

```

10 ' Basic loader Program for simulated random disk files
20 MEMORY 37000
30 FOR i=37001 TO 37853
40 READ a$:v=VAL("&"+a$)
50 cs=cs+v:POKE i,v
60 NEXT i
70 IF cs<>96918 THEN PRINT"DATA ERROR - Check listing";CHR$(7):STOP
80 CALL 37001:CLS:PRINT"New commands loaded successfully"
90 PRINT"You can now NEW this loader Program"
100 END
110 DATA 01,93,90,21,EA,A3,CD,D1,BC,C9,9E,90,C3,BD,90,C3,68,91,C3,D5,92,44
120 DATA 52,45,41,C4,44,57,52,49,54,C5,43,52,45,41,54,C5,00,0E,00,CD,0F,B9
130 DATA CD,00,B9,1E,1F,C3,94,CA,CD,AE,93,FE,03,20,66,DD,66,03,DD,6E,02,22
140 DATA E3,93,7C,CB,7F,20,58,CD,C1,92,B7,28,52,32,E5,93,ED,53,E6,93,CD,CD
150 DATA 92,62,6B,47,11,EA,93,CD,77,BC,30,31,2A,E3,93,2B,7C,B5,22,E3,93,28
160 DATA 07,CD,80,BC,30,21,18,EE,DD,2A,E6,93,DD,E5,CD,80,BC,DD,E1,30,12,DD
170 DATA 77,00,DD,23,3A,E5,93,3D,32,E5,93,20,E9,CD,7A,BC,C9,21,4A,91,CD,5F
180 DATA 91,CD,7A,BC,C3,B0,90,21,2F,91,18,F2,2A,49,6C,6C,65,67,61,6C,20,70
190 DATA 61,72,61,6D,65,74,65,72,20,76,61,6C,75,65,0D,0A,00,2A,52,65,61,64
200 DATA 20,65,72,72,6F,72,20,28,45,4F,46,3F,29,0D,0A,00,7E,B7,C8,CD,5A,BB
210 DATA 23,18,F7,CD,AE,93,FE,03,C2,28,92,DD,66,03,DD,6E,02,22,E3,93,7C,CB
220 DATA 7F,C2,28,92,CD,C1,92,B7,CA,28,92,32,E5,93,ED,53,E6,93,CD,CD,92,32
230 DATA DE,93,62,6B,22,DF,93,47,11,EA,93,CD,77,BC,D2,55,92,21,B0,92,06,08
240 DATA 11,EA,9B,CD,8C,BC,D2,55,92,AF,32,E1,93,CD,80,BC,30,42,32,E2,93,3A
250 DATA E1,93,B7,20,10,2A,E3,93,2B,22,E3,93,7C,B5,20,05,3E,FF,32,E1,93,3A
260 DATA E1,93,B7,20,0A,3A,E2,93,CD,95,BC,30,57,18,D2,2A,E6,93,7E,32,E2,93
270 DATA 23,22,E6,93,3A,E5,93,3D,32,E5,93,20,E2,32,E1,93,18,DD,28,3A,3A,E5
280 DATA 93,B7,20,57,CD,7A,BC,CD,8F,BC,DD,21,AE,92,21,BB,92,CD,D4,BC,3E,01
290 DATA CD,1B,00,DD,21,AC,92,21,B8,92,CD,D4,BC,3E,02,CD,1B,00,C9,21,2F,91
300 DATA CD,5F,91,CD,7D,BC,CD,8F,BC,C3,B0,90,21,5F,92,CD,5F,91,CD,7D,BC,CD
310 DATA 8F,BC,DD,21,AC,92,21,BB,92,CD,D4,BC,3E,01,CD,1B,00,C3,B0,90,21,6F

```

320 DATA 92,18,D1,21,87,92,18,DB,2A,57,72,69,74,65,20,65,72,72,6F,72,0D,0A
330 DATA 07,00,2A,46,61,69,6C,65,64,20,74,6F,20,6F,70,65,6E,20,66,69,6C,65
340 DATA 73,0A,0D,00,2A,49,6C,6C,65,67,61,6C,20,61,74,74,65,6D,70,74,20,74
350 DATA 6F,20,77,72,69,74,65,20,70,61,73,74,20,45,4F,46,0D,0A,00,BE,92,DE
360 DATA 93,54,45,4D,50,2E,52,41,4E,52,45,CE,45,52,C1,08,B0,92,DD,66,01,DD
370 DATA 6E,00,7E,23,5E,23,56,C9,DD,66,05,DD,6E,04,18,F2,CD,AE,93,FE,02,C2
380 DATA 28,92,DD,66,01,DD,6E,00,22,E8,93,7C,FE,61,30,2A,DD,23,DD,23,CD,C1
390 DATA 92,47,62,6B,11,EA,93,CD,8C,BC,30,21,2A,E8,93,7C,B5,28,0D,2B,22,E8
400 DATA 93,3E,2A,CD,95,BC,30,14,18,EC,CD,8F,BC,C9,21,28,93,CD,5F,91,C3,B0
410 DATA 90,21,4E,93,18,F5,21,64,93,18,F0,2A,4D,61,78,20,73,69,7A,65,20,6F
420 DATA 66,20,32,34,35,37,36,20,66,6F,72,20,72,61,6E,64,6F,6D,20,66,69,6C
430 DATA 65,73,0D,0A,00,2A,43,61,6E,6E,6F,74,20,63,72,65,61,74,65,20,66,69
440 DATA 6C,65,0D,0A,00,2A,57,72,69,74,65,20,65,72,72,6F,72,20,64,75,72,69
450 DATA 6E,67,20,63,72,65,61,74,69,6F,6E,2C,0D,0A,20,46,69,6C,65,20,6E,6F
460 DATA 74,20,70,72,6F,70,65,72,6C,79,20,63,72,65,61,74,65,64,20,2D,20,44
470 DATA 4F,20,4E,4F,54,20,55,53,45,21,0D,0A,00,32,E2,93,21,C5,93,CD,D4,BC
480 DATA 3A,E2,93,D8,E1,21,C9,93,CD,5F,91,C3,B0,90,44,49,53,C3,2A,43,61,6E
490 DATA 27,74,20,66,69,6E,64,20,41,4D,53,44,4F,53,0A,0D,00

```

10 '
20 ' Demo Program for simulated random disk files
30 '
40 DEF FNP$(a$)=STRING$(40-LEN(a$),32): 'Pad string function
50 DEF FNr(rec)=(rec-1)*40+1: 'find Position of record in file function
60 ON ERROR GOTO 450
70 INK 0,0:INK 1,15:BORDER 0
80 MODE 1:PEN 0:PAPER 2:PRINT"      Random Disk Files Demonstration      ":PEN 1:PA
PER 0
90 LOCATE 1,5:PRINT"Creating file called DEMOFILE.DAT with 20 records of 40 cha
racters each"
100 filename$="DEMOFILE.DAT":size=20*40
110 !CREATE,@filename$,size
120 WINDOW #1,1,40,2,25:CLS #1
130 LOCATE 1,5:PRINT"Writing a few records into file"
140 record=1:a$="Popular Computing Weekly record 1":buf$=a$+FNP$(a$)
150 !DWRITE,@filename$,FNr(record),@buf$
160 record=20:a$="This is the last record of the file":buf$=a$+FNP$(a$)
170 !DWRITE,@filename$,FNr(record),@buf$
180 record=10:a$="Record #10 of the file":buf$=a$+FNP$(a$)
190 !DWRITE,@filename$,FNr(record),@buf$
200 CLS #1
210 LOCATE 1,5:PRINT"Press (R) to read a record,              (W) to write
a record              (Q) to finish demonstration"
220 i$=INKEY$:IF i$="" THEN 220
230 IF UPPER$(i$)="Q" THEN 420
240 IF UPPER$(i$)="R" THEN 270
250 IF UPPER$(i$)="W" THEN 340
260 PRINT CHR$(7):GOTO 220
270 CLS #1:LOCATE 1,5:INPUT "read which record (1-20)":record
280 IF record<1 OR record>20 THEN 270
290 buf$=STRING$(40,32)
300 !DREAD,@filename$,FNr(record),@buf$
310 LOCATE 1,10:PRINT"Contents of this record are:"
320 PRINT:PRINT buf$
330 LOCATE 1,20:PRINT"Press any key for menu":WHILE INKEY$="" :WEND:GOTO 200
340 CLS #1:LOCATE 1,5:INPUT "write which record (1-20)":record
350 IF record<1 OR record>20 THEN 340
360 PRINT:PRINT"Enter record contents:"
370 LINE INPUT a$
380 IF LEN(a$)>40 THEN PRINT"Record too large - truncated!":a$=LEFT$(a$,40)
390 buf$=a$+FNP$(a$):PRINT:PRINT"Writing record":record
400 !DWRITE,@filename$,FNr(record),@buf$
410 GOTO 330
420 CLS #1:LOCATE 1,5:PRINT"deleting DEMOFILE.DAT from disk..."
430 !ERA,@filename$
440 CLS:END
450 'error handler
460 PRINT:PRINT:PRINT
470 IF ERR=28 THEN PRINT"You must use the loader Program before this demo to lo
ad in the new commands!":END
480 ERROR ERR:END

```

```

10 ;
20 ; Simulated random access disk files
30 ;
40     ORG 37001
50     ENT $
60     LD BC,CTAB
70     LD HL,RXBUF
80     CALL #BCD1          ;Log on new commands
90     RET
100 CTAB: DEFW NTAB
110     JP DREAD
120     JP DWGIT
130     JP CREAT
140 NTAB: DEFM "DREA"
150     DEFB "D"+128
160     DEFM "DWGIT"
170     DEFB "E"+128
180     DEFM "CREAT"
190     DEFB "E"+128
200     DEFB 0
210 BASIC: LD C,0          ;Basic ROM number
220     CALL #B90F        ;Select this ROM
230     CALL #B900        ;Enable upper ROM
240     LD E,31          ;Error code
250     JP #CA94          ;Jump to basic
260 DREAD: CALL CK4DOS    ;Check dos exists
270     CP 3              ;Check number of Parameters
280     JR NZ,DRERR1     ;Jump if wrong number
290     LD H,(IX+3)      ;Get read from value
300     LD L,(IX+2)
310     LD (RDFRM),HL
320     LD A,H
330     BIT 7,A          ;Check not negative
340     JR NZ,DRERR1
350     CALL GETBS       ;Get buffer string details
360     OR A
370     JR Z,DRERR1
380     LD (RDLEN),A
390     LD (RDBUF),DE
400     CALL GETFN       ;Get filename details
410     LD H,D
420     LD L,E
430     LD B,A
440     LD DE,BUF2K
450     CALL #BC77       ;Open disk file for input
460     JR NC,DRERR2     ;Jump if error occurred
470 SKIPC: LD HL,(RDFRM) ;Read & skip over unwanted
480     DEC HL           ;characters in file
490     LD A,H
500     OR L
510     LD (RDFRM),HL
520     JR Z,BEGRD
530     CALL #BC80       ;Read a character from file
540     JR NC,DRERR2
550     JR SKIPC
560 BEGRD: LD IX,(RDBUF) ;Read characters into string
570 RDLF:  PUSH IX      ;for length of the string
580     CALL #BC80
590     POP IX
600     JR NC,DRERR2
610     LD (IX+0),A
620     INC IX
630     LD A,(RDLEN)

```

```

640     DEC A
650     LD (RDLEN),A
660     JR NZ,RDLP
670     CALL #BC7A       ;Close the input file
680     RET
690 DRERR2: LD HL,HWERR ;Address of error message
700 MSOUT:  CALL MESS   ;Print message
710     CALL #BC7A       ;Close the input file
720     JP BASIC        ;Return to basic with error
730 DRERR1: LD HL,PRERR
740     JR MSOUT
750 PRERR:  DEFM "*Illegal Parameter value"
760     DEFB 13,10,0
770 HWERR:  DEFM "*Read error (EOF?)"
780     DEFB 13,10,0
790 MESS:   LD A,(HL)   ;Print string subroutine
800     OR A
810     RET Z
820     CALL #BB5A
830     INC HL
840     JR MESS
850 DWGIT:  CALL CK4DOS  ;Check that DOS is Present
860     CP 3            ;Check number of Parameters
870     JP NZ,DWERR1   ;Jump if wrong number
880     LD H,(IX+3)    ;Get write from value
890     LD L,(IX+2)
900     LD (RDFRM),HL
910     LD A,H
920     BIT 7,A        ;Check not negative
930     JP NZ,DWERR1
940     CALL GETBS     ;Get buffer string details
950     OR A
960     JP Z,DWERR1
970     LD (RDLEN),A
980     LD (RDBUF),DE
990     CALL GETFN     ;Get filename details
1000    LD (FNMLN),A
1010    LD H,D
1020    LD L,E
1030    LD (FNADR),HL
1040    LD B,A
1050    LD DE,BUF2K
1060    CALL #BC77     ;Open file for input
1070    JP NC,DWERR3   ;Jump if error occurred
1080    LD HL,FNTMP
1090    LD B,8
1100    LD DE,BUF2K+2048
1110    CALL #BC8C     ;Open temporary output file
1120    JP NC,DWERR3   ;Jump if error occurred
1130    XOR A
1140    LD (RPFLG),A
1150 WRTLP: CALL #BC80   ;Copy input to output until
1160    JR NC,TSEOF    ;Write from Position is reached
1170    LD (CHR),A
1180    LD A,(RPFLG)
1190    OR A
1200    JR NZ,NODEC
1210    LD HL,(RDFRM)
1220    DEC HL
1230    LD (RDFRM),HL
1240    LD A,H
1250    OR L
1260    JR NZ,NODEC

```

```

1270 LD A,255
1280 LD <RPFLG>,A
1290 NODEC: LD A,<RPFLG>
1300 OR A
1310 JR NZ,REWRT
1320 OUTCH: LD A,<CHR>
1330 CALL #BC95 ;Write character to output file
1340 JR NC,DWERR2 ;Jump if error occurred
1350 JR WRTLP
1360 REWRT: LD HL,<RDBUF> ;Insert string into new file
1370 LD A,<HL>
1380 LD <CHR>,A
1390 INC HL
1400 LD <RDBUF>,HL
1410 LD A,<RDLEN>
1420 DEC A
1430 LD <RDLEN>,A
1440 JR NZ,OUTCH
1450 LD <RPFLG>,A
1460 JR OUTCH
1470 TSEOF: JR Z,DWERR2 ;Is this the end-of-file?
1480 LD A,<RDLEN>
1490 OR A
1500 JR NZ,DWERR4 ;Tried to write Past end-of-file
1510 CALL #BC7A ;Close input file
1520 CALL #BC8F ;Close output file
1530 LD IX,PARM1
1540 LD HL,ERA
1550 CALL #BCD4
1560 LD A,1
1570 CALL #001B ;Erase old input file
1580 LD IX,PARM2
1590 LD HL,REN
1600 CALL #BCD4
1610 LD A,2
1620 CALL #001B ;Rename temporary file
1630 RET
1640 DWERR1: LD HL,PRERR
1650 EROUT: CALL MESS ;Print error message
1660 CALL #BC7D ;Abandon input file
1670 CALL #BC8F ;Close output file
1680 JP BASIC ;Jump to basic with error
1690 DWERR2: LD HL,HWERR2
1700 OUTDL: CALL MESS
1710 CALL #BC7D
1720 CALL #BC8F
1730 LD IX,PARM2
1740 LD HL,ERA
1750 CALL #BCD4
1760 LD A,1
1770 CALL #001B
1780 JP BASIC
1790 DWERR3: LD HL,OPNER
1800 JR EROUT
1810 DWERR4: LD HL,PSEOF
1820 JR OUTDL
1830 HWERR2: DEFM "*Write error"
1840 DEFB 13,10,7,0
1850 OPNER: DEFM "*Failed to open files"
1860 DEFB 10,13,0
1870 PSEOF: DEFM "*Illegal attempt to write past EOF"
1880 DEFB 13,10,0
1890 PARM2: DEFW TFDES
1900 PARM1: DEFW FNMLN
1910 FNTMP: DEFM "TEMP.RAN"
1920 REN: DEFB "R","E","N"+128
1930 ERA: DEFB "E","R","A"+128
1940 TFDES: DEFB 8
1950 DEFW FNTMP
1960 GETBS: LD H,<IX+1>
1970 LD L,<IX+0>
1980 LDSDC: LD A,<HL>
1990 INC HL
2000 LD E,<HL>

```

```

2010 INC HL
2020 LD D,<HL>
2030 RET
2040 GETFN: LD H,<IX+5>
2050 LD L,<IX+4>
2060 JR LDSDC
2070 CREAT: CALL CK4DOS ;Check that DOS is Present
2080 CP 2 ;Check number of Parameters
2090 JP NZ,DWERR1 ;Jump if wrong number
2100 LD H,<IX+1> ;Get size of file
2110 LD L,<IX+0>
2120 LD <SIZE>,HL
2130 LD A,H
2140 CP 97 ;Check for maximum size
2150 JR NC,BIGER
2160 INC IX
2170 INC IX
2180 CALL GETBS ;Get filename details
2190 LD B,A
2200 LD H,D
2210 LD L,E
2220 LD DE,BUF2K
2230 CALL #BC8C ;Open file for output
2240 JR NC,CRERR1 ;Jump if error occurred
2250 CRLP: LD HL,<SIZE> ;Write a file full of ***
2260 LD A,H
2270 OR L
2280 JR Z,FINIS
2290 DEC HL
2300 LD <SIZE>,HL
2310 LD A,42
2320 CALL #BC95 ;Write character to output file
2330 JR NC,CRERR2
2340 JR CRLP
2350 FINIS: CALL #BC8F ;Close the output file
2360 RET
2370 BIGER: LD HL,SIZMG
2380 CEROT: CALL MESS
2390 JP BASIC
2400 CRERR1: LD HL,COPER
2410 JR CEROT
2420 CRERR2: LD HL,WRFL
2430 JR CEROT
2440 SIZMG: DEFM "*Max size of 24576 for random files"
2450 DEFB 13,10,0
2460 COPER: DEFM "*Cannot create file"
2470 DEFB 13,10,0
2480 WRFL: DEFM "*Write error during creation,"
2490 DEFB 13,10
2500 DEFM " File not Properly created - DO NOT USE!"
2510 DEFB 13,10,0
2520 CK4DOS: LD <CHR>,A ;Routine checks to see if
2530 LD HL,DISC ;the IDISC command can be
2540 CALL #BCD4 ;found - check for DOS
2550 LD A,<CHR>
2560 RET C
2570 POP HL
2580 LD HL,NODOS
2590 CALL MESS
2600 JP BASIC
2610 DISC: DEFB "D","I","S","C"+128
2620 NODOS: DEFM "*Can't find AMSDOS"
2630 DEFB 10,13,0
2640 FNMLN: DEFS 1
2650 FNADR: DEFS 2
2660 RPFLG: DEFS 1
2670 CHR: DEFS 1
2680 RDFRM: DEFS 2
2690 RDLEN: DEFS 1
2700 RDBUF: DEFS 2
2710 SIZE: DEFS 2
2720 BUF2K: DEFS 2048+2048
2730 RXBUF: DEFS 4

```