

```

1 REM MACHINE CODE SHELL SORT & DEMO
2 REM           By Mike Mee
3 REM (c) Computing with the Amstrad
4 REM ----- CPC -----
5 MODE 1:DEFINT a-x
6 sort=&A000:MEMORY sort-1:ln=33
7 FOR adr=sort TO sort+142 STEP 13
8 READ byte$:chk=0
9 FOR i=0 TO 12
10 v=VAL("&" + MIDS(byte$,i*2+1,2))
11 POKE adr+i,v:chk=chk+v
12 NEXT i
13 sum=VAL("&" + RIGHTS(byte$,3))
14 IF chk<>sum THEN PRINT"Error in";l
n:STOP
15 ln=ln+1
16 NEXT adr

```

```

17 first=0:last=100:DIM ara$(last)
18 s$="ABCDEFGHIJKLMNOPQRSTUVWXYZ"
19 FOR x=first TO last
20 GOSUB 45
21 PRINT;ara$(x);" ";
22 NEXT x:PRINT:PRINT
23 INPUT"Ascend/Descend 0/1";ord:PRIN
T
24 IF ord=0 THEN ord=&38 ELSE ord=&30
25 y=TIME/300
26 REM ----- Sort. -----
27 CALL sort,sort,ord,@ara$(first),la
st-first
28 z=TIME/300-y
29 PRINT"Time =";z;"seconds":PRINT
30 FOR x=first TO last
31 PRINT;ara$(x);" ";

```

```

32 NEXT x:END
33 DATA 0D7E04DD6E06DD660711540019478
34 DATA 77DD4E00DD4601DD5E02DD5603539
35 DATA 210300D5E5C838CB193003EB194FC
36 DATA EB78B12920F2C1D5FDE1D1CB3C89B
37 DATA CB1DD5E519F5FDE5C5D5E5462387A
38 DATA 7E23666FEB4E237E23666F040C458
39 DATA 180A1ABE28043809181B13230D1DD
40 DATA 281610F106032108003970E1D13CC
41 DATA D5E51A4E777912132310F7E1D1613
42 DATA C109EB09E3ED5219E3EB30BBFD7AF
43 DATA E1F1E1D138AEED420920A5C900730
44 REM ----- Fill array. -----
45 ara$(x)=RIGHT$(MIDS(s$,RND*22+1,4)
,4)
46 RETURN

```