOL,NR);	37385000
46	REAL PROCEDURE FILEHEADER(MID,FID,NROWS,SIZE,BLEN,RLEN,S);%	37418000
47	PROCEDURE PURGEIT(U); VALUE U; INTEGER U;%	37449000
48	PROCEDURE KRUNCHER(H); ARRAY H[*];	37500000
49	PROCEDURE DISKFILEOPEN(ALPHA); VALUE ALPHA; INTEGER ALPHA;	38000000
50	PROCEDURE OTHERFILEOPENIN(ALPHA); VALUE ALPHA; INTEGER ALPHA;	38102000
51	PROCEDURE OTHERFILEOPENOUT(ALPHA); VALUE ALPHA; INTEGER ALPHA;	38200000
52	PROCEDURE DISKCLOSE(ALPHA); VALUE ALPHA; INTEGER ALPHA;%	38355000
53	PROCEDURE BACKCLOSE(ALPHA); VALUE ALPHA; INTEGER ALPHA;%	38540000
54	PROCEDURE OTHERCLOSE(ALPHA); VALUE ALPHA; INTEGER ALPHA;%	38648000
55	PROCEDURE FILEOPEN(XTRA,ALPHA);	39000000
56	PROCEDURE SUBSTATUS(A,DDD,B); VALUE A,DDD,B; REAL A,B; ARRAY DDD[*];	39900000
57	PROCEDURE DIRECTORYBUILDER(A,DDD);	40000000

PROCEDURE REALFILECLOSE(ALPHA); VALUE ALPHA; INTEGER ALPHA;%	41000000
PROCEDURE LINKUP(TYPE,KEY); VALUE TYPE,KEY; REAL TYPE,KEY;	41310100
PROCEDURE CHECKJOBORFILEMESS(MIX,FIB,U);	41312000
PROCEDURE LOGOUTMAINT(B); VALUE B; REAL B;	41316000
PROCEDURE MAINTLOGGER(B); VALUE B; REAL B;	41327000
PROCEDURE PSUEDOLOGENTRY(STATS);	41400000
PROCEDURE FILLSYSTAT;	41401800
PROCEDURE SAVESTATISTICS(FILE);	41405400
PROCEDURE MESSAGESTABLEBUILDER;	41430000
[24:6 ] = " KEYIN PROCEDURE TO BE CALLED "	41450900
0 = PROCEDURE KEYIN0 ( DIRECT CALL )	41451000
1 = PROCEDURE KEYIN1 ( DIRECT CALL )	41451100
2 = PROCEDURE KEYIN2 ( INDEPENDENT RUNNER )	41451200
[36:12] = LABEL NUMBER ( SWITCH LOCATION IN PROCEDURE)	41451900
"1**77**", ***% END OF FIRST KEYIN PROCEDURE CALLS	41459900
"1**77**", ***% END OF SECOND KEYIN PROCEDURES	41469900
PROCEDURE ENTERSYSFILE(N); VALUE N; REAL N;	41600000
PROCEDURE ARTN(A,N); VALUE A,N; ARRAY A[*]; INTEGER N;%	42474000
PROCEDURE ASR; BEGIN INTEGER I,BCNTH; ARRAY AIT[*]; REAL TEMP;%	42482000
SAVE REAL PROCEDURE COREND; FORWARD;	42509000
PROCEDURE INTERRUPT(TYPE); VALUE TYPE; REAL TYPE;	42510000
PROCEDURE CLEANUPABORT(A,DDD); VALUE A,DDD; ARRAY DDD[*]; REAL A;	42600000
PROCEDURE INITIALIZEDCFILES (A,DDD,LR); VALUE A,DDD,LR;	42948000
PROCEDURE INTFINISH(AA);	44008998
SAVE PROCEDURE MOREINITIALIZE;FORWARD;	44009000
SAVE PROCEDURE INITIALIZE;	44010000
SAVE PROCEDURE MOREINITIALIZE;	44206450
SAVE REAL PROCEDURE COREND;%	44441000
PROCEDURE INTFINISH(AA);VALUE AA;REAL AA;	45000000

LABEL 00000000PRINTER00100000CC EX OBJECT/KEYWORD;FILE SOURCEFILE=SYMBOL/MCP;END←

OBJECT /KEYWORD

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57

Data Documents/Inc.